

**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE CUENCA**

**CARRERA DE INGENIERÍA DE SISTEMAS**

Trabajo de titulación previo  
a la obtención del título de  
Ingeniero de Sistemas.

**PROYECTO TÉCNICO:**

“Tele-educación como servicio (TEaaS) Proyecto integrador Dentalav alojado en la nube pública de Microsoft Azure”

**AUTORES:**

Edison Adrián Carchipulla Morocho

Ronald Vicente Peralta Navarro

**TUTOR:**

Ing. Pablo Leonidas Gallegos Segovia

**CUENCA – ECUADOR**

**2018**

# CESIÓN DE DERECHOS DE AUTOR

Cuenca, Septiembre del 2018

Nosotros, **Edison Adrián Carchipulla Morocho**, con documento de identificación N° **1900627397** y **Ronald Vicente Peralta Navarro**, con documento de identificación N° **0105834014**, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos los autores del trabajo de titulación: **“Tele-educación como servicio (TEaaS) Proyecto integrador Dentalav alojado en la nube pública de Microsoft Azure”**, mismo que ha sido desarrollado para optar por el título de Ingeniero de Sistemas, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hagamos entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



-----  
**Edison Adrián Carchipulla Morocho**  
1900627397



-----  
**Ronald Vicente Peralta Navarro**  
0105834014

# CERTIFICACIÓN

Cuenca, Septiembre del 2018

Yo declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **“Tele-educación como servicio (TEaaS) Proyecto integrador Dentalav alojado en la nube pública de Microsoft Azure”**, realizado por: Edison Adrián Carchipulla Morocho y Ronald Vicente Peralta Navarro, obteniendo este Proyecto Técnico que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.



**Pablo Leonidas Gallegos Segovia**  
C.I 0102593589

# DECLARATORIA DE RESPONSABILIDAD

Cuenca, Septiembre del 2018

Nosotros, **Edison Adrián Carchipulla Morocho**, con documento de identificación N° **1900627397** y **Ronald Vicente Peralta Navarro**, con documento de identificación N° **0105834014**, autores del trabajo de titulación “**Tele-educación como servicio (TEaaS) Proyecto integrador Dentalav alojado en la nube pública de Microsoft Azure**”, certificamos que el total contenido de este proyecto técnico es de nuestra exclusiva responsabilidad y autoría.



-----  
**Edison Adrián Carchipulla Morocho**  
1900627397



-----  
**Ronald Vicente Peralta Navarro**  
0105834014

## AGRADECIMIENTOS

*Queremos hacer un extenso agradecimiento a la Universidad Politécnica Salesiana, a todos los docentes que conforman la carrera de Ingeniería de Sistemas ya que fueron nuestros guías a lo largo de nuestra formación Universitaria.*

*A los Integrantes del grupo de Investigación de Cloud Computing, Smart Cities & High Performance Computing, por el apoyo que nos dieron en todo este tiempo.*

*A nuestros compañeros de la Carrera de Sistemas especialmente a los Compañeros Max Ulloa y Joselo LLivicura por ayudarnos durante las diferentes fases del desarrollo del proyecto.*

*Finalmente queremos expresar nuestro más sincero agradecimiento a nuestro tutor el Ing. Pablo Gallegos que siempre estuvo apoyándonos incondicionalmente a lo largo del proyecto con lo cual más que nuestro tutor llego a ser nuestro mejor amigo.*

***Adrián Carchipulla – Ronald Peralta***

## DEDICATORIA

*A Dios por haberme dado la vida y permitirme el haber llegado hasta este momento tan importante de mi formación profesional, además de su infinita bondad y amor. A mis padres Luis y Mercedes, por el apoyo en todo momento con sus consejos y valores que me han inculcado para ser una persona de bien. A mi abuelita Emilia, por los ejemplos de bondad, respeto, responsabilidad y trabajo que la caracterizan, pero más que nada, por su amor. A mis tíos, en especial a mi tío Carlos, quien ha sido como un padre para mí y que gracias a su apoyo incondicional he logrado concluir mi carrera profesional. A mis hermanos Verónica, Luisa y Luis, por estar conmigo en los malos y buenos momentos, apoyándome de manera incondicional. A mis amigos y compañeros con los cuales he compartido gratos momentos dentro y fuera del establecimiento educativo. A todos mis profesores quienes han sido el pilar fundamental en mi preparación tanto académica como en lo personal y de manera especial al Ing. Pablo Gallegos. A mi familia en general, por sus consejos y siempre alentarme a seguir adelante. Finalmente, a mi compañero Ronald Peralta que sin su apoyo no hubiese sido posible la culminación del presente proyecto.*

***Edison Adrián Carchipulla Morocho.***

## DEDICATORIA

*Quiero dedicar este logro principalmente a Dios por haberme dado la vida, salud y sabiduría para poder culminar este proyecto Técnico con éxito y ayudarme a cumplir las metas que me he propuesto. A mi madre Blanca, mi padre Vicente y a mi hermano Bryam que gracias a sus consejos, valores y amor incondicional han sido el pilar fundamental, mi motivación para seguir adelante y todo este trabajo ha sido posible gracias a ellos. A mis Abuelitas Rosa Matilde, María Esperanza y mi abuelito Manuel, que gracias a sus consejos y cariño incondicional me han brindado todo su apoyo. A toda mi familia por siempre estar apoyándome especialmente a mis Padrinos Ángel, Odalia y a mis tíos Norma, Blanca, Manuel, Imelda, América, Héctor que siempre estuvieron aconsejándome y me han brindado todo su apoyo en toda mi carrera. A mis profesores por brindarme su conocimiento y permitirme crecer como persona y profesional, especialmente al Ing. Pablo Gallegos. A mi amiga Lisseth, en todo este tiempo me ayudo a salir adelante en mi carrera poniendo su confianza en mí, contando con su apoyo de manera incondicional y estando ahí cuando más la necesite. Finalmente, a mis compañeros que siempre estuvieron ayudándome de manera desinteresada, especialmente a mi compañero Adrián Carchipulla sin el cual no hubiera sido posible el desarrollo del proyecto Técnico.*

**Ronald Vicente Peralta Navarro.**

# ÍNDICE DE CONTENIDOS

1	RESUMEN.....	1
2	ABSTRACT.....	2
3	INTRODUCCIÓN.....	3
4	PROBLEMA.....	5
4.1	ANTECEDENTES .....	5
4.2	IMPORTANCIA Y ALCANCES.....	6
4.3	DELIMITACIÓN.....	6
5	OBJETIVOS.....	7
5.1	GENERAL.....	7
5.2	ESPECÍFICOS.....	7
6	TRABAJOS RELACIONADOS.....	7
7	MARCO TEÓRICO .....	12
7.1	TELEEDUCACIÓN.....	12
7.2	E-LEARNING .....	12
7.3	LEARNING MANAGAMENT SYSTEM, LMS .....	13
7.4	CLOUD COMPUTING.....	14
7.4.1	MODELOS DE SERVICIO.....	15
7.4.2	MODELOS DE DESPLIEGUE .....	16
7.5	OPENSTACK .....	18
8	TOPOLOGÍA Y SOLUCIÓN PROPUESTA.....	20
9	MARCO METODOLÓGICO .....	22
9.1	VIRTUALIZACIÓN PREVIA DE LOS SERVICIOS.....	23
9.1.1	SERVIDOR MOOC .....	23
9.1.2	SERVIDOR WEBRTC .....	30
9.2	INSTALACIÓN OPENSTACK.....	32
9.2.1	ACTUALIZACIÓN DE REPOSITORIOS .....	32
9.2.2	INSTALAR GIT .....	33
9.2.3	CLONACIÓN DE OPENSTACK .....	33
9.2.4	CREAR EL ARCHIVO LOCAL.CONF .....	33

9.2.5	CREAR USUARIO OPENSTACK .....	34
9.2.6	INICIAR INSTALACIÓN.....	34
9.2.7	VERIFICACIÓN DE LA INSTALACIÓN .....	34
9.2.8	ACCESO AL DASHBOARD WEB .....	35
9.3	CONFIGURACIÓN DE OPENSTACK .....	36
9.3.1	CREACIÓN DEL ROUTER NEUTRÓN .....	36
9.3.2	CREACIÓN DE LA RED LOCAL (LAN).....	38
9.3.3	CREACIÓN DE UNA SUBRED .....	39
9.3.4	CREACIÓN DEL POOL DE DIRECCIONES DHCP.....	39
9.3.5	CREACIÓN DE SABORES PARA LAS INSTANCIAS.....	41
9.3.6	SUBIR MAQUINA (EXTENSIÓN VDI) DE MOODLE PREINSTALADO.....	42
9.3.7	CREACIÓN DE POLÍTICAS DE SEGURIDAD .....	43
9.3.8	CREACIÓN DE LA INSTANCIA .....	45
9.3.9	VERIFICACIÓN DE LA RED CREADA.....	48
9.3.10	CREACIÓN DE RUTAS.....	49
9.4	INTEGRACIÓN SERVIDOR MOOC CON WEBRTC.....	50
9.4.1	INSTALACIÓN DE MÓDULOS .....	51
9.4.2	COPIAR URL Y CLAVE SECRETA AL SERVIDOR MOOC.....	53
9.5	CONFIGURACIÓN DEL ENTORNO DE DESARROLLO PARA LA APLICACIÓN MÓVIL 53	
9.5.1	INSTALAR GIT .....	53
9.5.2	INSTALAR NODE.JS .....	54
9.5.3	INSTALAR IONIC.....	54
9.5.4	INSTALAR PAQUETES REQUERIDOS .....	54
9.5.5	EJECUTAR LA APLICACIÓN.....	55
9.5.6	CONFIGURAR APLICACIÓN .....	56
9.6	MIGRACIÓN HACIA CLOUD PÚBLICO MICROSOFT AZURE Y FUNCIONAMIENTO 57	
9.6.1	ACCESO A LA PLATAFORMA.....	58
9.6.2	CREACIÓN DE USUARIOS.....	59
9.6.3	CREACIÓN DE CATEGORÍAS.....	60
9.6.4	CREAR CURSO Y ASIGNAR A UNA CATEGORÍA.....	61

9.6.5	MÓDULO DE MATRICULACIÓN USUARIOS .....	61
9.6.6	CREACIÓN DE UNA SESIÓN (VIDEOCONFERENCIA) .....	63
9.6.7	ESTABLECIENDO VIDEOCONFERENCIA .....	66
9.6.8	APLICACIÓN MÓVIL.....	67
9.6.9	CLIENTE HTML5 (OPCIÓN DE WEBRTC EN SMARTPHONE).....	70
10	PRUEBAS Y RESULTADOS .....	72
10.1	PRUEBAS DEL SERVIDOR WEB (MOOC).....	73
10.1.1	CREACIÓN DE CURSOS .....	74
10.1.2	ACCESO CONCURRENTE DE USUARIOS.....	77
10.1.3	DISPONIBILIDAD DE LA PLATAFORMA (ACTIVIDADES) .....	82
10.2	PRUEBAS DE SERVIDOR WEBRTC .....	87
11	CONCLUSIONES .....	91
12	RECOMENDACIONES.....	92
13	REFERENCIAS BIBLIOGRÁFICAS .....	93

## INDICE DE FIGURAS

Figura 1. Abstracción de Cloud Computing. Fuente [23] .....	15
Figura 2. Servicios ofrecidos, en los diferentes tipos de despliegue. Fuente [28].....	16
Figura 3. Arquitectura propuesta. Cloud privado Openstack .....	20
Figura 4. Etapas del proyecto técnico. Fuente Autor .....	22
Figura 5. Versiones a instalar de Moodle.....	26
Figura 6. Selección de idioma para la instalación .....	28
Figura 7. Selección de rutas para la instalación .....	28
Figura 8. Selección de Base de Datos a utilizar .....	29
Figura 9. Configuración de parámetros de la base de datos .....	29
Figura 10. Información general del Administrador .....	30
Figura 11. Arquitectura general de BigBlueButton. Fuente [32] .....	31
Figura 12. Contenido del archivo local.conf.....	34
Figura 13. IP del servidor y credenciales para el acceso web .....	35
Figura 14. Interfaz de acceso a OpenStack .....	35
Figura 15. Menú de Openstack .....	36
Figura 16. Creación de Router Neutrón .....	37
Figura 17. Configuración de parámetros, Router Neutrón .....	37
Figura 18 Listado de Routers en OpenStack .....	37
Figura 19. Opción para crear una red .....	38
Figura 20. Configuración de parámetros, para la nueva red .....	38
Figura 21. Parámetros, para crear la subred local .....	39
Figura 22. Parámetros del pool de direcciones .....	40
Figura 23. Listado de las redes existentes en él OpenStack .....	40
Figura 24. Parámetros para crear un sabor .....	41
Figura 25. Sabores existentes en Openstack .....	41
Figura 26. Creación de imágenes en OpenStack.....	42
Figura 27. Configuración de parámetros para la creación de la imagen en OpenStack..	42
Figura 28. Listado de las imágenes existentes en OpenStack.....	43

Figura 29. Creación de grupos de seguridad en Openstack.....	43
Figura 30. Parámetros para crear un de grupo de seguridad en Openstack.....	44
Figura 31. Parámetros para agregar una nueva regla en OpenStack .....	44
Figura 32. Listado de políticas de seguridad, creadas en OpenStack .....	45
Figura 33. Opción para la creación de una instancia en OpenStack.....	45
Figura 34. Asignación de nombre a la instancia en OpenStack .....	46
Figura 35. Asignación de la imagen vdi a la nueva instancia en OpenStack.....	46
Figura 36. Selección del sabor, para la nueva instancia .....	47
Figura 37. Configuración del tipo de red al que pertenecerá la instancia en OpenStack	47
Figura 38. Configuración, políticas de seguridad para la instancia en OpenStack .....	48
Figura 39. Listado de las instancias creadas en OpenStack .....	48
Figura 40. Topología de las instancias y Enrutador en OpenStack .....	49
Figura 41. Verificación de la interfaz br-ex perteneciente a Neutron .....	49
Figura 42. Creación de ruta en Ubuntu.....	50
Figura 43. Tabla de enrutamiento en Ubuntu .....	50
Figura 44. Instalación de Módulos externos.....	50
Figura 45. Instalación del módulo desde el directorio de extensiones de Moodle.....	51
Figura 46. Página Oficial de Moodle .....	51
Figura 47. Módulos necesarios para sincronizar MOOC con WebRTC .....	52
Figura 48. Actualización de la base de datos MOOC para instalar el módulo BigBlueButtonBN.....	52
Figura 49. URL y clave secreta del Servidor BigBlueButton .....	53
Figura 50. Inicio de la App en Ionic .....	56
Figura 51. Configuración de URL en la App con el servidor MOOC .....	56
Figura 52. Esquema de funcionamiento de la plataforma alojado en la nube pública Microsoft Azure.....	57
Figura 53. Acceso a la página informativa, DENTALAV .....	58
Figura 54. Acceso de usuarios a la plataforma MOOC.....	59
Figura 55. Formulario para registrar un usuario.....	60

Figura 56. Formulario para registrar una categoría .....	60
Figura 57. Formulario para registrar un curso .....	61
Figura 58. Formulario de matriculación de usuarios .....	62
Figura 59. Habilitar la edición del curso.....	63
Figura 60. Creación de una nueva sesión (videoconferencia) .....	63
Figura 61. Ventana emergente para crear actividades.....	64
Figura 62. Parámetros de nueva sesión (videoconferencia), asignación de nombre.....	65
Figura 63. Restricción de la sesión (esperar al moderador y opción de grabar la videoconferencia) .....	65
Figura 64. Asignación de roles en la nueva sesión.....	65
Figura 65. Programación de fecha para la nueva sesión .....	65
Figura 66. Sesión de videoconferencia entre moderador y participantes (estudiantes)	66
Figura 67. Presentación de clase, compartiendo documentos.....	67
Figura 68. Acceso a la Aplicación Móvil .....	68
Figura 69. Vista general de cursos .....	68
Figura 70. Menú principal .....	69
Figura 71. Mensaje de actividades en el curso .....	69
Figura 72. Calificaciones aplicación móvil.....	70
Figura 73. Cliente Html5.....	70
Figura 74. Chat Cliente html5.....	71
Figura 75. Chat público, cliente html5 .....	71
Figura 76. Chat privado, cliente html5.....	72
Figura 77. Tráfico de entrada, creación de cursos .....	75
Figura 78. Tráfico de salida, creación de cursos .....	76
Figura 79. Consumo de memoria RAM, creación de cursos .....	76
Figura 80. Consumo del CPU, creación de cursos.....	77
Figura 81. Tráfico de entrada de usuarios concurrentes (Login – Logout).....	79
Figura 82. Tráfico de salida de usuarios concurrentes (Login – Logout) .....	80
Figura 83. Consumo de RAM, usuarios concurrentes (Login – Logout).....	80

Figura 84. Consumo de CPU, usuarios concurrentes (Login – Logout).....	81
Figura 85. Latencia, usuarios concurrentes (Login – Logout) .....	82
Figura 86. Tráfico de entrada, usuarios concurrentes (Actividades) .....	84
Figura 87. Tráfico de salida, usuarios concurrentes (Actividades) .....	84
Figura 88. Uso de CPU, usuarios concurrentes (Actividades).....	85
Figura 89. Consumo de Memoria RAM, usuarios concurrentes (Actividades).....	86
Figura 90. Latencia, usuarios concurrentes (Actividades) .....	86
Figura 91. Medición de ancho de banda en Openstack.....	87
Figura 92. Medición de ancho de banda en Microsoft Azure.....	88
Figura 93. Tráfico de entrada de Videoconferencias establecidas .....	88
Figura 94. Tráfico de salida de Videoconferencias establecidas .....	89
Figura 95. Consumo de CPU, servidor WebRTC.....	90
Figura 96. Consumo de Memoria RAM, servidor WebRTC.....	90

## INDICE DE TABLAS

Tabla 1. Servicios de Openstack.....	20
Tabla 2. Características del servidor MOOC.....	24
Tabla 3. Características del servidor WebRTC .....	32
Tabla 4. Características del servidor (Openstack).....	32
Tabla 5. Tamaño del curso con sus diferentes actividades.....	75
Tabla 6. Plan de pruebas (usuarios concurrentes).....	78
Tabla 7. Plan de pruebas, concurrencia de usuarios (actividades).....	83

# 1 RESUMEN

El presente proyecto técnico tiene la finalidad de desarrollar un sistema de teleeducación (aprendizaje virtual), que a través del uso de las tecnologías de la información y comunicación TICs permitan la inmersión de nuevos escenarios de enseñanza-aprendizaje en el que se dicten cursos online de diversas carreras, sin embargo, nos enfocaremos en 4 ramas de la medicina odontológica como son: Endodoncia, Ortodoncia, Rehabilitación y Periodoncia. Esta plataforma contará con los servicios de WebRTC, Mensajería Instantánea, Pizarra Digital, Notas compartidas y que a su vez el mismo será desplegado en la nube pública de Microsoft Azure, adicionalmente la integración de una aplicación móvil para brindar mayor ubicuidad y acceso multiplataforma a los cursos.

La plataforma desarrollada se divide en dos etapas: En la primera, desarrollamos un prototipo sobre la nube privada Openstack en el que se despliega dos instancias, la primera instancia para el servidor MOOC, la que se encarga de la gestión y administración de los cursos virtuales. Dentro de la segunda instancia configuramos el servicio de WebRTC, el mismo que permite realizar las Videoconferencias de las clases virtuales entre el profesor y estudiante, mediante salas con herramientas de pizarra digital, mensajería instantánea y notas compartidas; permitiendo dinamizar la enseñanza-aprendizaje.

La segunda etapa es la ejecución de las pruebas de concepto, con la finalidad de determinar las métricas adecuadas de uso de la Memoria RAM, el Ancho de banda, Procesamiento (CPU) y Almacenamiento; necesarias para el correcto funcionamiento de la plataforma en la fase de producción.

## 2 ABSTRACT

The present technical project has the purpose of developing a tele-education system (virtual learning), which through the use of ICT information and communication technologies allow the immersion of new teaching-learning scenarios in which online courses are taught. Different careers, however, we will focus on 4 branches of dental medicine such as: Endodontics, Orthodontics, Rehabilitation and Periodontics. This platform will have the services of WebRTC, Instant Messaging, Digital Whiteboard, Shared Notes, which in turn will be deployed in the Microsoft Azure public cloud, in addition to the integration of a mobile application to provide greater ubiquity and multiplatform access to courses.

The developed platform is divided into two stages: In the first, we developed a prototype on the Openstack private cloud in which two instances are deployed, the first instance for the MOOC server, which is responsible for the management and administration of the virtual courses. Within the second instance we configure the WebRTC service, the same one that allows the Videoconferences of virtual classes between the teacher and student, through rooms with digital whiteboard tools, instant messaging and shared notes; allowing the teaching-learning to be dynamic.

The second stage is the execution of the proofs of concept, in order to determine the appropriate metrics for the use of RAM, Bandwidth, Processing (CPU) and Storage; necessary for the correct functioning of the platform in the production phase.

### 3 INTRODUCCIÓN

La educación a través del uso de las tecnologías de la información y comunicación (TIC), nace a finales del siglo XX y a principios del siglo XXI con la evolución tecnológica, el desarrollo de las tecnologías de la información en la sociedad y su introducción en los sistemas educativos [1]. A partir de esta era surgen las áreas metropolitanas (tele-trabajo, tele-compra, tele-información, tele-diversión y **tele-educación**) que de una forma u otra estos aportan cambios positivos en la vida cotidiana de la sociedad. Así mismo Jorge Salinas autor de “Redes y Educación: Tendencias en educación flexible y a distancia” [1], menciona que las TIC y las Telecomunicaciones son instrumentos que permiten transformar la perspectiva de la educación tradicional mediante la integración y/o adopción de nuevas de tecnologías que facilitan el aprendizaje virtual (educación a distancia).

El termino educación a distancia surge con las Instituciones de Educación Superior en el cual su principal objetivo era impartir conocimientos mediante la utilización del internet. Dicho de otra manera, esta modalidad de educación conlleva a que los participantes del proceso educativo pueden estar separados geográficamente tanto el docente como el estudiante, sin importar el lugar donde cada uno de ellos se encuentren siempre tendrán acceso al material educativo y dicha distancia es cubierta a través de los medios tecnológicos [2].

Los procesos de educación y enseñanza mediante el uso de tecnologías es una idea que ha ido creciendo poco a poco, no solo se trata de formarse para la vida sino a lo largo de toda la vida, ya que cada vez la tecnología avanza a paso agigantados y se necesita siempre estar al tanto de nueva información en todos los ámbitos. Las TIC hoy en día están tomando mucho interés en los procesos de Educación y enseñanza siendo en los últimos años uno de los principales temas de debate, conferencias, papers entre otros.

El estudio y aprendizaje es permanente, los profesionales de la salud emplean muchos métodos para estar siempre al tanto de nueva información y cumplir con su continua educación. Esta educación se daba por medio de revistas, libros, documentos,

asistiendo a conferencias etc. Uno de los problemas que se presenta es que al no tener disponibilidad de tiempo es muy difícil seguir preparándose profesionalmente.

Las nuevas tecnologías orientadas a la enseñanza son herramientas de trabajo que pueden ayudar a las personas otras formas más beneficiosas para el proceso de educación, ya que estar conectados a Internet, se puede ofrecer una serie de herramientas para hacer de las clases mucho más sencillas, eso es lo que pretende ofrecer la Tele-educación como Servicio, incorporar herramientas orientadas a la educación ,para que por medio del Internet sin restricciones de espacio ni tiempo, los profesores puedan dar sus clases y los estudiantes puedan recibirlas sin estar en un mismo lugar.

En la actualidad existen muchas Universidades e Instituciones que han tomado esta metodología, en el cual los estudiantes toman sus clases en línea, los estudiantes se matriculan y obtienen su cuenta de correo, además de la documentación de cómo funciona la Universidad. En caso de un alumno tener dudas o problema, puede hacer una consulta online con el profesor además de poder realizar Videoconferencia.

Por otra parte, actualmente siendo la era digital; el utilizar la tecnología para emprender un aprendizaje de forma virtual y ser nosotros mismo los principales actores de nuestra formación. Porque a través de la teleeducación el paradigma de aprendizaje hace que sea más dinámica con el uso de los foros (compartir opiniones), videoconferencias (preguntas al docente), pizarra digital (explicación gráficamente de temarios importantes). A diferencia de la educación tradicional los cuales se pueden considerar demasiado estáticos [3].

Las principales formas de ofrecer teleeducación como Servicio son: audio, video, pizarras digitales y el uso de dispositivos inteligentes. *“En la década de 1990, hubo avances significativos en información y tecnología de la comunicación. Estos desarrollos habilitados una nueva generación de tecnologías para facilitando la teleeducación.”* [4]

## 4 PROBLEMA

### 4.1 ANTECEDENTES

En el Ecuador y el resto de mundo las personas tienen la necesidad de mejorar sus capacidades competitivas y laborales que le permita mejorar sus ingresos. Sin embargo, se enfrentan a barreras como la falta de tiempo, el cruce de horarios, la distancia geográfica, las situaciones económicas y familiares. Aunque, las tecnologías de la información y comunicación (TICS) proveen de nuevas herramientas que generan escenarios de aprendizaje que lleguen hasta los hogares a través del Internet y de aplicaciones móviles.

Es así que desde sus inicios la educación a distancia, nació conjuntamente con el desarrollo de Internet y el resto de tecnologías de redes, pero con el paso del tiempo, el concepto de la educación a distancia (tele-educación) ha cambiado de tal forma que a la Tele-educación se le puede ver con la unión de Teleformación y Teleaprendizaje. *“El primero conlleva una interacción alumno-profesor en la que el profesor sigue actuando como guía en el proceso educativo, mientras que el segundo representa un sistema de información en el que el alumno es el único responsable de su educación, basándose ésta en materiales lectivos” [5].*

Por otro lado, la teleeducación ha evolucionado desmesuradamente, que en la actualidad existen cursos de toda índole. Existen Universidades virtuales, que no tienen aulas ni alumnos que asisten a clase, son espacios cibernéticos en los que los alumnos que se matriculan obtienen una cuenta de correo y la documentación necesaria para conocer el funcionamiento de la universidad. Cada alumno puede hacer consultas online a los profesores e incluso tener una comunicación de forma virtual a través de videoconferencias. Pero a pesar de todas estas facilidades de tomar los cursos en línea, todavía sigue habiendo vacíos al momento de aprender, por mencionar algunos de ellos; los cursos no son constantemente actualizados; deficiencias en ejercicios prácticos; el enfoque no adecuado a la temática a desarrollarse.

## 4.2 IMPORTANCIA Y ALCANCES

Es importante desarrollar una plataforma de aprendizaje virtual (MOOC), en la cual se dicten cursos online de cualquier ámbito educativo, pero sin embargo en nuestro proyecto técnico nos enfocaremos en un área específica como es la medicina odontológica, ya que dentro del País no existe o acorde a nuestra investigación no hemos encontrado una plataforma que brinde las facilidades de estudiar diferentes ramas de la odontología. Mediante el cual se implementen cursos online tratando temas puntuales sobre la odontología para que de una forma u otra los profesionales de esta área sobresalgan y sigan una continua preparación. Lo que buscamos con esta plataforma es llegar a los profesionales de la medicina odontológica que, a través de un terminal informático y los recursos de Internet, tengan la facilidad y comodidad de escoger el curso o los cursos que más le convengan, contando con la flexibilidad del cómo, cuándo y donde realiza sus actividades didácticas.

Además de ser una plataforma que está relacionada directamente con el proyecto de **DENTALAV**, que es un sistema colaborativo entre el Profesor y el Estudiante. Este módulo está desarrollado en el Sistema de Gestión de Aprendizaje (LMS) Moodle al que se ha modificado y agregado características como WebRTC, mensajería instantánea, pizarra digital entre otras herramientas para hacer que el aprendizaje sea de una forma más interactiva entre el Profesor y el Estudiante.

## 4.3 DELIMITACIÓN

El presente proyecto tiene como finalidad implementar una plataforma de aprendizaje virtual para dictarse cursos online enfocados en 4 ramas de la medicina odontológica como: Endodoncia, Ortodoncia, Rehabilitación y Periodoncia en el que se integra características de WebRTC, mensajería instantánea, pizarra digital entre otras herramientas y que a su vez el mismo será desplegado en una nube pública de Microsoft Azure, adicionalmente la integración de una aplicación móvil para brindar mayor ubicuidad y acceso multiplataforma a los cursos.

## 5 OBJETIVOS

### 5.1 GENERAL

Desarrollar una plataforma en la cual se pueda desplegar un MOOC (Massive Open Online Course) con herramientas colaborativas bajo un Cloud Público (Microsoft Azure), para dictar cursos online.

### 5.2 ESPECÍFICOS

- Desarrollar un prototipo de la plataforma MOOC dentro de una nube privada en Openstack.
- Desplegar recursos de aprendizaje dentro de la plataforma para la evaluación de las funcionalidades de los cursos de aprendizaje.
- Realizar la puesta en producción en la nube pública de Microsoft Azure y realizar el lanzamiento de un primer MOOC de evaluación.
- Desarrollar una aplicación móvil para el acceso a los recursos de nuestra plataforma MOOC.
- Desarrollar la memoria técnica de nuestra plataforma.

## 6 TRABAJOS RELACIONADOS

El autor ROS MARTÍNEZ [6], en su artículo “**MOODLE, LA PLATAFORMA PARA LA ENSEÑANZA Y ORGANIZACIÓN SOCIAL**”, enfatiza el potencial ecológico (permite ahorrar millones de fotocopias en papel) y económico (software libre), que aporta un LMS Open Source como lo es Moodle para el aprendizaje virtual (e-Learning). Además, el autor recomienda usar Moodle porque al ser un software libre y gratis es popularmente utilizado a nivel mundial implementándose en centros educativos; desde

escuelas hasta centros universitarios por diversas ventajas que presenta como son: la facilidad de uso, despliegue multiplataforma, seguridad en los datos, creación de diversas aulas virtuales entre otros. Todo esto combinando con las tecnologías de la información y comunicación (TICS) permite dinamizar la educación a través de ambientes virtuales mediante el uso de dispositivos electrónicos conectados a internet como Computadores, Smartphones y Tabletas.

La autora María de Lourdes Hernández en su tesis denominado **“MÉTODO DE MIGRACIÓN AL MODELO CLOUD COMPUTING SOFTWARE AS A SERVICE DE UN LMS OPEN SOURCE”** [7], menciona que actualmente la implementación de un Sistema de Gestión de Aprendizaje (Learning Management System, LMS) en entornos web requiere de una infraestructura para almacenamiento y comunicaciones para atender una gran demanda de estudiantes, además de estos requisitos también se suman los precios de mantenimiento y mano de obra de personal especializado, lo cual no es conveniente por temas económicos la implementación en infraestructura propia. Entonces, lo que la autora propone es la adopción del modelo de Cloud Computing con un modelo de despliegue tipo SaaS (Software como Servicio) para un LMS Open Source como lo es Moodle. Previamente las pruebas realizadas se ejecutan sobre las interrogantes de “¿Cuáles son las implicaciones técnicas y organizacionales de realizar un proceso de migración hacia cloud computing? ¿Cuáles son los elementos que debe considerar un proveedor para cambiar el modelo de negocio basado en un producto a uno basado en servicios?”. Finalmente implementa el sistema LMS considerando métricas de; acceso concurrente de usuarios, interoperabilidad y facilidad de mantenimiento del software.

José Palacios autor del proyecto **“SISTEMA DE TELE – EDUCACIÓN COMO SERVICIO SOBRE UNA INFRAESTRUCTURA DE COMPUTACIÓN EN LA NUBE PARA LA UNIDAD EDUCATIVA DEL MILENIO ANGEL POLIBIO CHAVES”** [8], en base a encuestas y entrevistas a los estudiantes y docentes, determina que el establecimiento no cuenta con un laboratorio de computación adecuado para fortalecer la educación de los alumnos tanto interna como externa. En base a estos antecedentes, el autor propone

un sistema de Teleeducación sobre un LMS open Source (Moodle) para mejorar la calidad de aprendizaje de los estudiantes. El mismo que se implementa en una infraestructura de computación en la nube sobre **VMware vSphere**, contando con una aplicación web como móvil, los mismos que aportan bajo costo de implementación y ubicuidad de acceso al material educativo; logrando dinamizar la educación por parte de los docentes a través de la educación virtual. Además, a esto implementa un servidor Red5 para video streaming de las clases creadas en cada una de las aulas virtuales.

Los autores del proyecto **“MAIS MÉDICOS PARA BRASIL”** [9], basándose en el paradigma de la educación a distancia proponen la implementación de una herramienta de gestión de aprendizaje LMS Moodle; orientado a la Telesalud, Teleconsultoría, Telediagnóstico con el objetivo de proveer la preparación continua de los profesionales y de esta forma, “puedan desarrollar conocimientos, habilidades y aptitudes que son necesarias para el buen desempeño de sus funciones”, garantizando la calidad de atención en la salud hacia la población. Además, con la Tele-educación integran; conferencias, seminarios, clases y cursos; todo esto desarrollarse por medio de las TIC, para resolver las actividades a distancia en el que se evalúa los conocimientos, se realiza a través de prácticas y lecciones.

Eddie Muñoz autor de la tesis **“IMPLEMENTACIÓN DE ENLACES INALÁMBRICOS APLICADO A TELEEDUCACIÓN EN ZONAS RURALES”** [10], propone la implementación de una plataforma de Teleeducación para la ciudad de Ayacucho-Perú, zona aislada de la tecnología la misma que contempla 5 fases de implementación. La primera fase inicia con la conexión satelital para proveer internet a la zona a través de una red de radio enlaces. Segunda fase es el diseño de la red para dar servicios a los usuarios mediante antenas inalámbricas. Tercera fase, la instalación de un sistema de administración y monitoreo de red. Cuarta fase como servicio extra para la comunidad es la instalación de una centralita de telefonía VoIP. Finalmente, una vez levantada toda la infraestructura de red en la quinta fase realiza la implementación del sistema de teleeducación basada en Moodle para la facultad de ingeniería del Instituto del Paucar

del Sarasara-Ayacucho para ofrecer educación a distancia. Además, a esto le agrega servicios de chat, video conferencia, foros y evaluación todo esto aprovechando las nuevas capacidades que ofrecen las TIC; cambiando la perspectiva de la educación actual.

Wilson Sanipatín, en su proyecto de investigación denominado **“SERVICIO DE TELE-EDUCACIÓN PARA LA MODALIDAD SEMIPRESENCIAL DEL “INSTITUTO TÉCNICO IMBABURA”** [11], este autor plantea una serie de encuestas a docentes, estudiantes sobre que herramientas son necesarias para realizar tutorías, clases de manera interactiva, sin tener una interacción directa entre docentes y estudiantes. Mediante los datos recopilados en las encuestas, implemento su plataforma basada en Moodle, a la cual sometió a una serie pruebas con alumnos y docentes. Mediante estas pruebas la plataforma obtuvo un 90% de aceptación y demuestra que es totalmente aplicable en el instituto.

Leonardo Yamasaki Maza, Carolina Franco Espinosa en su artículo denominado **“SLOR: SIMPLE LEARNING OBJECT REPOSITORY, DISEÑANDO UN REPOSITORIO DE OBJETOS DE APRENDIZAJE PARA LA PLATAFORMA DE TELEDUCACIÓN EDUCAD-UNACH (BASADA EN MOODLE)”** [12], diseñan un SLOR (Repositorio simple de objetos de aprendizaje) para la plataforma Moodle, mediante esta plataforma se pretende facilitar la reutilización, mejorar el diseño de los cursos, además de ofrecer un aprendizaje superior, al mejorar la experiencia de los usuarios en los nuevos modelos educativos. La utilización de un SLOR en la plataforma, beneficio mucho a los profesores ya que cuenta con múltiples recursos para mejorar el aprendizaje de los alumnos, además de la automatización de las tareas.

Los autores M.T. CARRACEDO, C. PÉREZ, P. RAMIREZ y M.B. SALAZAR, en su paper denominado **“IMPLANTACIÓN COORDINADA DEL ENTORNO VIRTUAL MOODLE Y SU UTILIZACIÓN EN LA ESCUELA UNIVERSITARIA DE INFORMÁTICA DE LA UNIVERSIDAD POLITÉCNICA DE MADRID”** [13], realizaron encuestas a los estudiantes para valorar su grado de satisfacción, al recibir distintas asignaturas a través de la plataforma, la misma

que está basada en Moodle. Además, a esto realizan capacitación de asignaturas como; Álgebra, Física, Fundamentos de Computadores, Matemática Discreta, Programación I, en lo cual en los resultados muestran que el grado de aprobación de los alumnos a las asignaturas mencionadas anteriormente, tuvieron un buen grado de satisfacción ya que la puntuación es 3,3 sobre 4 demostrando el potencial de la plataforma, así como otras herramientas que pueden ser muy aprovechados en beneficio de la educación.

Los autores Ana Gallego, Rosario Gimeno, Francisco Gutiérrez Javier Velázquez en su artículo denominado **“TELE ENSEÑANZA: NUEVOS MÉTODOS PARA EL APRENDIZAJE DE LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA APLICADOS A PROYECTOS DE INGENIERÍA”** [14], implementan en la Universidad Politécnica de Madrid una plataforma de teleeducación basada en Moodle, la asignatura a dictarse es **“Aplicación de los S.I.G para la planificación y gestión en los espacios protegidos”**. Los diferentes temas de la asignatura son explicados a través de videos con una duración de 30 minutos mediante streaming. En el cual los estudiantes acceden a la plataforma y disponen de las grabaciones, que a su vez poden ver y escuchar todas las explicaciones del profesor, para después resolver unos ejercicios al cabo de una semana y subir a la plataforma para su revisión; este método tuvo una gran acogida por parte de los profesores y estudiantes demostrando el éxito de ese nuevo método de enseñanza.

Carlos Fernando Meléndez en su proyecto **“PLATAFORMAS VIRTUALES COMO RECURSO PARA LA ENSEÑANZA EN LA UNIVERSIDAD: ANÁLISIS, EVALUACIÓN Y PROPUESTA DE INTEGRACIÓN DE MOODLE CON HERRAMIENTAS DE LA WEB 2.0”** [15], propone el uso de la plataforma Moodle con herramientas WEB 2.0 para mejorar el aprendizaje, donde los docentes pueden preparar sus documentos y usar estas herramientas como blogs, Slideshare, Scribd para interactuar con los estudiantes, además de que los mismos pueden consultar cualquier duda, de esta forma mejorando el proceso de aprendizaje tradicional. Esta propuesta lo hace para la Universidad Técnica de Ambato, para permitir desarrollar; el aprendizaje virtual, la calidad en la educación, mejorar los procesos de innovación, capacitación y aprendizaje entre alumnos y docentes.

## 7 MARCO TEÓRICO

### 7.1 TELEEDUCACIÓN

La teleeducación es la integración de las tecnologías de información y comunicación (TIC) en el ámbito educativo con el objeto de planificar y desarrollar nuevas formas de aprendizaje y otras actividades educativas, en el que la educación se convierte en un entorno dinámico, sin que los participantes tengan que estar simultáneamente en un mismo lugar [16].

Actualmente la teleeducación es una modalidad de aprendizaje a distancia que, a través de un terminal informático como los Ordenadores, Tablet, Smartphones y recursos de internet, el participante cuenta con la flexibilidad de imponer su ritmo de aprendizaje sobre las actividades didácticas.

### 7.2 E-LEARNING

E-Learning traducido al español “aprendizaje electrónico” comprende cualquier actividad educativa que utiliza medios electrónicos como un computador de escritorio, ordenador portátil, Tablet o Smartphone; mediante los cuales se realiza una parte o todo el proceso formativo [17].

El aprendizaje electrónico también se conoce como aprendizaje basado en la web, aprendizaje en línea, instrucción asistida por computadora o aprendizaje basado en internet [18], comúnmente existen dos modos comunes de e-learning: aprendizaje a distancia e instrucción asistida por computadora. El aprendizaje a distancia utiliza tecnologías de la información para proveer el material educativo a los alumnos los cuales tiene acceso remotamente. La instrucción asistida por computadora se realiza a través de la entrega de paquetes multimedia (textos, gráficos, animación, audio o video) independientes para el aprendizaje y la enseñanza; al que los alumnos acceden a través de un computador o teléfono inteligente.

## Ventajas de E-learning frente a la Educación Tradicional [19]

- Extender y facilitar el acceso a la formación a colectivos e individuos que no pueden acceder a la modalidad presencial.
- Incrementar la autonomía y responsabilidad del estudiante en su propio proceso de aprendizaje.
- Superar las limitaciones provocadas por la separación en espacio y/o tiempo del profesor-alumnos.
- Flexibilidad en los tiempos y espacios educativos.
- Aprendizaje colaborativo entre comunidades virtuales de docentes y estudiantes.

Para la implementación de un sistema de formación basado en e-learning es necesario el uso de un LMS (Learning Management System, Sistemas de gestión de aprendizaje), considerado como “plataforma o sistemas de software que permiten la comunicación e interacción entre profesores, alumnos y contenidos” [20] .

### 7.3 LEARNING MANAGMENT SYSTEM, LMS

Un LMS (Sistema de gestión de aprendizaje) es un entorno que se encarga de la administración del proceso de aprendizaje [21]. En otras palabras, es una aplicación instalada en un servidor, que administra, distribuye y controla las actividades de formación de una institución u organización. Por citar un ejemplo de un LMS Open Source tenemos a Moodle el mismo que es utilizado para desarrollo del proyecto técnico.

Los sistemas de gestión de aprendizaje aportan herramientas el cual permite dinamizar el proceso educativo, dichas herramientas son [22]:

#### Herramientas de gestión y distribución de contenidos

Es el encargado de almacenar, organizar, recuperar y distribuir contenidos educativos y estructurarlos en contenido de mayor complejidad y alcance temático.

### Herramientas de administración de usuarios

Encargado del registro de usuarios en el sistema para su control de acceso a la plataforma y presentación personalizada de los contenidos y cursos que este matriculado.

### Herramientas de comunicación

Chats, foros, correo electrónico, tableros de anuncios, permiten la comunicación entre estudiantes y tutores en una vía o en doble vía, síncrona o asíncrona.

### Herramientas de evaluación y seguimiento

Permite la elaboración y presentación de evaluaciones mediante la utilización de diferentes tipos de preguntas: abierta, falso o verdadero, selección múltiple, opción a completar y apareamiento entre otras. Algunas veces también permite la construcción de banco de preguntas usados con frecuencia para seleccionar aleatoriamente y asignar a los estudiantes.

## 7.4 CLOUD COMPUTING

“Cloud Computing es el uso de tecnologías basadas en internet para el suministro de servicios” [23]. Dichas tecnologías pueden estar representados por un conjunto de computadoras personales interconectadas o servidores de red, que al desplegar un Cloud puede hacerse de forma pública o privada.

El Instituto Nacional de Estándares y Tecnología (NIST por su acrónimo en inglés National Institute of Standards and Technology) [24], especifica 5 características esenciales. Dichas características son autoservicio bajo demanda, amplio acceso a la red, convergencia de recursos, rápida elasticidad y optimización de recursos. Además, este mismo organismo menciona tres modelos de servicio. Servicios como; Software

como Servicio (SaaS), Plataforma como Servicio (PaaS) e Infraestructura como Servicio (IaaS). A esto se incluye cuatro modelos de despliegue: nube privada, comunitaria, pública e híbrida.

### 7.4.1 MODELOS DE SERVICIO

A. Marinos and G. Briscoe [23], mencionan que en muchas ocasiones hay confusiones sobre los servicios que cada modelo proporciona y su relación entre ellos. Al instante de optar por un modelo de servicio. Y que para resolver esta confusión en la figura 1, se muestra los diferentes servicios que se acentúan en cada modelo, el cómo están relacionados entre sí y a qué tipo de usuarios está dirigido.

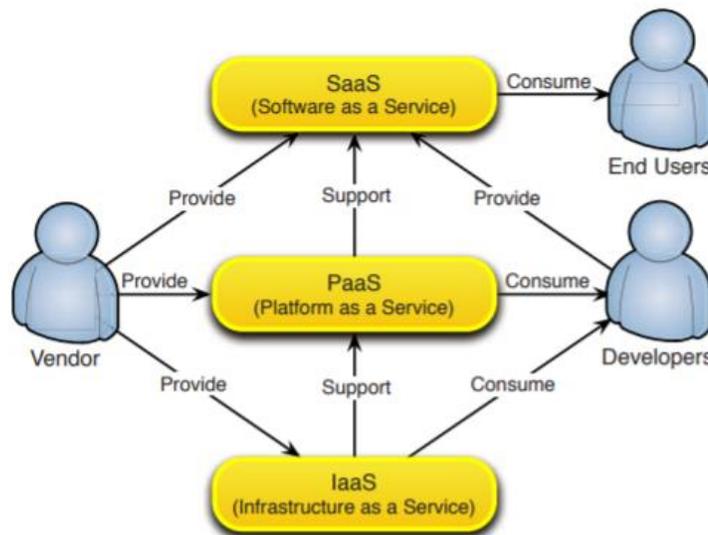


Figura 1. Abstracción de Cloud Computing. Fuente [23]

#### Infraestructura como Servicio (IaaS)

Es la capacidad de recursos que se le provee al cliente, tales recursos como; procesamiento, almacenamiento, redes y otros recursos de computación [24]. Proveedores como Amazon y Microsoft, proporcionan instancias de máquinas a los desarrolladores, donde cliente es capaz de desplegar y ejecutar software arbitrario que pueda incluir sistemas operativos y aplicaciones. Este tipo de servicio está dirigido netamente para desarrolladores.

## Software como Servicio (SaaS)

La facilidad que el cliente tiene de utilizar las aplicaciones que se encuentran alojados en una infraestructura de Cloud Computing. Estas aplicaciones son accesibles desde cualquier dispositivo a conectados a internet a través de una interfaz de cliente ligero, como lo es un navegador. [23] , menciona que los ejemplos más populares que ofrecen este tipo de servicio son: Hotmail de Microsoft, Google Drive y Dropbox.

## Plataforma como Servicio (PaaS)

“Es entregar una plataforma de desarrollo de aplicaciones como un servicio para desarrolladores en la web” [25]. [26] Permite al usuario desplegar sus propias aplicaciones (adquiridas o desarrolladas por el propio usuario), [27] a diferencia de Software como Servicio (IaaS) los clientes no tienen el acceso a la gestión y control de la infraestructura del Cloud, pero si a sus aplicaciones desplegadas.

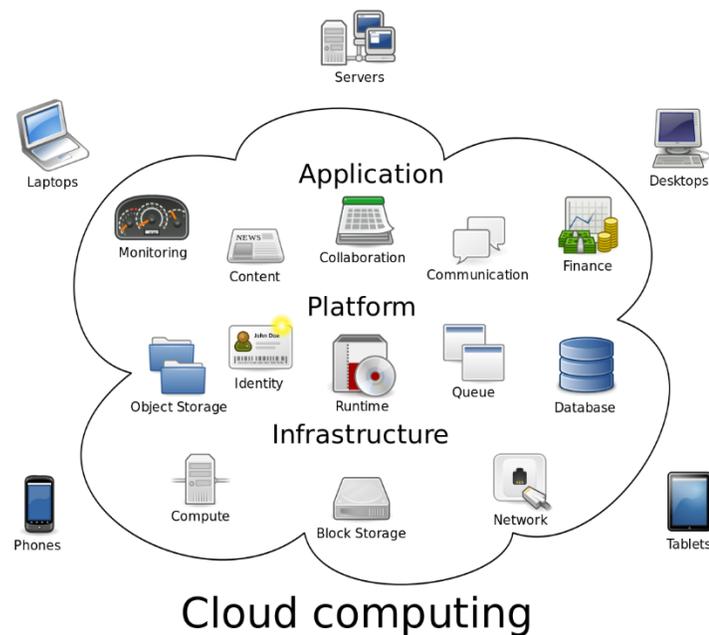


Figura 2. Servicios ofrecidos, en los diferentes tipos de despliegue. Fuente [28]

### 7.4.2 MODELOS DE DESPLIEGUE

#### Cloud Público [24]

Esta infraestructura es operada por un proveedor que provee servicios al público en general. Puede ser propiedad, administrado y operado por una organización empresarial, además u gubernamental o alguna combinación entre ellos.

#### Cloud Privado [27]

Este modelo de despliegue, es una emulación del Cloud Público a diferencia que está desplegada en una red privada. La infraestructura es totalmente gestionada por una organización. Luis Espino [25], menciona algunas ventajas que un Cloud privado trae consigo.

- Disponible en demanda.
- Rápido aprovisionamiento de servicios de negocio.
- Reducción del costo a través de economías a escala.
- Basado en el uso.
- Controlado y asegurado por corporación de IT.

#### Cloud Híbrido [25]

Es una combinación de los modelos públicos y privados, presentando como ventaja aumentar la capacidad de un Cloud privado haciendo uso de los recursos del Cloud público. El organismo NIST [24], menciona que, para poder combinar dos o más infraestructuras, tienen que estar unidas por una tecnología estandarizada o propietaria que permita la portabilidad de datos y aplicaciones (por ejemplo, Cloud Bursting para el balanceo de carga).

#### Cloud Comunitario [24]

Esta modelo de despliegue, está orientado a servir a organizaciones con objetivos comunes (por ejemplo, misión, requisitos de seguridad, políticas y consideraciones de cumplimiento). A la vez puede ser operado y administrado por una o más

organizaciones de la comunidad, un tercero o una combinación entre ellos y puede existir dentro o fuera de las instalaciones.

## 7.5 OPENSTACK

OpenStack [29], es un software de código abierto que permite implementar Computación en la nube de tipo (IaaS, Infraestructura como Servicio), controlando los recursos de computación como CPU, RAM, almacenamiento y red a través de un DataCenter, administrados a través de un panel de control o mediante el API de OpenStack. Principalmente OpenStack trabaja con tecnologías empresariales y de código abierto, lo que le hace ideal para infraestructuras heterogéneas. Para lograr el funcionamiento del servicio según el objetivo para el que haya implementado, se realiza mediante un conjunto de servicios que posee OpenStack. A continuación se presenta la tabla 1 con los diferentes servicios que proporciona Openstack [30].

<b>Servicio</b>	<b>Nombre del proyecto</b>	<b>Descripción</b>
Dashboard	Horizon	Proporciona una interfaz gráfica web basado en los servicios subyacentes de OpenStack permitiendo el lanzamiento de una instancia, la asignación de direcciones IP y la configuración de los controles de acceso, etc.
Compute	Nova	Gestiona el ciclo de vida de las instancias de proceso en un ambiente de OpenStack. Las responsabilidades incluyen la descomposición, programación y cierre definitivo de las maquinas en la demanda.
Networking	Neutron	Permite la conectividad de la red para uno o varios servicios de OpenStack, como OpenStack Compute. Proporciona una API para que los usuarios definan las redes y los archivos adjuntos

		en ellos. Tienen una arquitectura conectable que soporta varios proveedores y tecnologías de redes populares.
<b>Almacenamiento</b>		
Object Storage	Swift	Almacena y recupera los objetos de datos no estructurados de su elección mediante una REST, es una API basada en HTTP. Es muy tolerante a fallos con su réplica de datos y su posible escalabilidad. Su puesta en práctica no es como un servidor de archivos con los directorios montables.
Block Storage	Cinder	Proporciona almacenamiento persistente en bloques para las instancias en ejecución. Soporta dispositivos plug and play y permite gestionar, por lo tanto, todos aquellos dispositivos de almacenamiento en bloque.
<b>Servicios compartidos</b>		
Identity Service	Keystone	Proporciona un servicio de autenticación y autorización para otros servicios de OpenStack. Proporciona también un catálogo de endpoints para todos los servicios de OpenStack.
Image Service	Glance	Permite almacenar y recuperar imágenes de disco de máquinas virtuales. OpenStack Compute hace uso de esta instancia durante la instancia de provisión.
Telemetry	Ceilometer	Monitorea la nube de OpenStack para la evaluación comparativa, la escalabilidad y con fines estadísticos.
<b>Servicios alto nivel</b>		
Orchestration	Heat	Organiza múltiples aplicaciones en la nube de

		material compuesto utilizando ya sea el formato de la plantilla HOT nativo o el formato de la plantilla AWS, tanto a través de una API REST nativa de OpenStack nativo y una forma de nube compatible con la API de querys.
--	--	---

Tabla 1. Servicios de Openstack

## 8 TOPOLOGÍA Y SOLUCIÓN PROPUESTA

El desarrollo del proyecto está dividido en dos etapas. El primero que es el desarrollo de un prototipo sobre el cloud privado Openstack, el mismo que está constituido por dos servidores. El servidor MOOC que esta implementado utilizando un LMS Moodle, encargado de la administración y gestión académica de los cursos a dictarse. Así mismo se tiene un servidor WebRTC para las clases virtuales, a través de Videoconferencias, el cual integra módulos de Pizarra digital, notas compartidas, chat público y privado entre otros, el mismo que será integrado con el servidor MOOC para la creación de salas y control de usuario que acceden a la plataforma virtual.

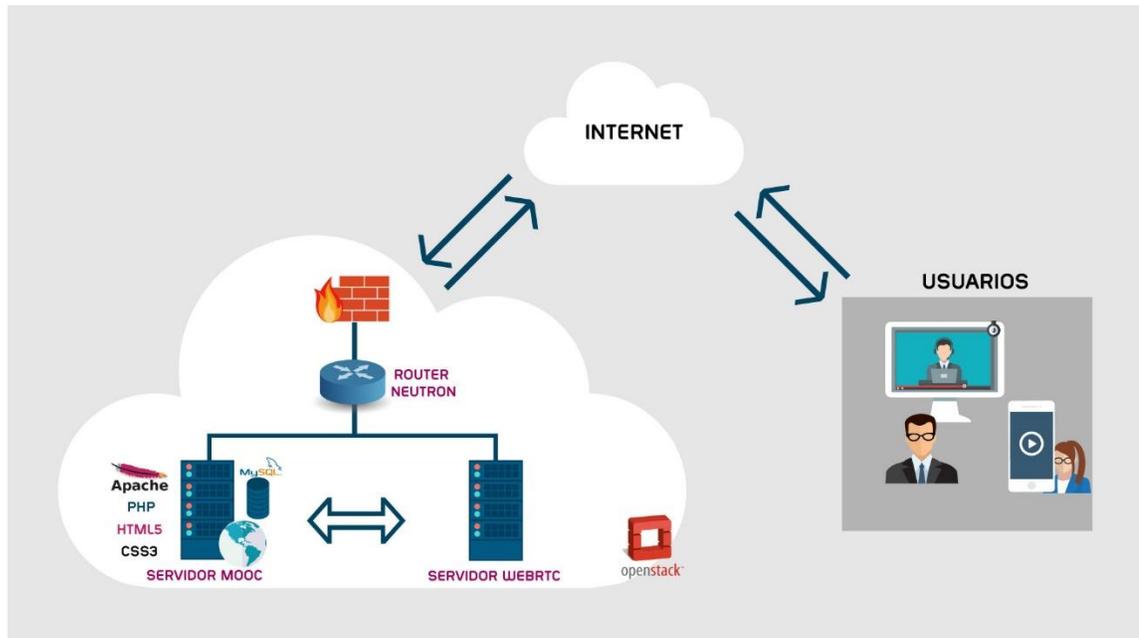
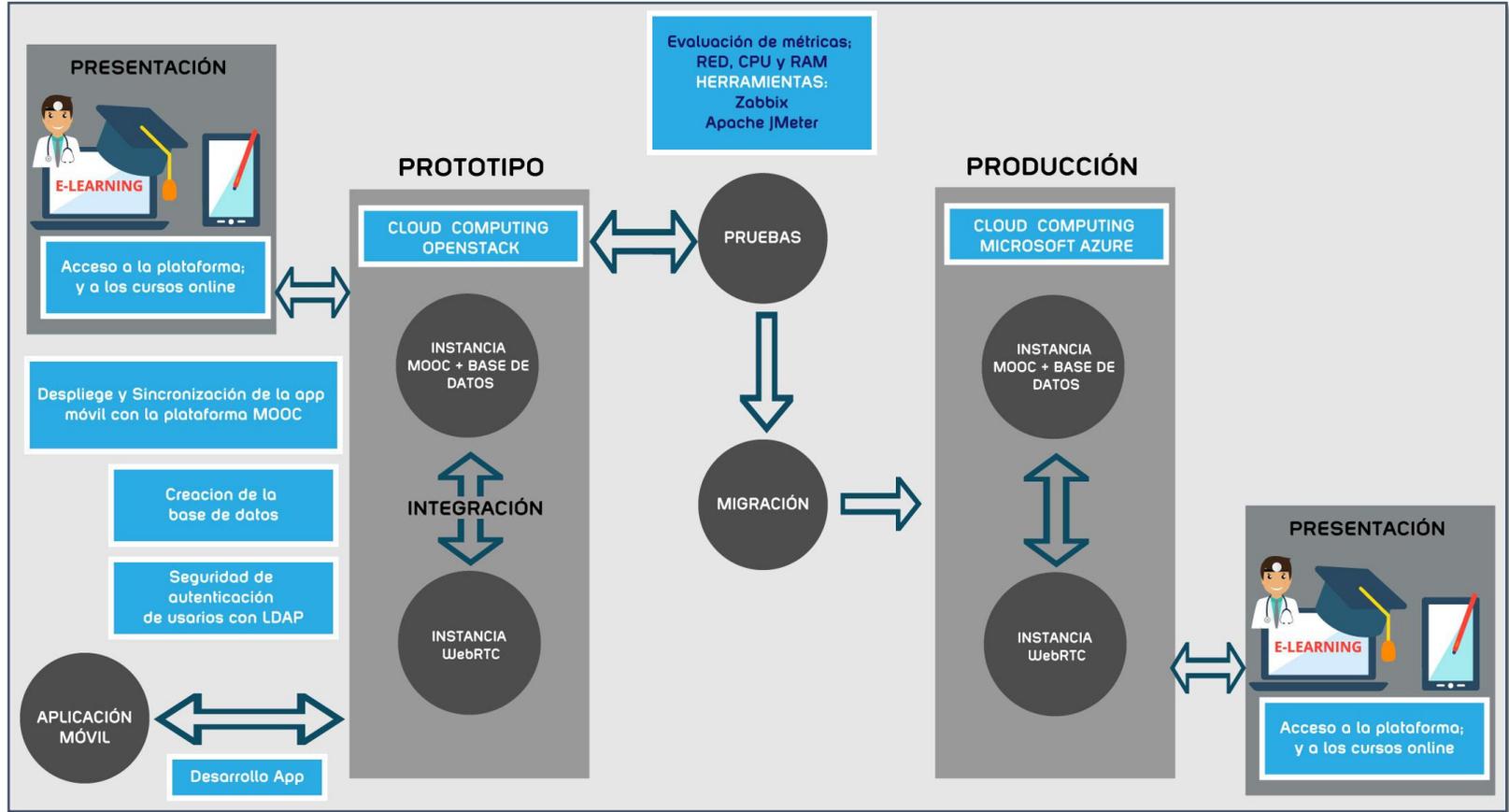


Figura 3. Arquitectura propuesta. Cloud privado Openstack

La segunda etapa es el proceso de migración a la nube pública de Microsoft Azure para brindar ubicuidad multiplataforma de los servicios a través de la página web y aplicación móvil; de los cuales los usuarios dispondrán para el proceso de aprendizaje. Además, la integración del cifrado de datos a nivel de aplicación para ofrecer confidencialidad e integridad de la información solicitada al servidor web.

## 9 MARCO METODOLÓGICO

En esta sección se especifica cómo se ejecutaron las tareas o etapas para el desarrollo de la plataforma Teleeducación como



servicio (TEaaS) (figura 4).

Figura 4. Etapas del proyecto técnico. Fuente Autor

## PROTOTIPO

### 9.1 VIRTUALIZACIÓN PREVIA DE LOS SERVICIOS

Para la implementación de los servicios, se optó inicialmente por la virtualización sobre el Hypervisor Oracle VM VirtualBox, en la que se ejecutó todo el proceso de desarrollo del servidor MOOC como el del WebRTC. A continuación, se muestra el proceso de cada uno de los servidores.

#### 9.1.1 SERVIDOR MOOC

Para el despliegue del servidor MOOC se instaló un LMS Open Source como lo es Moodle, al mismo que se modificara y agregara características para dinamizar las clases virtuales.

La definición de **Moodle** citada de su página oficial, define que, *“es una plataforma de aprendizaje diseñada para proporcionarle a educadores, administradores y estudiantes un sistema integrado único, robusto y seguro para crear ambientes de aprendizaje personalizados”* [31]. Así mismo nos indica que la palabra **Moodle** originalmente proviene del acrónimo de Modular Object-Oriented Dynamic Learning Environment (Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular).

Moodle es proporcionado gratuitamente como un programa de código abierto, bajo la Licencia Publica General, que proviene de su acrónimo en inglés (General Public License - GNU). Esta plataforma nos da la flexibilidad de extender o modificar según nuestros requerimientos, además a esto provee algunas características como son: sin cargos por licenciamiento, constantemente es actualizado implementando mejoras para adaptarse a las necesidades actuales y cambiantes de sus usuarios, disponible en más de 120 idiomas, la agregación de librerías (módulos), flexibilidad de personalizar, escalable y lo más importante la seguridad de los datos y privacidad del usuario.

#### Levantamiento del servidor

Para instalar el LMS, previamente se creó una máquina virtual en el Hypervisor Oracle VirtualBox, el mismo que dispone de las siguientes características:

Recurso	Descripción
<b>Sistema Operativo (SO)</b>	Ubuntu 16.04 LTS Server
<b>Memoria RAM</b>	3 GB
<b>Disco Duro</b>	100 GB
<b>Procesador</b>	1 Core

Tabla 2. Características del servidor MOOC

### Instalación de requisitos

Teniendo el servidor listo, se procede a la instalación de requisitos para el correcto funcionamiento del LMS. Dichos requisitos son los siguientes:

- **Mysql**, Gestor de Base de datos.
- **Apache**, Servidor web HTTP.
- **PHP**, Componente que procesara el código para mostrar el contenido dinámicamente, conectándose con la base de datos.

Desde la terminal del server ejecutamos, el siguiente comando para la instalación de los requisitos.

```
# sudo apt-get install apache2 mysql-client mysql-server php7.0 libapache2-mod-php7.0
```

### Instalar Software Adicional

Es necesario este software adicional para que la plataforma funcione correctamente y no haya fallas al momento de la instalación. Comando de instalación.

```
# sudo apt-get install aspell graphviz php7.0-curl php7.0-gd php7.0-intl php7.0-ldap php7.0-mysql php7.0-pspell php7.0-xml php7.0-xmlrpc php7.0-zip
```

Finalizada la instalación de requisitos y software adicional, se procede a reiniciar Apache para iniciar todos sus módulos de funcionamiento.

```
# sudo service apache2 restart
```

## Instalar Git

Para obtener la versión más reciente de Moodle, es necesario instalar GIT el mismo que sirve para el control de versiones. Ejecutamos el siguiente comando para su instalación.

```
# sudo apt-get install git
```

## Descargar Moodle

Para obtener el sistema completo de Moodle debemos hacer uso del comando git instalado en el paso anterior, primero nos ubicamos en la ruta **/opt**.

```
# sudo cd /opt
```

```
# sudo git clone git://git.moodle.org/moodle.git
```

Ya descargado el sistema de Moodle, renombramos la carpeta por default **MOODLE** a  **cursos**, el comando a ejecutar se muestra a continuación.

```
# sudo mv moodle cursos
```

Accedemos al directorio de cursos y listamos las versiones de Moodle utilizando el comando git conjuntamente con branch. Y escogemos la versión 3.4 que es la estable actualmente.

```
root@moodle:/opt/cursos# sudo git branch -a
* master
-a
remotes/origin/HEAD -> origin/master
remotes/origin/MOODLE_13_STABLE
remotes/origin/MOODLE_14_STABLE
remotes/origin/MOODLE_15_STABLE
remotes/origin/MOODLE_16_STABLE
remotes/origin/MOODLE_17_STABLE
remotes/origin/MOODLE_18_STABLE
remotes/origin/MOODLE_19_STABLE
remotes/origin/MOODLE_20_STABLE
remotes/origin/MOODLE_21_STABLE
remotes/origin/MOODLE_22_STABLE
remotes/origin/MOODLE_23_STABLE
remotes/origin/MOODLE_24_STABLE
remotes/origin/MOODLE_25_STABLE
remotes/origin/MOODLE_26_STABLE
remotes/origin/MOODLE_27_STABLE
remotes/origin/MOODLE_28_STABLE
remotes/origin/MOODLE_29_STABLE
remotes/origin/MOODLE_30_STABLE
remotes/origin/MOODLE_31_STABLE
remotes/origin/MOODLE_32_STABLE
```

Figura 5. Versiones a instalar de Moodle

Finalmente descargamos la versión 3.4 de Moodle, comando a ejecutar.

```
# sudo git branch --track MOODLE_34_STABLE origin/MOODLE_34_STABLE
```

### Copiar el repositorio local

Copiamos la carpeta cursos al directorio /var/www/html/ para posteriormente ejecutar el proceso de instalación.

```
# sudo cp -R /opt/cursos /var/www/html/
```

### Crear directorio moodata y asignación de propietario <<www-data>>

Este directorio es el lugar donde se almacena todos los archivos que son subidos o creados desde la interfaz de Moodle. Y a su vez es necesario inicialmente para la instalación de un sitio MOOC. Por otra parte, se utiliza el comando chown para cambiar de propietario al nuevo directorio.

```
# sudo mkdir /var/moodata
```

```
# sudo chown -R www-data /var/moodata
```

### CREACIÓN Y CONFIGURACIÓN DE BASE DE DATOS

Iniciamos el gestor de base de datos Mysql desde el terminal del servidor e ingresamos la contraseña proporcionada al momento de la instalación.

```
# mysql -u root -p
```

Creamos la base de datos para Dentalav

```
CREATE DATABASE 'nombreBD' DEFAULT CHARACTER SET utf8 COLLATE  
utf8_unicode_ci;
```

### Creación del usuario

Posteriormente creamos el usuario, en nuestro caso llamado **carper**, el mismo que se encargara de la administración de la Base de Datos Dentalav. Además de asignarle una contraseña.

```
create user 'nombreUsuario'@'localhost' IDENTIFIED BY 'contraseñaUsuario';
```

### Asignación de permisos al usuario

Y por último le otorgamos el permiso al usuario **carper** sobre la base Dentalav

```
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,CREATE TEMPORARY  
TABLES,DROP,INDEX,ALTER ON nombreBD.* TO nombreUsuario@localhost  
IDENTIFIED BY 'contraseñaUsuario';
```

### Instalación de Moodle

Para finalizar con la implementación del servidor MOOC, se procede a instalar Moodle desde un navegador Web, para lo cual accedemos con la dirección IP y la ruta que contiene los repositorios de Moodle. Por ejemplo, IP/cursos. A continuación, se presenta imágenes de la configuración de parámetros que son importantes para este proceso.

### Seleccionar idioma

Por favor, seleccione un idioma para el proceso de instalación. Este idioma se usará también como idioma por defecto del sitio, si bien puede cambiarse más adelante.

Idioma

[Siguiete »](#)

*Figura 6. Selección de idioma para la instalación*

Configuración de rutas, donde se especifica la dirección IP del servidor, directorio de la carpeta donde está el código de Moodle y el directorio de datos; todos estos creados anteriormente.

### Confirme las rutas

**Dirección Web**  
Dirección web completa para acceder a Moodle. No es posible acceder a Moodle utilizando múltiples direcciones. Si su sitio tiene varias direcciones públicas debe configurar redirecciones permanentes en todas ellas, excepto en ésta. Si su sitio web es accesible tanto desde una intranet como desde Internet, escriba aquí la dirección pública y configure su DNS para que los usuarios de su intranet puedan también utilizar la dirección pública.

**Directorio de Moodle**  
Ruta completa del directorio que contiene el código de Moodle.

**Directorio de Datos**  
Usted necesita un espacio donde Moodle puede guardar los archivos subidos. En este directorio debe poder LEER y ESCRIBIR el usuario del servidor web (por lo general 'nobody', 'apache' o 'www-data'), pero no debe poderse acceder a esta carpeta directamente a través de la web. El instalador tratará de crearla si no existe.

**Dirección Web**

**Directorio de Moodle**

**Directorio de Datos**

[« Anterior](#) [Siguiete »](#)

*Figura 7. Selección de rutas para la instalación*

En este apartado seleccionamos la base de datos a utilizar el mismo que se instaló en pasos anteriores.

**Seleccione el controlador de la base de datos**

Moodle soporta varios tipos de servidores de base de datos. Por favor, póngase en contacto con el administrador del servidor si no sabe qué tipo usar.

Tipo

Figura 8. Selección de Base de Datos a utilizar

Continuando con la instalación en este punto se configura los parámetros de la base de datos, tal como se muestra en la figura 9.

**Ajustes de base de datos**

**MySQL mejorado (native/mysqli)**

Ahora tiene que configurar la base de datos donde se almacenarán la mayoría de los datos de Moodle. La base de datos solo podrá crearse si el usuario de la base de datos tiene los permisos necesarios. El nombre de usuario y la contraseña ya deben existir. El prefijo de la tabla es opcional.

Servidor de la base de datos

Nombre de la base de datos

Usuario de la base de datos

Contraseña de la base de datos

Prefijo de tablas

Puerto de la base de datos

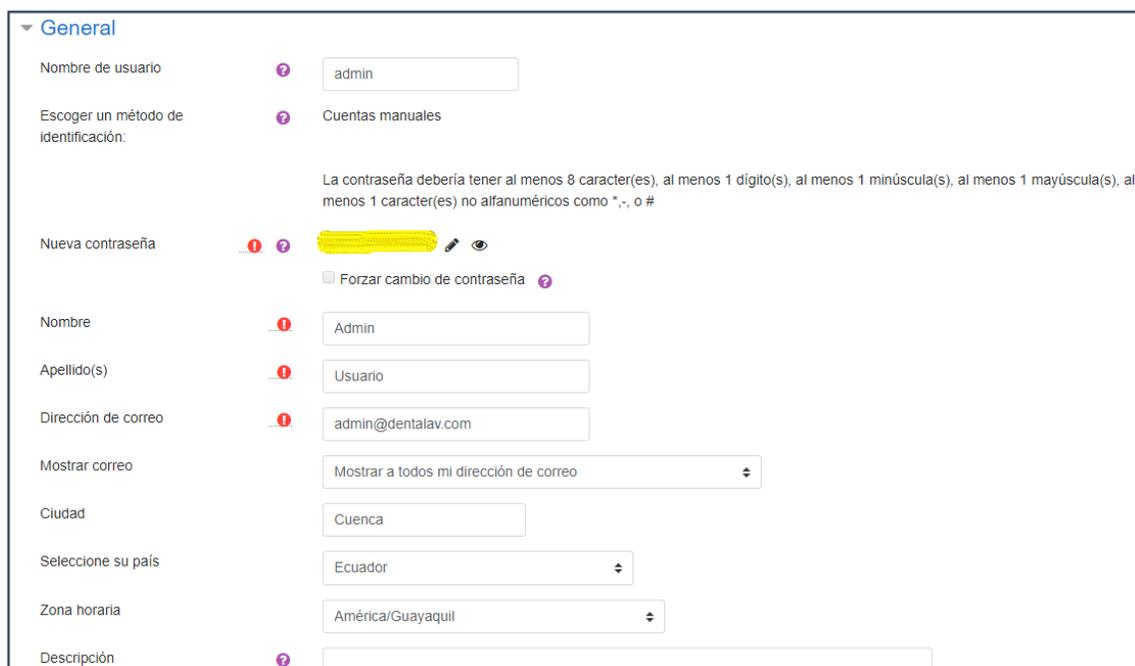
Socket Unix

Figura 9. Configuración de parámetros de la base de datos

Después de la configuración de estos parámetros habrá que realizar una serie de pasos más hasta llegar a la instalación.

[Actualizar información del Administrador](#)

Finalizada la instalación de Moodle habrá que actualizar la información del administrador del sitio como se muestra en la figura 10.



The screenshot shows the 'General' settings for the Moodle administrator. The fields and their values are as follows:

- Nombre de usuario:** admin
- Escoger un método de identificación:** Cuentas manuales
- Nueva contraseña:** [Redacted]
- Nombre:** Admin
- Apellido(s):** Usuario
- Dirección de correo:** admin@dentalav.com
- Mostrar correo:** Mostrar a todos mi dirección de correo
- Ciudad:** Cuenca
- Seleccione su país:** Ecuador
- Zona horaria:** América/Guayaquil
- Descripción:** [Empty]

Figura 10. Información general del Administrador

### 9.1.2 SERVIDOR WEBRTC

Para levantar el servicio de Videoconferencia, se realizará a través del software BigBlueButton (BBB). En su página oficial, lo define como, “un sistema de conferencia web de código abierto para el aprendizaje en línea. El objetivo del proyecto es permitirles a los docentes involucrar a estudiantes remotos en una experiencia de aprendizaje en línea de alta calidad” [32].

Para el funcionamiento de BBB es necesario entender su arquitectura, la misma que está compuesta por algunos componentes de software como nginx, red5, FreeSwitch, Tomcat 7, redis entre otros. A continuación se describe brevemente cada uno de los componentes.

**Nginx:** Es un servidor proxy HTTP (web) y reveso, un servidor proxy de correo y un servidor proxy genérico TCP/UDP, siendo una alternativa rápida, ligera y de código abierto similar a **Apache** [33].

**Red5:** Es un servidor basado en Flash Media Server utilizado para comunicaciones en tiempo real; utilizando sintaxis de *ActionScript Communication*.

**FreeSwitch:** Es un software de código abierto, diseñado para enrutar e interconectar protocolos de comunicación usando audio, video, texto o cualquier otra forma de medio. Considerándose como una plataforma versátil por ser utilizado en comunicaciones de voz, video y chat en dispositivos que van desde llamadas únicas en un Raspberry Pi hasta grandes Centro de Datos (Clusters) [34].

**Tomcat:** Es un servidor web multiplataforma, publicado bajo la licencia de Apache versión 2 y que funciona como contenedor de servlets, implementado especificaciones de los servlets, Java Server Pages (JSP) y Sockets [35].

**Redis:** Es un motor de base de datos en memoria de código abierto con licencia (BSD). Basada en el almacenamiento de datos como cadenas, hashes, listas, mapas de bits y otros; usada opcionalmente como una base de datos durable o persistente [36].

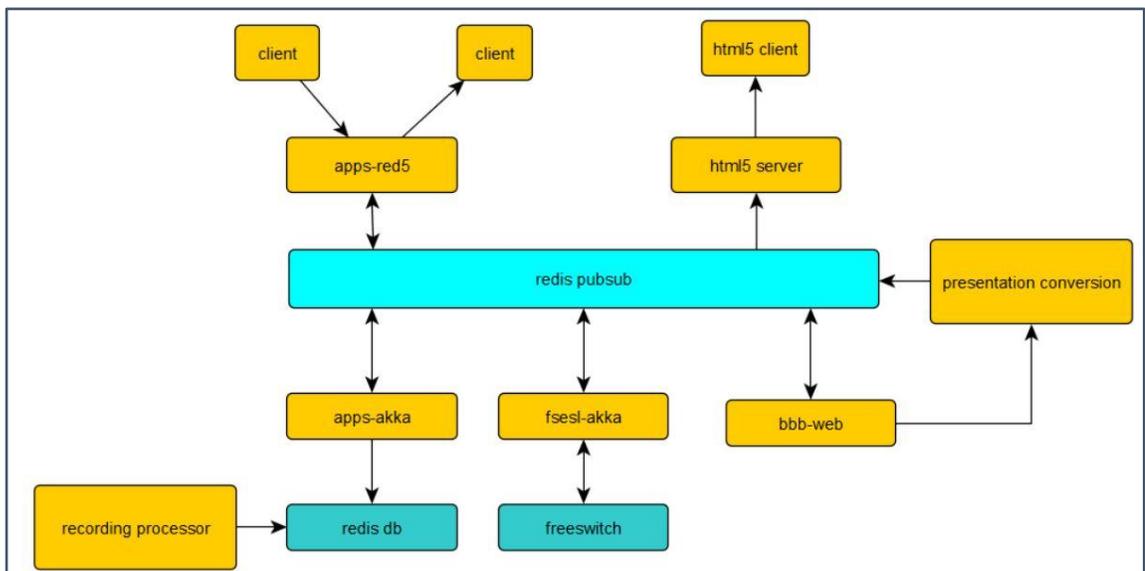


Figura 11. Arquitectura general de BigBlueButton. Fuente [32]

Para realizar la previa instalación del servicio WebRTC, antes se crea una máquina virtual con las siguientes características.

Recurso	Descripción
<b>Sistema Operativo (SO)</b>	Ubuntu 16.04 LTS Server
<b>Memoria RAM</b>	4 GB
<b>Disco Duro</b>	100 GB
<b>Procesador</b>	1 Core

*Tabla 3. Características del servidor WebRTC*

## 9.2 INSTALACIÓN OPENSTACK

Para instalar OpenStack, previamente se creó una máquina virtual en el Hypervisor VMware Workstation con las siguientes características descritas a continuación (tabla 4).

Recurso	Descripción
<b>Sistema Operativo (SO)</b>	Ubuntu 16.04 LTS Desktop
<b>Memoria RAM</b>	12 GB
<b>Disco Duro</b>	300 GB
<b>Procesador</b>	2 Cores

*Tabla 4. Características del servidor (Openstack)*

### 9.2.1 ACTUALIZACIÓN DE REPOSITORIOS

Luego de crear de la máquina virtual; y la instalación el sistema operativo Ubuntu, es necesario realizar la actualización de los repositorios y del Sistema; para lo cual se ejecuta los siguientes comandos.

```
# sudo apt-get update
```

```
# sudo apt-get upgrade
```

### 9.2.2 INSTALAR GIT

De forma similar al despliegue del servidor MOOC, también se procede a instalar git para obtener la versión más reciente de Openstack.

```
# sudo apt-get install git
```

### 9.2.3 CLONACIÓN DE OPENSTACK

A continuación, haremos uso del comando **git** para clonar **devstack**. Devstack es un conjunto de script bash que se encargará de la clonación automática de OpenStack Ocata, descargando sus respectivos componentes para su funcionamiento.

```
# cd /  
  
# sudo git clone https://git.openstack.org/openstack-dev/devstack -b  
stable/ocata
```

**Nota:** Si desea instalar una versión diferente de OpenStack, como newton, pike, mikata o queens, simplemente cambie “ocata” en el comando anterior, por la distribución que desee instalar.

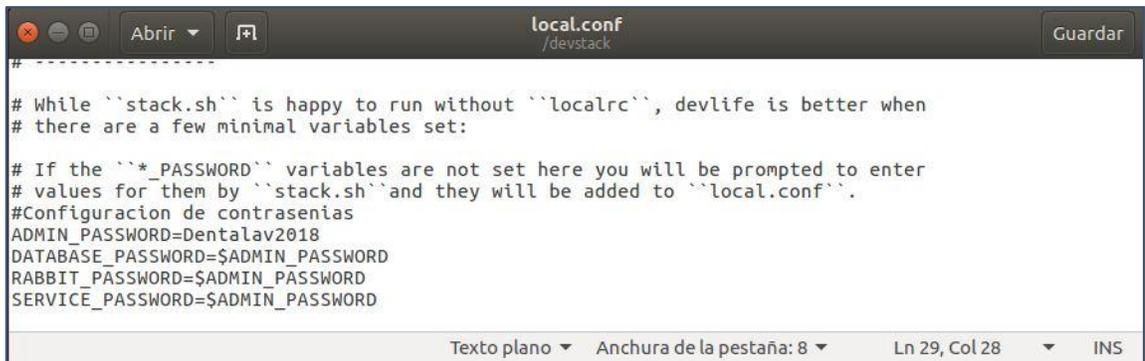
### 9.2.4 CREAR EL ARCHIVO LOCAL.CONF

A continuación, necesitamos copiar la plantilla del archivo **sample.local.conf** a nuestro nuevo archivo local.conf y establecer una contraseña que se utilizara durante el proceso de instalación, la misma que servirá para poder ingresar al entorno grafico una vez terminada la instalación de OpenStack.

```
# cd devstack/  
  
# sudo cp samples/local.conf local.conf  
  
# sudo gedit local.conf
```

Una vez abierto el archivo de configuración nos desplazamos hacia la sección de contraseñas (**\*\_PASSWORDS**). Y le asignamos una contraseña a **ADMIN\_PASSWORD** y

cambiar los otros 3 a **\$ADMIN\_PASSWORD**. Esto hará que los servicios utilicen la misma contraseña durante la instalación. Tal como se muestra en la figura 12.



```
#-----  
# While ``stack.sh`` is happy to run without ``localrc``, devlife is better when  
# there are a few minimal variables set:  
  
# If the ``*_PASSWORD`` variables are not set here you will be prompted to enter  
# values for them by ``stack.sh`` and they will be added to ``local.conf``.  
#Configuracion de contraseñas  
ADMIN_PASSWORD=Dentalav2018  
DATABASE_PASSWORD=$ADMIN_PASSWORD  
RABBIT_PASSWORD=$ADMIN_PASSWORD  
SERVICE_PASSWORD=$ADMIN_PASSWORD
```

Figura 12. Contenido del archivo local.conf

### 9.2.5 CREAR USUARIO OPENSTACK

Posteriormente, ejecutamos un script para crear un nuevo usuario para OpenStack, y luego convertiremos a ese nuevo usuario en el propietario de la carpeta **devstack**.

```
# sudo /devstack/tools/create-stack-user.sh  
  
# sudo chown -R stack:stack /devstack
```

### 9.2.6 INICIAR INSTALACIÓN

Una vez realizado todos los pasos anteriores, finalmente se procede con la instalación de Openstack. Este proceso tardara entre 30 a 40 minutos, dependiendo de la velocidad de nuestro internet. Para lo cual se ejecutará el comando **stack.sh**

```
# sudo su stack  
  
# /devstack/stack.sh
```

### 9.2.7 VERIFICACIÓN DE LA INSTALACIÓN

Concluida la instalación de Openstack nos presenta la siguiente información donde, la primera línea nos indica la dirección IP con la que se ingresará al Dashboard y las líneas 5 y 6 son las credenciales para acceder a su entorno Web, también podemos observar en la línea 7 la versión instalada, figura 13.

```
This is your host IP address: 192.168.88.70
This is your host IPv6 address: ::1
Horizon is now available at http://192.168.88.70/dashboard
Keystone is serving at http://192.168.88.70/identity/
The default users are: admin and demo
The password: Dentalav2018
DevStack Version: ocata
Change: 0c980c6f8162e04c69ecfa46119e0d814963a764 Do not use pip 10 or higher 2018-04-16 11:57:50 +0000
OS Version: Ubuntu 16.04 xenial

stack@carper:~/devstack$
```

Figura 13. IP del servidor y credenciales para el acceso web

### 9.2.8 ACCESO AL DASHBOARD WEB

Para ingresar a su entorno web, en nuestro navegador ingresamos la **IP** que nos dio al finalizar la instalación, figura 13; y en el apartado de usuario y contraseña ingresamos las credenciales proporcionadas en la misma; que a su vez la contraseña es el misma que se agregó en el archivo de (**local.conf**), todo este proceso se muestra en la figura 14.

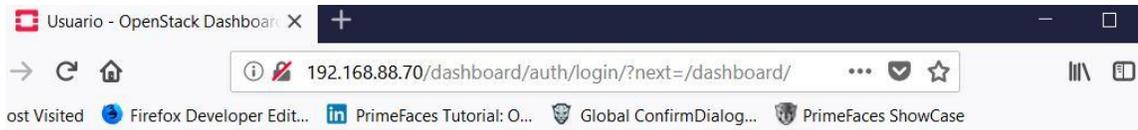


Figura 14. Interfaz de acceso a OpenStack

### 9.3 CONFIGURACIÓN DE OPENSTACK

Una vez dentro de la página de administración de Openstack se procede a la configuración del Cloud, pero antes verificamos que Neutron esté instalado correctamente; para verificar nos dirigimos al menú lateral izquierdo como se muestra en la figura 16. Neutron es un API que permite crear “redes como servicio”, además de la configuración de las políticas de red.



Figura 15. Menú de Openstack

#### 9.3.1 CREACIÓN DEL ROUTER NEUTRÓN

Para crear, nos ubicamos en el apartado **Project** → **Network** → **Router** y procedemos a crear un nuevo Router.



Figura 16. Creación de Router Neutrón

Al seleccionar *Create Router* se nos desplegará una pantalla similar al el de figura 17, donde se deberá asignar un nombre y el tipo de Red en este caso *public*.

Figura 17. Configuración de parámetros, Router Neutrón

Finalizada la creación del Router, se observará una pantalla similar a la figura 18.

Name	Status	External Network	Admin State	Actions
Router1	Active	public	UP	Clear Gateway

Figura 18 Listado de Routers en OpenStack

### 9.3.2 CREACIÓN DE LA RED LOCAL (LAN)

Lo siguiente será configurar una red local, la misma que servirá para el alojamiento de las instancias y la conexión entre la red pública y local. Para esto nos ubicaremos en el menú y seleccionaremos **Project** → **Network** → **Networks** y seguidamente seleccionamos la opción **Create Network**.

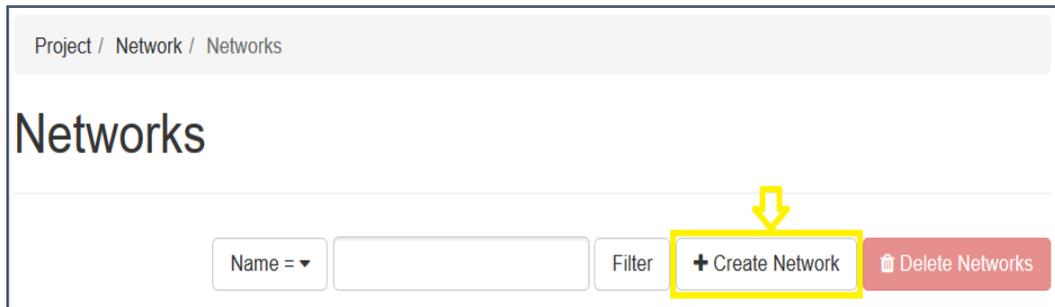


Figura 19. Opción para crear una red

Similar a la creación del Router, nos pedirá un nombre y también seleccionamos la opción *Enable Admin State* como se muestra en la figura 20.

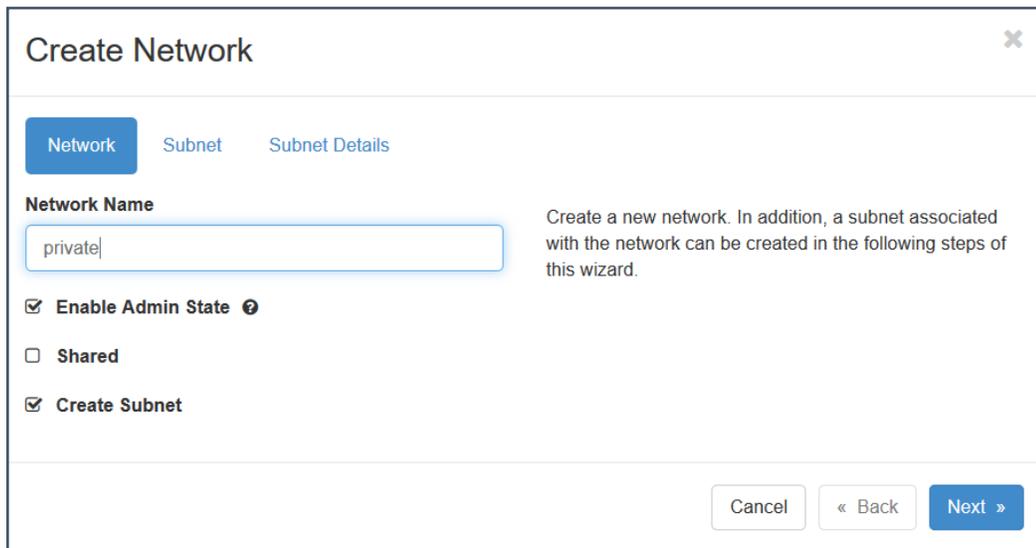


Figura 20. Configuración de parámetros, para la nueva red

### 9.3.3 CREACIÓN DE UNA SUBRED

Para crear nuestra subred, debemos asignarle a nuestra red local un nombre, direccionamiento IP, versión del Protocolo de Internet (IP) y su Gateway. Todos estos parámetros deberán ser configurados tal como se muestra en la Figura 21.

**Crear red**

Red **Subred** Detalles de Subred

**Nombre de subred**  
private-subnet

**Origen Dirección de Red**  
Introducir la Dirección de Red manualmente

**Direcciones de red**  
192.168.5.0/24

**Versión de IP**  
IPv4

**IP de la puerta de enlace**  
192.168.5.1

Deshabilitar puerta de enlace

Crea una subred asociada a la red. Es necesario añadir una "dirección de red" y una "IP de la puerta de enlace" válidos. Si no añade una "IP de la puerta de enlace", el primer valor de la red se asignará por defecto. Si no quiere puerta de enlace, seleccione "Deshabilitar puerta de enlace". La configuración avanzada está disponible haciendo click en la pestaña "Detalles de subred".

Cancelar < Anterior **Siguiete >**

Figura 21. Parámetros, para crear la subred local

### 9.3.4 CREACIÓN DEL POOL DE DIRECCIONES DHCP

Para finalizar la configuración de la red local, agregamos el rango de direcciones y sus DNS que serán posteriormente serán asignadas a las instancias que crearemos en él OpenStack.

Figura 22. Parámetros del pool de direcciones

Al finalizar la configuración y creación de la red tendremos una interfaz como se muestra en la figura 23 donde, tenemos una red pública y otra de tipo privada (LAN).

<input type="checkbox"/>	Nombre	Subredes asociadas	Compartido	Externa	Estado	Estado de administración	Acciones
<input type="checkbox"/>	private	private-subnet 192.168.5.0/24	no	no	Activo	ARRIBA	Editar red ▼
<input type="checkbox"/>	public	public-subnet 172.24.4.0/24 ipv6-public-subnet 2001:db8::/64	no	Sí	Activo	ARRIBA	Editar red ▼

Figura 23. Listado de las redes existentes en el OpenStack

### 9.3.5 CREACIÓN DE SABORES PARA LAS INSTANCIAS

El crear los sabores nos permite definir los recursos que utilizarán cada una de nuestras instancias y no las predefinas por Openstack. Para crear nos ubicamos en el menú **Project** → **Administrator** → **Flavors** y seleccionamos la opción **Create Flavor**. En nuestro caso creamos uno para la instancia MOOC asignándole un nombre, CPU, Memoria y RAM; tal como se muestra en la figura 24.

Información del sabor \* Acceso al sabor

**Nombre \***  
disk-mooc

**ID**  
auto

**VCPU \***  
1

**RAM (MB) \***  
3072

**Disco raíz (GB) \***  
100

**Disco efimero (GB)**  
0

Los sabores definen los tamaños de memoria RAM, disco, número de cores y otros recursos que pueden ser seleccionados por los usuarios al desplegar instancias.

Figura 24. Parámetros para crear un sabor

Luego de crear el sabor nos mostrara el listado de todos los sabores existentes, tal como se puede observar en la figura 25.

Mostrando 13 artículos

<input type="checkbox"/>	Nombre del sabor	VCPU	RAM	Disco raíz	Disco efimero	Disco de intercambio (swap)	Factor RX/TX	ID	Publico	Metadatos	Acciones
<input type="checkbox"/>	cirros256	1	256MB	0GB	0GB	0MB	1,0	c1	Si	no	Editar Sabor
<input type="checkbox"/>	disk-mooc	1	3GB	100GB	0GB	0MB	1,0	87935a4a-ae6f-481b-8b77-ade6fc23f396	no	no	Editar Sabor
<input type="checkbox"/>	ds1G	1	1GB	10GB	0GB	0MB	1,0	d2	Si	no	Editar Sabor

Figura 25. Sabores existentes en Openstack

Para continuar creando más sabores lo hacemos siguiendo el mismo procedimiento expuesto anteriormente.

### 9.3.6 SUBIR MAQUINA (EXTENSIÓN VDI) DE MOODLE PREINSTALADO.

A continuación, se debe subir la imagen del sistema operativo, pero en nuestro caso subiremos un VDI de la maquina preinstalada el LMS Moodle en Ubuntu 16.04, dicha preinstalación se realizó en VirtualBox y le exportamos con el tipo de extensión **VDI** para cargarle a OpenStack. Para esto nos iremos al menú **Project** → **Compute** → **Images** y seleccionaremos la opción **Create Image** como se muestra en la figura 26.

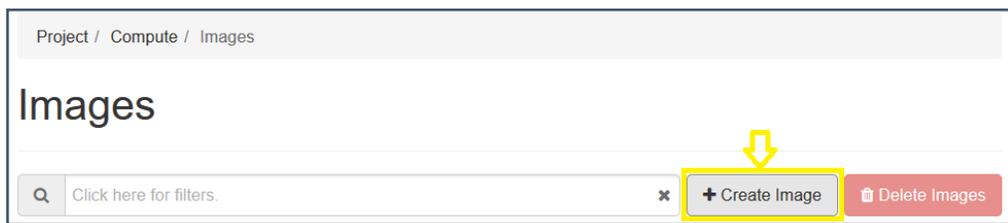


Figura 26. Creación de imágenes en OpenStack

Una vez seleccionada la opción crear imagen; se debe llenar el nombre de la imagen y el formato de la imagen y por ultimo escoger el archivo a subir, dicha configuración se muestra en la figura 27.

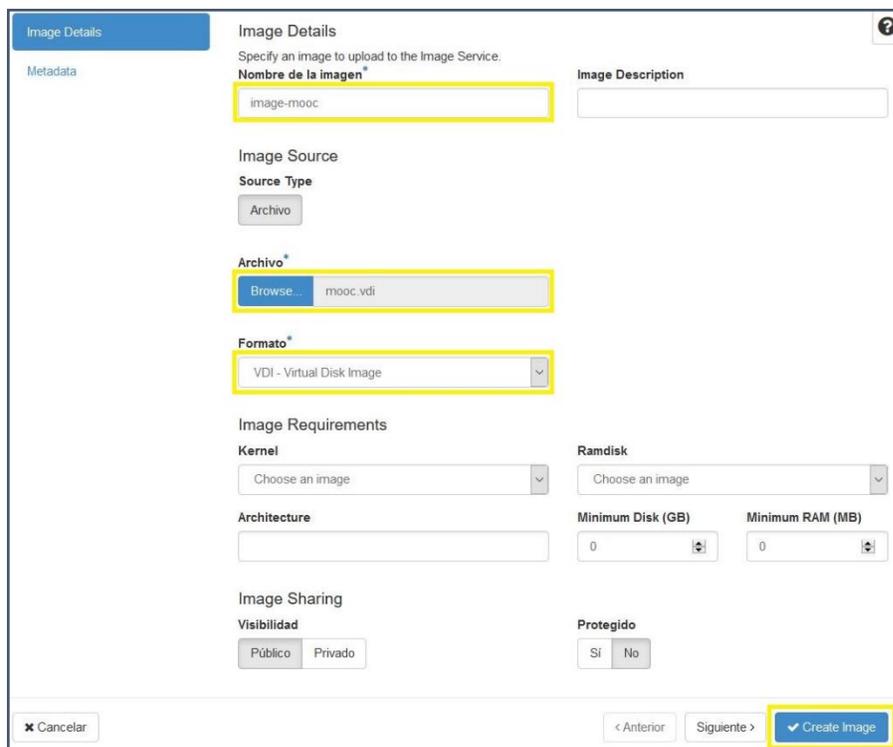


Figura 27. Configuración de parámetros para la creación de la imagen en OpenStack

Para crear la imagen de la instancia de WebRTC realizamos el proceso anterior. Luego de haberse creado cada una de las imágenes, nos mostrara un listado de las imágenes existentes, tal y como se muestra en la figura 28.

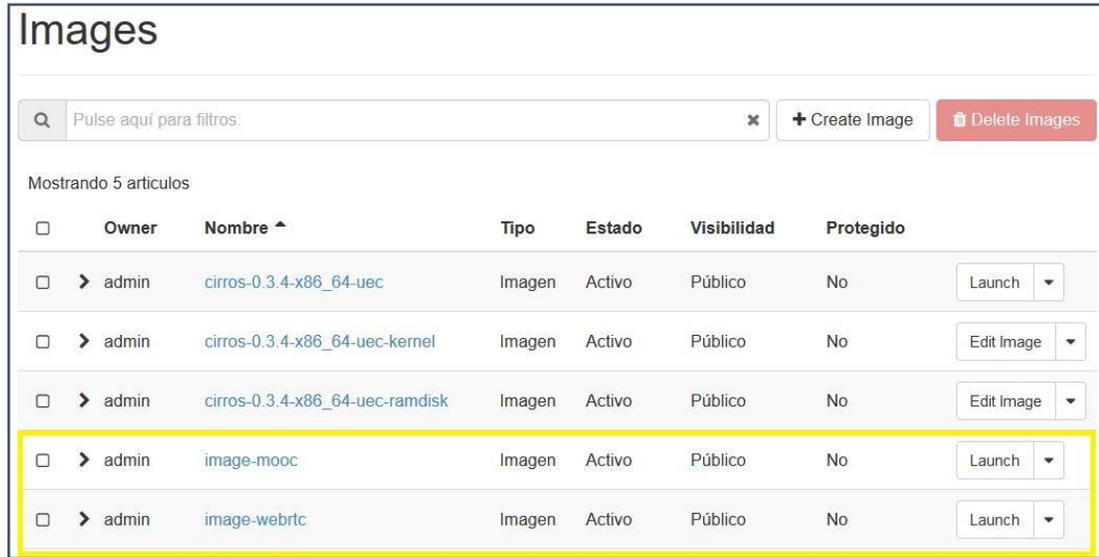


Figura 28. Listado de las imágenes existentes en OpenStack

### 9.3.7 CREACIÓN DE POLÍTICAS DE SEGURIDAD

La creación de las políticas de seguridad es muy importante para el funcionamiento correcto de los servicios y el acceso a las instancias, en el cual se restringe y se acepta puertos. Para acceder a las políticas de seguridad, nos ubicamos en el menú **Proyecto** → **Red** → **Grupos de seguridad** y luego creamos un nuevo grupo de seguridad seleccionando *Crear grupo de seguridad*, tal como se muestra en la figura 29.



Figura 29. Creación de grupos de seguridad en Openstack

A continuación, se muestra la ventana para crear un nuevo grupo de seguridad figura 30.

**Crear grupo de seguridad**

**Nombre \***  
Dentalav

**Descripción**  
Filtros para las instancias

**Descripción:**  
Los grupos de Seguridad son conjuntos de reglas de filtros de IP que son aplicadas a la configuración de red para la máquina virtual. Después de que el grupo de seguridad es creado, puedes agregar reglas al grupo de seguridad.

Cancelar    Crear grupo de seguridad

Figura 30. Parámetros para crear un de grupo de seguridad en Openstack

Luego de crear el grupo de seguridad procedemos a “Administrar reglas”. En este apartado se deberá agregar las reglas necesarias, a modo ejemplo se agrega una regla de ICMP para nuestra red local tal como se muestra en la figura 31.

**Agregar regla**

**Regla \***  
Todos los ICMP

**Dirección**  
Entrante

**Remoto \* ⓘ**  
CIDR

**CIDR ⓘ**  
0.0.0.0/0

**Descripción:**  
Las reglas definen el tráfico permitido a las instancias asociadas al grupo de seguridad. Una regla de un grupo de seguridad contiene tres partes principales:  
**Regla:** Puede especificar una plantilla de reglas deseada o usar reglas TCP, UDP e ICMP personalizadas.  
**Puerto abierto/Rango de puertos** Para las reglas de TCP y UDP puede optar por abrir un solo puerto o un rango de ellos. La opción "Rango de puertos" le proporcionará el espacio para especificar tanto el puerto de comienzo como de final del rango. Para las reglas de ICMP por el contrario debe especificar el tipo y código ICMP en los espacios proporcionados.  
**Remoto:** Debe especificar el origen del tráfico a permitir a través de esta regla. Lo puede hacer bien con el formato de un bloque de direcciones IP (CIDR) o especificando un grupo de origen (Grupo de Seguridad). Al seleccionar un grupo de seguridad como origen, se permitirá que cualquier instancia de ese grupo de seguridad pueda acceder a cualquier otra instancia a través de esta regla.

Cancelar    Añadir

Figura 31. Parámetros para agregar una nueva regla en OpenStack

Después de agregar las reglas necesarias para el correcto funcionamiento de la plataforma, se lista a continuación en la figura 32.

<input type="checkbox"/>	Dirección	Tipo Ethernet	Protocolo IP	Rango de puertos	Prefijo de IP Remota	Grupo de Seguridad Remoto	Acciones
<input type="checkbox"/>	Saliente	IPv4	Cualquier	Cualquier	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	ICMP	Cualquier	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	22 (SSH)	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	53 (DNS)	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	80 (HTTP)	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	443 (HTTPS)	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	1935	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	5060	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	5066	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	5070	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	10050	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Saliente	IPv4	TCP	10051	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	UDP	16384 - 32768	0.0.0.0/0	-	Eliminar Regla

Figura 32. Listado de políticas de seguridad, creadas en OpenStack

### 9.3.8 CREACIÓN DE LA INSTANCIA

Una vez creada las imágenes y las políticas de seguridad, lo siguiente es crear nuestra instancia MOOC, en este punto nos ubicaremos en las imágenes de OpenStack y seleccionamos la imagen “*image-mooc*” cargado anteriormente y le daremos a “*Launch*”.

<input type="checkbox"/>	>	admin	<a href="#">cirros-0.3.4-x86_64-uec-ramdisk</a>	Imagen	Activo	Público	No	Edit Image
<input type="checkbox"/>	>	admin	<a href="#">image-mooc</a>	Imagen	Activo	Público	No	Launch
<input type="checkbox"/>	>	admin	<a href="#">image-webrtc</a>	Imagen	Activo	Público	No	Launch

Figura 33. Opción para la creación de una instancia en OpenStack

Luego de ejecutar “*Launch*” debemos realizar una secuencia de pasos en los cuales se configura parámetros de asignación de red, disco y el grupo de políticas creada anteriormente. Los pasos a ejecutar se muestran a continuación.

En primer lugar, le asignamos un nombre a nuestra instancia, en nuestro caso “*srv-mooç*” figura 34.

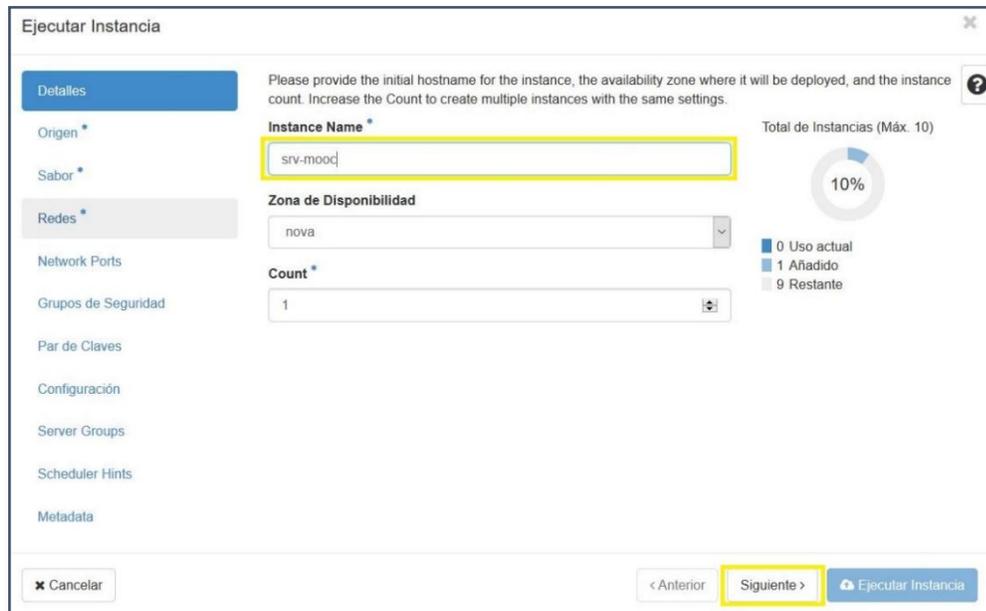


Figura 34. Asignación de nombre a la instancia en OpenStack

En el mismo “*Ejecutar instancia*” de la figura 34, lo siguiente es agregar nuestro **vdi** de *image-mooç*, como se muestra en la figura 35. Y continuamos.

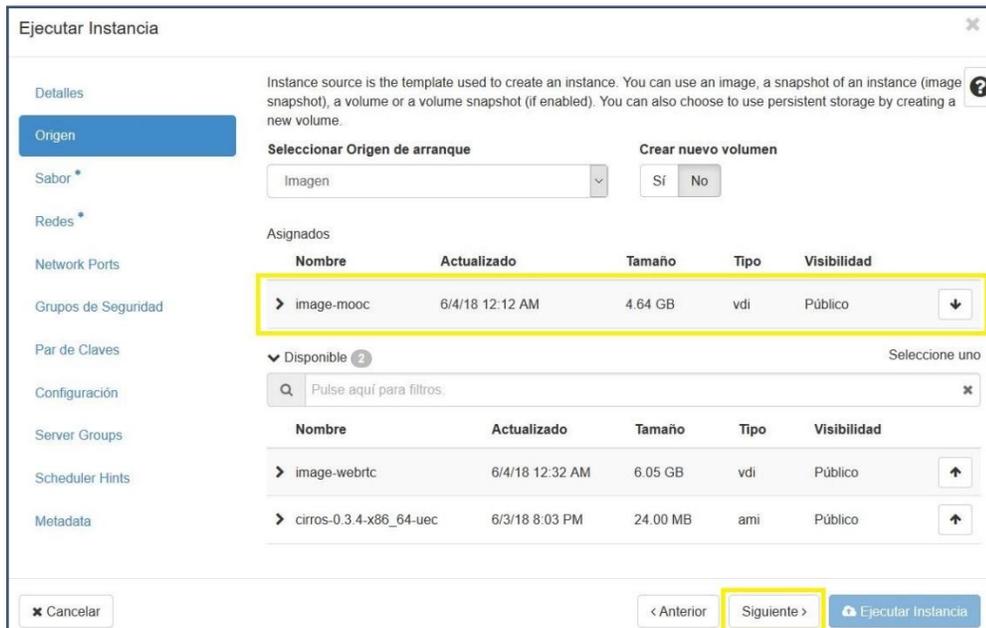


Figura 35. Asignación de la imagen vdi a la nueva instancia en OpenStack

Continuando con la creación de la instancia, en este punto le asignamos el sabor creado anteriormente, figura 25.

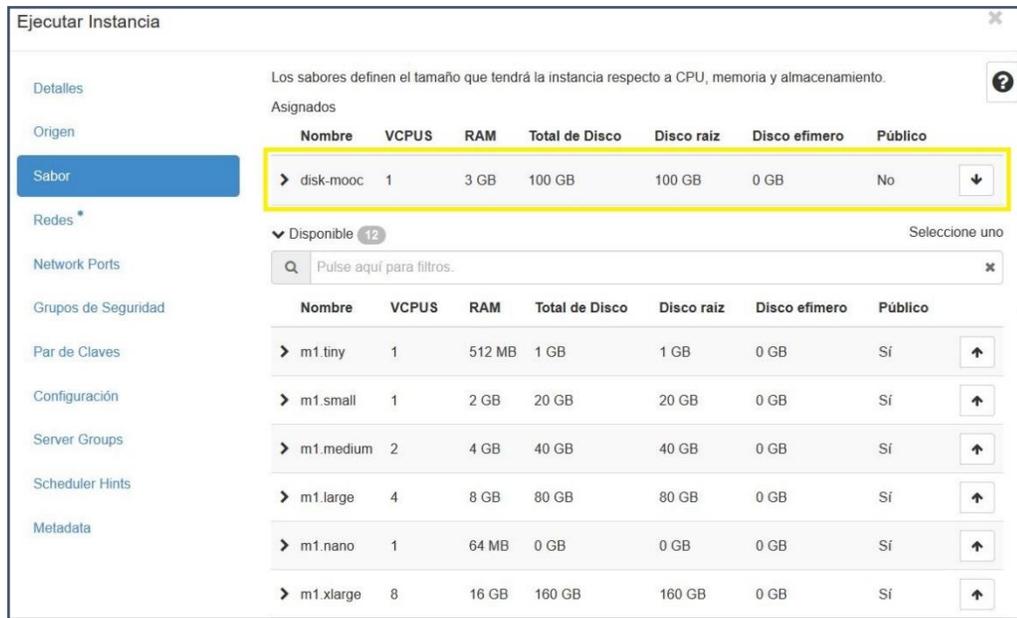


Figura 36. Selección del sabor, para la nueva instancia

Seleccionamos el tipo de red a la que se conectara nuestra instancia, en este caso nuestra red privada.

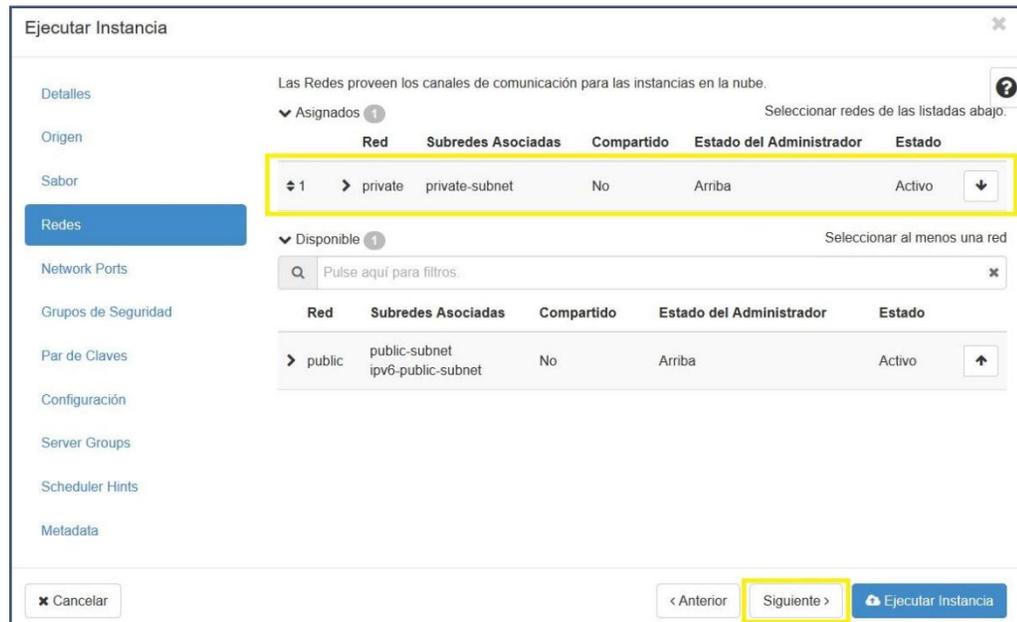


Figura 37. Configuración del tipo de red al que pertenecerá la instancia en OpenStack

En la sección de “*Grupos de Seguridad*”, que son las políticas de seguridad que configuramos anteriormente le agregamos a nuestra nueva instancia tal como se muestra en la figura 38. Y finalmente damos a **Ejecutar Instancia**, luego de varios minutos nuestra instancia será creada.

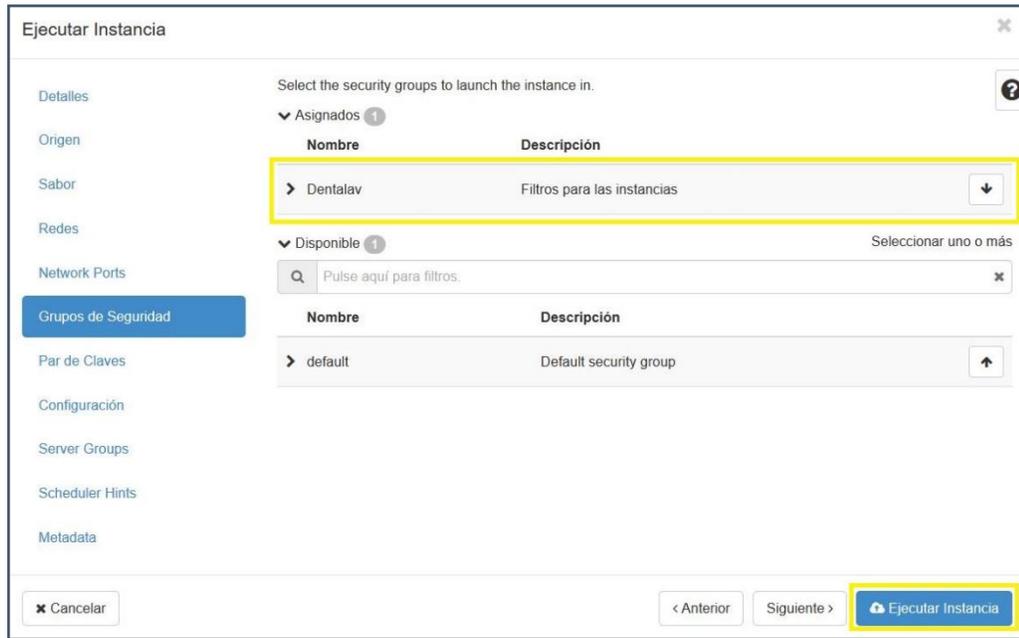


Figura 38. Configuración, políticas de seguridad para la instancia en OpenStack

Una vez creada y ejecutada la instancia, nos mostrara una ventana similar a la figura 39 en la cual se observa que todo está funcionando correctamente, y debemos fijarnos en la columna *Estado de energía*, la cual nos muestra, que nuestra instancia se está ejecutando.

<input type="checkbox"/>	Nombre de la instancia	Nombre de la imagen	Dirección IP	Sabor	Par de claves	Estado	Zona de Disponibilidad	Tarea	Estado de energía	Tiempo desde su creación
<input type="checkbox"/>	srv-mooc	image-mooc	192.168.5.14	disk-mooc	-	Activo	nova	Ninguno	Ejecutando	58 minutos

Figura 39. Listado de las instancias creadas en OpenStack

### 9.3.9 VERIFICACIÓN DE LA RED CREADA

Una vez finalizada todos los pasos anteriores; desde la creación del enrutador, red local, políticas de seguridad y nueva instancia. Podremos observar la topología creada a partir del Router Neutron y la instancia generada (MOOC). En la figura 40 se muestra la

topología de OpenStack en la que se visualiza el Enrutador, la instancia que está representada por una computadora conectada a la red privada y la red pública que se conecta directamente al enrutador. Para visualizar la topología, nos desplazamos en el menú **Proyecto → Red → Topología de red** y seleccionamos “Graph”.

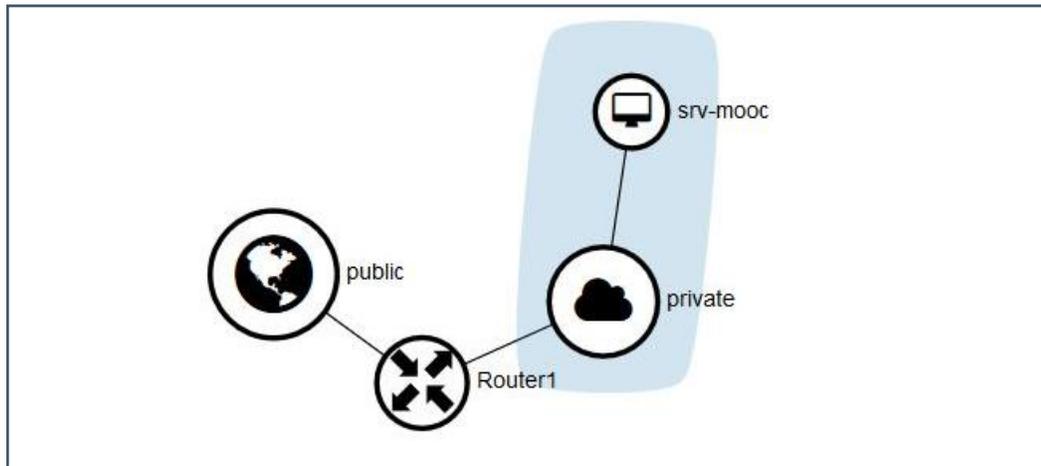


Figura 40. Topología de las instancias y Enrutador en OpenStack

Para visualizar la dirección IP del Router Neutron, el encargado de conectar los dispositivos de la red externa; abrimos el terminal e ingresamos el comando **ifconfig** y nos listara varias interfaces de los cuales nos interesa el “*br-ex*”.

```
br-ex  Link encap:Ethernet direcciónHW c6:1f:ae:5a:2c:45
      Direc. inet:172.24.4.1 Difus.:0.0.0.0 Másc:255.255.255.0
      Dirección inet6: 2001:db8::2/64 Alcance:Global
      Dirección inet6: fe80::c41f:aef:fe5a:2c45/64 Alcance:Enlace
      ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
      Paquetes RX:36 errores:0 perdidos:0 overruns:0 frame:0
      Paquetes TX:89 errores:0 perdidos:0 overruns:0 carrier:0
      colisiones:0 long.colaTX:1000
      Bytes RX:1716 (1.7 KB) TX bytes:11298 (11.2 KB)
```

Figura 41. Verificación de la interfaz br-ex perteneciente a Neutron

### 9.3.10 CREACIÓN DE RUTAS

Para establecer la conexión entre la red pública y privada previamente agregamos una ruta teniendo como destino la red privada a través del Gateway de **Router1 (Neutron)**, tal como se muestra en la figura 42.

```
# route add -net 192.168.5.0 netmask 255.255.255. gw 172.24.4.4
```

```
root@carper:/home/srv-cloudopenstack# route add -net 192.168.5.0 netmask 255.255.255.0 gw 172.24.4.4
```

Figura 42. Creación de ruta en Ubuntu

A continuación, listamos las rutas creadas en Ubuntu con el siguiente comando.

```
# route -n
```

```
root@carper:/home/srv-cloudopenstack# route -n
Tabla de rutas IP del núcleo
Destino          Pasarela          Genmask           Indic Métric Ref       Uso Interfaz
0.0.0.0          192.168.88.1     0.0.0.0          UG    100   0        0 ens33
169.254.0.0     0.0.0.0          255.255.0.0     U     1000  0        0 ens33
172.24.4.0      0.0.0.0          255.255.255.0   U     0     0        0 br-ex
192.168.5.0     172.24.4.4      255.255.255.0   UG    0     0        0 br-ex
192.168.88.0    0.0.0.0          255.255.255.0   U     100   0        0 ens33
192.168.122.0   0.0.0.0          255.255.255.0   U     0     0        0 virbr0
root@carper:/home/srv-cloudopenstack#
```

Figura 43. Tabla de enrutamiento en Ubuntu

## 9.4 INTEGRACIÓN SERVIDOR MOOC CON WEBRTC

Para realizar el proceso de integración de Moodle con BigBlueButton se debe usar un módulo llamado BigBlueButtonBN y RecordingsBN que se descarga de la página oficial de Moodle, para lo cual debemos ingresar en Administración del Sitio-Extensiones-Instalar Módulos externos.



Figura 44. Instalación de Módulos externos

### 9.4.1 INSTALACIÓN DE MÓDULOS

Tenemos dos formas de instalar el modulo desde el directorio de extensiones de Moodle o descargarnos el archivo zip e importarlo. En este caso optamos por la primera opción.



Figura 45. Instalación del módulo desde el directorio de extensiones de Moodle

Posteriormente accedemos al sitio de Moodle con una cuenta de Google, Microsoft o Facebook. Nos dirigimos a Downloads, Extend Moodle y seleccionamos la opción *Moodle Plugins Directory*.

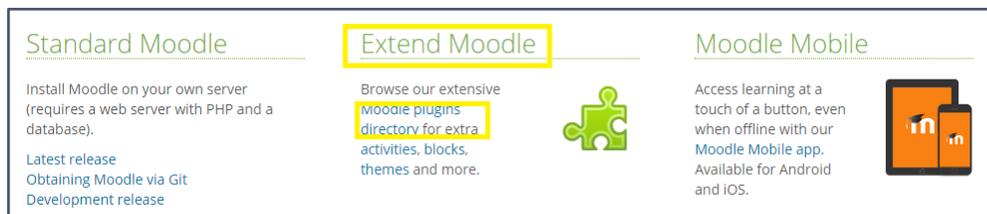


Figura 46. Página Oficial de Moodle

Buscamos e Instalamos los dos módulos necesarios para la integración de Moodle con BigBlueButton. El primer módulo *BigBlueButtonBN* permite crear enlaces dentro del sitio MOOC hacia los salones de clases en tiempo real o en línea. El segundo módulo

*RecordingsBN* se encarga de la administración de las grabaciones realizadas en las videoconferencias, los mismos que estarán disponible en sitio MOOC para que los estudiantes lo pueden visualizar cuantas veces desee.

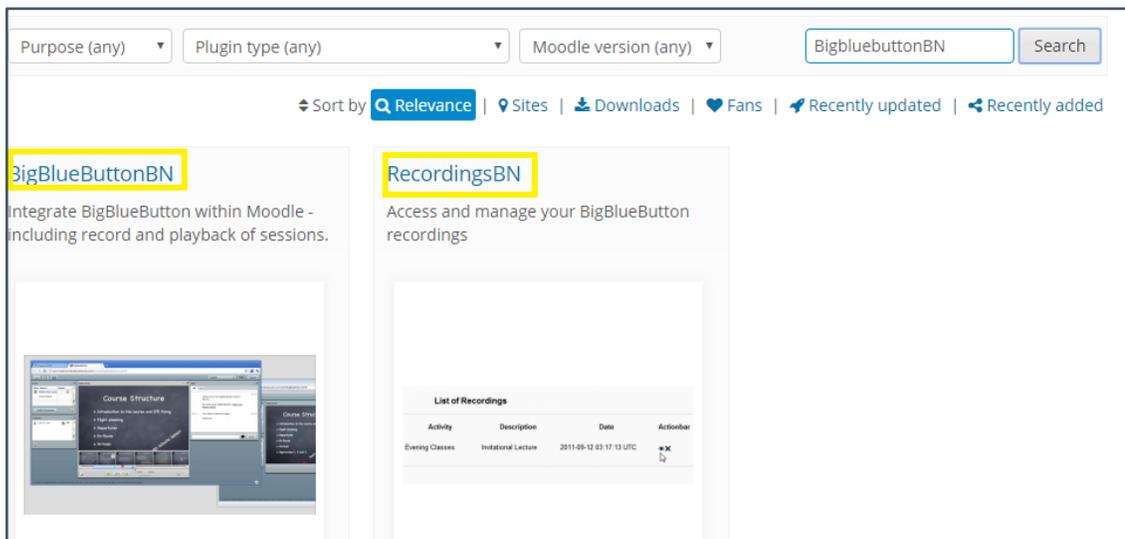


Figura 47. Módulos necesarios para sincronizar MOOC con WebRTC

Y procedemos a instalar, seguidamente nos informara si los módulos seleccionados son o no compatibles con la versión de Moodle y finalmente actualizamos la base de datos, tal como se muestra en la figura 48.



Figura 48. Actualización de la base de datos MOOC para instalar el módulo BigBlueButtonBN

## 9.4.2 COPIAR URL Y CLAVE SECRETA AL SERVIDOR MOOC

Para terminar la integración de Moodle y BigBlueButton necesitamos la URL del Servidor BigBlueButton y la clave secreta, para lo cual en el servidor BigBlueButton corremos el siguiente comando.

```
# bbb-conf --secret
```

El resultado de ejecutar el comando anterior lo copiamos a nuestro sitio MOOC para terminar con la sincronización, tal como se muestra en la figura 49.

URL del servidor BigBlueButton <small>bigbluebuttonbn_server_url</small>	<input type="text" value="https://my.edacamo.com/bigbluebutton"/>	Valor por defecto: <a href="http://test-install.blindsidenetworks.com/bigbluebutton/">http://test-install.blindsidenetworks.com/bigbluebutton/</a>
La URL de su servidor BigBlueButton debe terminar en /bigbluebutton/. (Esta URL por defecto es para un servidor BigBlueButton proporcionado por Blindside Networks que Usted puede usar para pruebas.)		
Clave secreta de BigBlueButton <small>bigbluebuttonbn_shared_secret</small>	<input type="text" value="74637ea7b657d86e9d89730dbbc86d2"/>	Valor por defecto: <a href="#">8cd8ef52e8e101574e400365b55e11a6</a>

Figura 49. URL y clave secreta del Servidor BigBlueButton

## 9.5 CONFIGURACIÓN DEL ENTORNO DE DESARROLLO PARA LA APLICACIÓN MÓVIL

Para configurar el ambiente de desarrollo de la aplicación, para la plataforma MOOC se tomó como base la aplicación Moodle Mobile ya que la misma cuenta con una versión para desarrolladores mediante la cual podemos adaptar y desarrollar de acuerdo a las necesidades especificadas.

El ambiente de desarrollo está instalado en el Sistema Operativo Windows 10, se debe tener en cuenta que se puede instalar en cualquier otro sistema Operativo.

### 9.5.1 INSTALAR GIT

Git es un software de control de versiones de código abierto ofreciendo eficiencia y confiabilidad ya que proporciona herramientas para desarrollar trabajos en equipo de una manera inteligente y rápida. Git está diseñado para trabajar desde proyectos pequeños a proyectos muy grandes. Para instalar Git debemos descargar desde su página oficial dependiendo de la versión del Sistema Operativo que tengamos (ya sea 32 o 64 bits).

### 9.5.2 INSTALAR NODE.JS

Node.js es un entorno de código abierto para JavaScript creado por Google V8. Utilizando Google V8 se proporciona un entorno de ejecución que permite ejecutar y compilar código JavaScript en el lado del servidor. Mediante Node se garantiza que el servidor soporte decenas de miles de conexiones concurrentes. Así mismo se lo obtiene desde su página oficial. Es preferible descargar una versión estable para no tener problemas de compatibilidad.

### 9.5.3 INSTALAR IONIC

Ionic es un Framework de código abierto utilizado para el desarrollo de aplicaciones híbridas (Móviles, Web) multiplataforma, en conjunto con AngularJS con su versatilidad y potencia facilitara la creación de aplicaciones. Ionic utiliza Html5, JS y CSS además de integrar Cordova como base.

El siguiente comando borra todos los datos de la carpeta de cache

```
npm cache clean
```

Instalamos el Framework Ionic además de Cordova mediante el cual vamos a compilar nuestro código Html5, CSS3 para el desarrollo multiplataforma, en nuestro caso solo compilaremos nuestra aplicación para la plataforma Android.

```
npm install -g cordova ionic
```

### 9.5.4 INSTALAR PAQUETES REQUERIDOS

En esta sección debemos instalar la herramienta Gulp mediante la cual vamos a automatizar las tareas en proyectos JavaScript minimizando el código, recarga del navegador, compresión de las imágenes, entre otros.

```
sudo npm install -g gulp
```

#### Instalar dependencia para Windows

En Windows se necesita instalar dependencias de compilación nativa.

```
npm install --global --production windows-build-tools
```

### Clonar el código de la aplicación

Ejecutamos el siguiente en un directorio raíz en nuestra computadora donde estará alojado la aplicación, cabe recalcar que en esta carpeta se encontrará todos los archivos necesarios para que la aplicación funcione correctamente y en donde realizaremos nuestros respectivos cambios.

```
git clone https://github.com/moodlehq/moodlemobile2.git moodle
```

Accedemos a la carpeta moodle

```
cd moodle
```

Mediante el siguiente comando actualizamos los directorios en el archivo de trabajo para que coincida con la versión almacenada en esa rama y git registra las nuevas confirmaciones en esa rama.

```
git checkout v3.5.0
```

### Configuramos el ambiente

Ejecutamos el siguiente comando en la carpeta de Moodle.

```
npm run setup
```

#### 9.5.5 EJECUTAR LA APLICACIÓN

Para ejecutar la aplicación escribimos el siguiente comando en la ruta donde está ubicada nuestra aplicación.

```
ionic serve
```

Al correr el comando anterior podemos ver en la siguiente Figura 50, que nuestra aplicación se inició correctamente.

```
Running live reload server: http://192.168.140.1:35729
Watching: www/**/*.*.html www/build/**/*.*.html, !www/lib/**/*.*
✓ Running dev server: http://192.168.140.1:8100
Ionic server commands, enter:
restart or r to restart the client app from the root
goto or g and a url to have the app navigate to the given url
consolelogs or c to enable/disable console log output
serverlogs or s to enable/disable server log output
quit or q to shutdown the server and exit

ionic $
```

Figura 50. Inicio de la App en Ionic

## 9.5.6 CONFIGURAR APLICACIÓN

Para acceder solo a nuestra plataforma mediante la aplicación debemos configurar el siguiente archivo ubicado en la carpeta www.

```
config.json
1 {
2   "app_id": "com.moodle.moodlemobile",
3   "appname": "Dentalav",
4   "desktopappname": "Moodle Desktop",
5   "versioncode": "2021",
6   "versionname": "3.4.0",
7   "cache_expiration_time": 300000,
8   "default_lang": "en",
9   "languages": {"ar": "العربية", "bg": "Български", "ca": "Català", "cs": "Čeština", "da": "Dansk", "de": "Deutsch", "de-du": "Deutsch - Du", "el": "Ελληνικά", "en":
10  "English", "es": "Español", "fr": "Français", "he": "עברית", "id": "Bahasa Indonesia", "it": "Italiano", "ja": "日本語", "ko": "한국어", "lt": "Lietuvių", "lv": "Latviešu", "nl": "Nederlands", "no": "Norsk", "pl": "Polski", "pt": "Português", "pt-br": "Português (Brasil)", "ro": "Română", "ru": "Русский", "sk": "Slovenčina", "sl": "Slovenski", "sv": "Svenska", "th": "ไทย", "tr": "Türkçe", "uk": "Українська", "vi": "Tiếng Việt", "zh": "中文", "zh-cn": "中文 (中国大陆)", "zh-tw": "中文 (台湾)"
11  "wsservice": "moodle_mobile_app",
12  "wsextservice": "local_mobile",
13  "demo_sites": {"student": {"url": "https://school.demo.moodle.net", "username": "student", "password": "moodle"}, "teacher": {"url": "https://school.demo.moodle
14  "gcmpn": "694767596569",
15  "customurlscheme": "moodlemobile",
16  "siteurl": "https://www.dentalav.org/cursos",
17  "newsitesprivacy": "strict",
18  "skipsoconfirmation": "false",
19  "forcedefaultlanguage": "false",
20  "privacypolicy": "https://moodle.org/mod/page/view.php?id=8148"
21 }
```

Figura 51. Configuración de URL en la App con el servidor MOOC

Al modificar imágenes, iconos, entre otros componentes de nuestra App, se recomienda ejecutar el comando indicado a continuación para que la aplicación se adapte a todas las resoluciones de los dispositivos.

*ionic resources*

Para compilar la aplicación en apk para nuestros dispositivos Android debemos ejecutar el siguiente comando, tener en cuenta que se debe tener instalado Android Studio que es uno de los requisitos de Cordova además del JDK de Java.

*cordova build*

Actualizar Ionic y Cordova, en caso de incompatibilidad de versiones

Para actualizar automáticamente Ionic y las plataformas de Cordova ejecutamos el siguiente comando.

### Ionic

```
sudo npm update -g cordova
```

### Cordova

```
sudo npm update -g ionic
```

## 9.6 MIGRACIÓN HACIA CLOUD PÚBLICO MICROSOFT AZURE Y FUNCIONAMIENTO

El proceso de migración hacia el Cloud público de Microsoft Azure, se efectúa luego de ejecutar las pruebas sobre el cloud Openstack (Capítulo 10) y definir las métricas de Memoria RAM, Procesamiento (CPU) y de ancho de banda necesarios para el correcto funcionamiento de la plataforma MOOC como el de WebRTC, para un determinado número de usuarios. En esta ocasión el servicio a desplegar será para 25 usuarios respecto al servidor de videoconferencia, en el caso del MOOC la cantidad de usuarios que soportara por 1 GB de memoria RAM es aproximadamente de 30 usuarios.

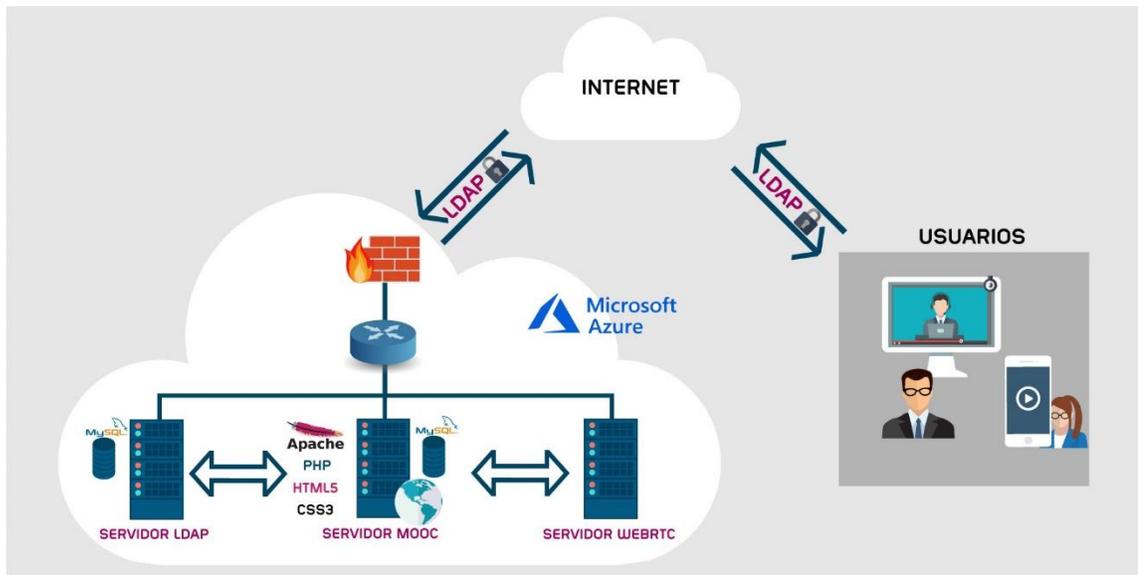


Figura 52. Esquema de funcionamiento de la plataforma alojada en la nube pública Microsoft Azure

## 9.6.1 ACCESO A LA PLATAFORMA

Luego de haber realizado la migración e integración de los servicios de MOOC y WebRTC a la nube pública de Microsoft Azure se procede exponer las principales funcionalidades del sistema de teleeducación, empezando por el acceso a la página informativa, figura 53.

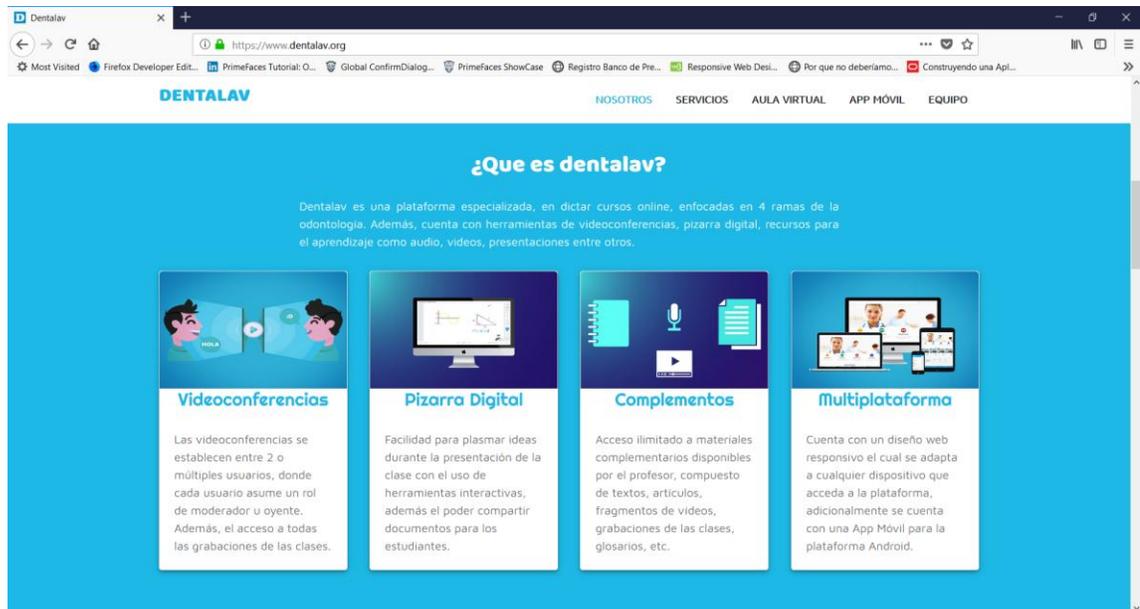
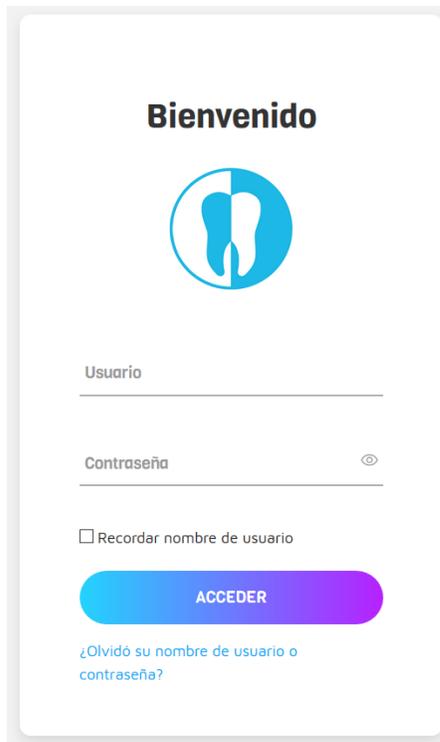


Figura 53. Acceso a la página informativa, DENTALAV

Una vez ingresado al sitio web accedemos al ambiente virtual como usuario admin para crear categorías, cursos, salas de videoconferencia y la matriculación de usuarios.



The image shows a login form titled "Bienvenido" (Welcome). At the top center is a circular logo containing a stylized tooth. Below the logo are two input fields: "Usuario" (Username) and "Contraseña" (Password). The password field has a small eye icon to its right. Underneath the password field is a checkbox labeled "Recordar nombre de usuario" (Remember username). A prominent, rounded rectangular button with a blue-to-purple gradient is labeled "ACCEDER" (Log In). At the bottom of the form, there is a link that reads "¿Olvidó su nombre de usuario o contraseña?" (Forgot your username or password?).

Figura 54. Acceso de usuarios a la plataforma MOOC

### 9.6.2 CREACIÓN DE USUARIOS

Ya dentro del sitio MOOC, se tiene el módulo de creación de usuarios, para lo cual en el menú nos ubicamos en *Administración del sitio* → *Usuarios* → *Agregar un usuario* y procedemos a llenar el formulario con los datos de nuevo usuarios como se muestra en la figura 55.

Nombre de usuario

Escoger un método de identificación:

Cuenta de usuario suspendida

Generar contraseña y notificar al usuario

La contraseña debería tener al menos 8 caracter(es), al menos 1 dígito(s), al menos 1 minúscula(s), al menos 1 mayúscula(s), al menos 1 caracter(es) no alfanuméricos como \*, -, o #

Nueva contraseña

Forzar cambio de contraseña

Nombre

Apellido(s)

Dirección de correo

Figura 55. Formulario para registrar un usuario

### 9.6.3 CREACIÓN DE CATEGORÍAS

Previa a la creación de cursos es importante crear una categoría con la finalidad de categorizar los cursos de nuestro MOOC y para su creación dentro del menú *Administración del sitio* → *Cursos* → *Añadir categoría* y los datos de la nueva categoría tal como se muestra en la figura 56. Cabe recalcar que se puede crear una categoría padre o una subcategoría.

Agrega otra categoría

Categoría padre

Nombre de la categoría

Número ID de la categoría

Descripción

Figura 56. Formulario para registrar una categoría

#### 9.6.4 CREAR CURSO Y ASIGNAR A UNA CATEGORÍA

Después de crear la categoría procedemos a crear un curso y le asignamos la categoría creada además de un nombre, fecha de inicio y finalización entre otros parámetros que se muestra en la figura 57.

The screenshot shows a web form titled "Agrega otro curso" with a "General" section. The form contains the following fields and options:

- Nombre completo del curso:** Text input field containing "Control de infecciones".
- Nombre corto del curso:** Text input field containing "CDI".
- Categoría de cursos:** Dropdown menu with "Endoncia" selected.
- Visibilidad del curso:** Dropdown menu with "Mostrar" selected.
- Fecha de inicio del curso:** Date picker showing 5, March, 2018.
- Fecha de finalización del curso:** Date picker showing 7, August, 2018, with a checked "Habilitar" checkbox.
- Número ID del curso:** Empty text input field.

Figura 57. Formulario para registrar un curso

#### 9.6.5 MÓDULO DE MATRICULACIÓN USUARIOS

En este apartado se procede a la matriculación de usuarios el mismo que se deberá ingresar al curso, en la sección **Usuarios matriculados**, de los cuales los roles a asignar son Rol de profesor, estudiante, gestor y profesor sin permiso de edición tal como se puede observar en la figura 58. Los roles disponibles serán explicados más adelante.

Matricular usuarios

Opciones de matriculación

Select users

× Juan Torres juan@dentalav.com

j

Asignar rol

Ver más...

Estudiante

Gestor

Profesor

Profesor sin permiso de edición

Estudiante

Matricular usuarios

Cancelar

Figura 58. Formulario de matriculación de usuarios

A continuación, se detalla los diferentes roles que maneja el sistema MOOC.

### Rol de Manager

*Manager* (gestor) puede acceder a los diferentes cursos y modificarlos, además de realizar ciertos trabajos a nivel administrativo relacionados con cursos, usuarios, configuración de calificaciones entre otras, siendo un rol seguro de usar.

### Rol de Profesor

El profesor tiene las funciones de añadir o cambiar el contenido que se dictaran en el curso o los cursos que le hayan designado, matricular usuarios y calificar a los estudiantes. Así mismo el de poder crear las diferentes videoconferencias que se dictaran durante el periodo de la clase adaptando a su conveniencia acorde al calendario de planificación que el docente llevara.

### Rol de profesor no-editor

Solo se le permite ver y calificar los trabajos de los estudiantes en un curso con la restricción de alterar o borrar las actividades o recursos, este rol por ejemplo se da al ayudante del profesor.

### Rol de Estudiante

Un usuario con el rol de estudiante puede participar en las diferentes actividades del curso o cursos que se encuentra matriculado y ver los diferentes recursos como (archivos, audio, videos y videoconferencias) todo referente para su preparación académica. Una funcionalidad extra que tienen los estudiantes es el de poder corregir o calificar el trabajo de otros estudiantes con el objetivo de interactuar con el resto de participantes siempre y cuando tenga los permisos otorgados por parte del profesor.

### 9.6.6 CREACIÓN DE UNA SESIÓN (VIDEOCONFERENCIA)

En esta sección, se indica el proceso de la creación de una sesión dentro de un curso virtual, la cual estará disponible acorde a la fecha establecida por el docente. Para crear la sesión ingresamos con el rol de Profesor y habilitamos la edición del curso, tal como se muestra en la figura 59.

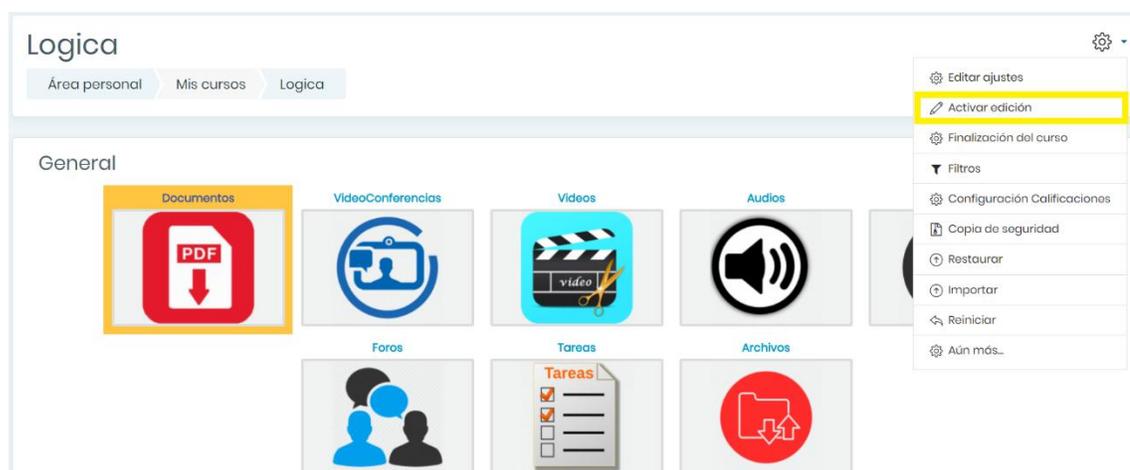


Figura 59. Habilitar la edición del curso

Una vez activada la edición del curso nos ubicamos en la sección de **VideoConferencias** y añadimos una nueva actividad.



Figura 60. Creación de una nueva sesión (videoconferencia)

Al seleccionar la opción agregar nueva actividad se nos desplegara una ventana emergente similar a la figura 61, en el que se muestran las diferentes actividades que podemos crear en el curso, en esta ocasión debemos seleccionar el modulo que agregamos anteriormente en la sección de la sincronización del MOOC con WebRTC.

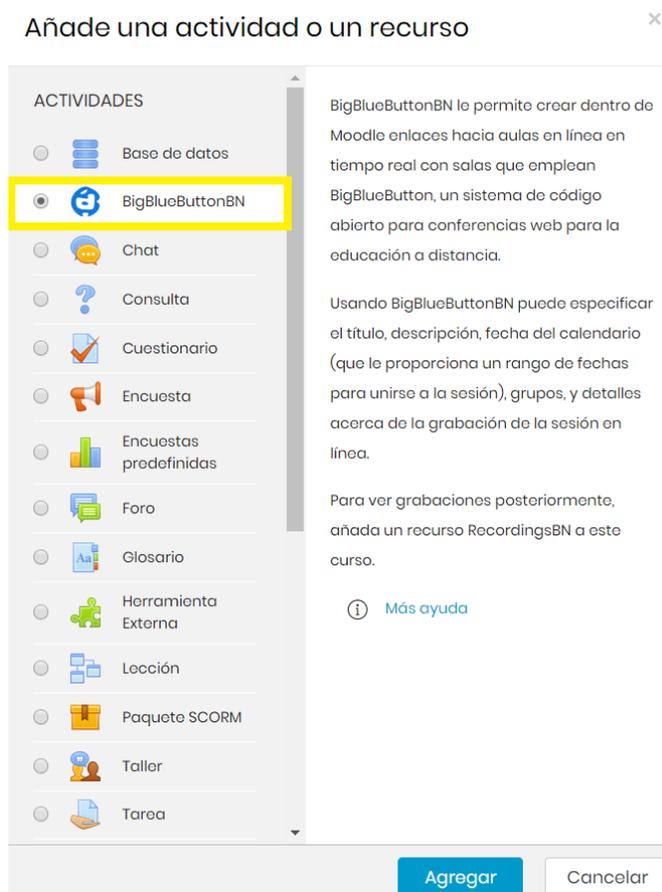


Figura 61. Ventana emergente para crear actividades

Posteriormente nos muestra una pantalla (figura 62) en la que se procede a programar la videoconferencia, donde los parámetros a llenar son el nombre, tipo de sesión (sala con opción a grabar la videoconferencia o solo sala de videoconferencia), restricción de la sesión, rol de los participantes, fecha disponible entre otras opciones.

En esta ocasión los parámetros a configurar son: el nombre de la sesión con opción de grabar la videoconferencia, enviar notificación a los estudiantes del curso, restricción de esperar al moderador para ingresar a la videoconferencia, el rol de los participantes

y la fecha de inicio y finalización. A continuación, se detallan cada una de las especificaciones antes mencionada (figura 62 – figura 65).

Instance type ? Room/Activity with recordings ⇅

▼ **Ajustes generales**

Nombre de la sala virtual ! SesionPrueba

Enviar notificación ?

Figura 62. Parámetros de nueva sesión (videoconferencia), asignación de nombre

▼ **Activity/Room settings**

Mensaje de bienvenida ?

Bienvenidos a la clase de introducción

Esperar al moderador ?

La sesión puede ser grabada

Figura 63. Restricción de la sesión (esperar al moderador y opción de grabar la videoconferencia)

▼ **Participantes**

Agregar participante Todos los usuarios inscritos ⇅ ----- ⇅ Agregar

Lista de participantes

<b>Todos los usuarios inscritos</b>	como	Observador ⇅	🗑️
<b>Usuario:</b> Jack Bravo	como	Moderador ⇅	🗑️

Figura 64. Asignación de roles en la nueva sesión

▼ **Programación de sesiones**

Acceso abierto 19 ⇅ June ⇅ 2018 ⇅ 23 ⇅ 30 ⇅ 🗓️  Habilitar

Acceso cerrado 20 ⇅ July ⇅ 2018 ⇅ 23 ⇅ 30 ⇅ 🗓️  Habilitar

Figura 65. Programación de fecha para la nueva sesión

### 9.6.7 ESTABLECIENDO VIDEOCONFERENCIA

Una vez terminada la creación de las videoconferencias en las diversas materias se les enviara un mensaje notificando la creación de la misma en la cual se indica la fecha y hora para el acceso. Y para su acceso todo depende de la configuración que el docente realice donde configura si los estudiantes esperan al moderador o tienen libre acceso. A continuación, se presenta un ejemplo donde se estableció una videoconferencia entre moderado y participantes.



*Figura 66. Sesión de videoconferencia entre moderador y participantes (estudiantes)*

A continuación, se expone una sesión de videoconferencia utilizando las herramientas interactivas para una comunicación real y directa entre moderador y participantes; los cuales son chat público y privado, pizarra digital, notas compartidas y compartir documentos. Adicionalmente la restricción del uso de la cámara para mejorar la calidad de servicio durante la videoconferencia.

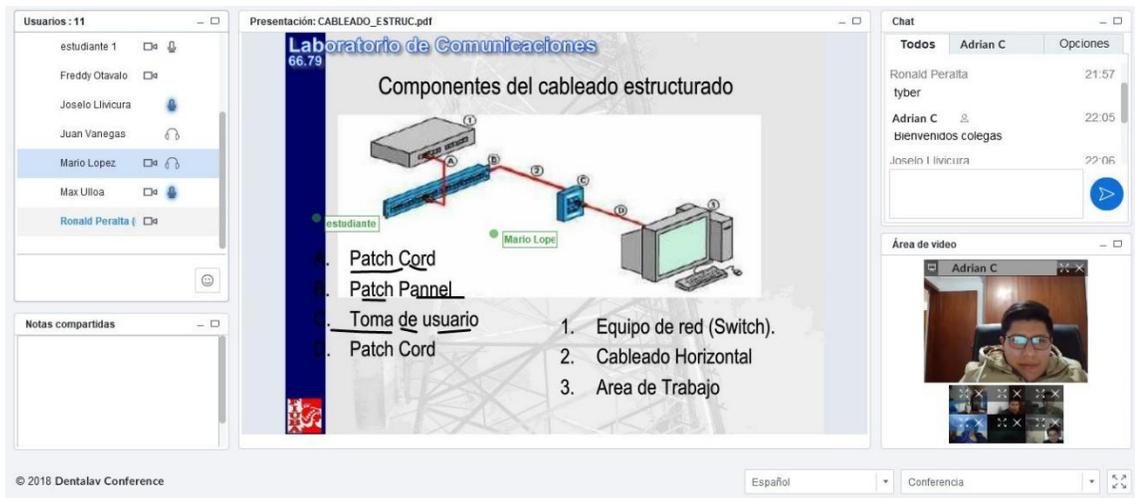
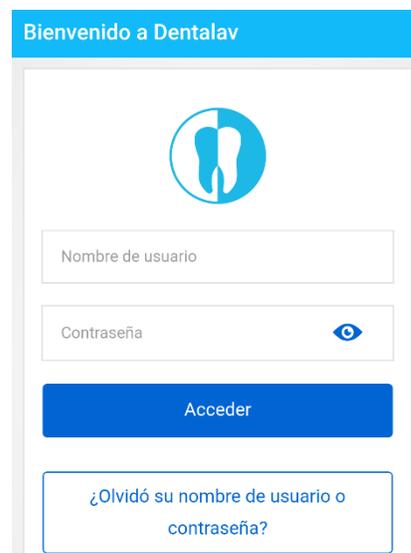


Figura 67. Presentación de clase, compartiendo documentos

### 9.6.8 APLICACIÓN MÓVIL

En cuanto a la aplicación móvil, para el presente proyecto solo está disponible para la plataforma Android y a continuación se muestra sus principales funcionalidades que en su mayoría cumple con las mismas funcionalidades que la plataforma web con la única diferencia que el acceso es a través de un Smartphone.

Así mismo tenemos una ventana de acceso de usuarios los mismos que pueden ser estudiante o profesor. A continuación, se muestra el acceso de un usuario con el rol de



estudiante, figura 68.

Figura 68. Acceso a la Aplicación Móvil

Una vez verificado las credenciales del usuario y concedido el acceso, en primera instancia nos presenta el listado de los cursos en el que se encuentra inscrito, además del porcentaje de progreso que se tiene completado el curso.

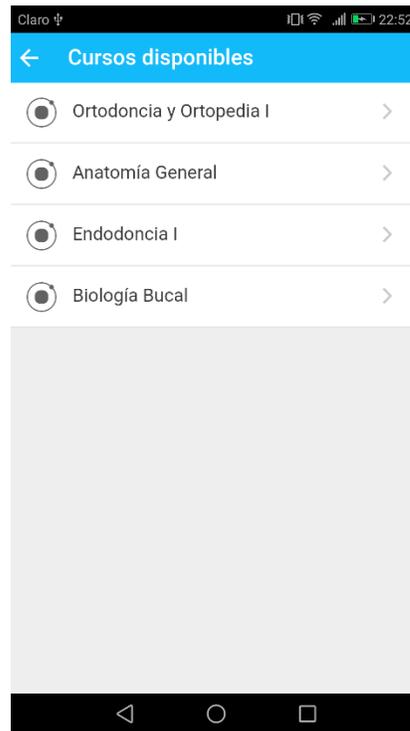


Figura 69. Vista general de cursos

A continuación, se presenta el menú principal de la aplicación (figura 70) en el que se tiene las siguientes opciones:

- Vista general del Curso
- Inicio del Sitio
- Calificaciones
- Avisos y/o notificaciones
- Mensajes
- Calendario
- Archivos

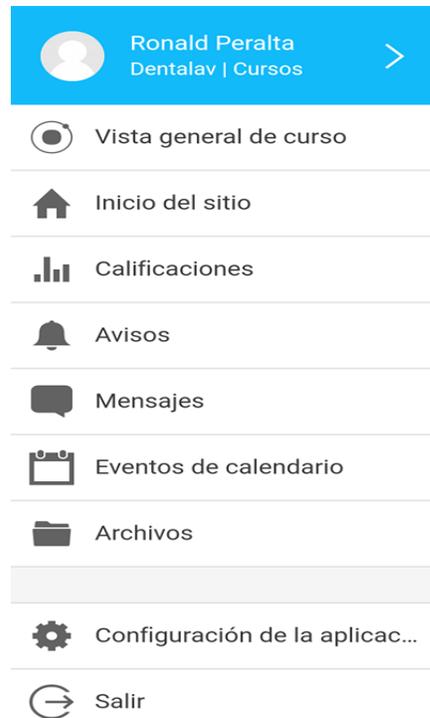


Figura 70. Menú principal

## Mensajes

Por lo que se refiere a mensajes, en esta sección se mostrara todas las actividades a realizarse en el curso; desde la creación de un trabajo hasta una sesión de videoconferencia.

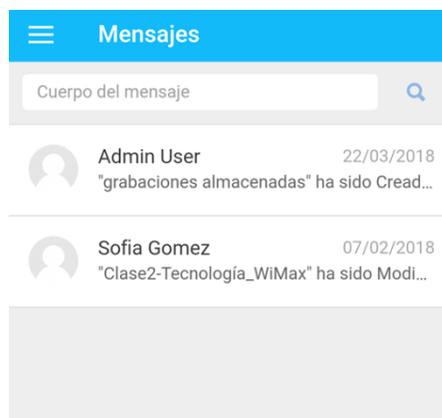


Figura 71. Mensaje de actividades en el curso

## Calificaciones

En este apartado se podrá visualizar todas las calificaciones de trabajos, pruebas, entre otros.

← Calificaciones	
Nombre del ítem	Calificación
<b>Logica</b>	
✓ Prueba 1	20,00
Σ Total del curso	<b>20,00</b>

Figura 72. Calificaciones aplicación móvil

### 9.6.9 CLIENTE HTML5 (OPCIÓN DE WEBRTC EN SMARTPHONE)

El cliente Html5 es más rápido y confiable; muy útil para dispositivos móviles ya que los mismos no soportan Flash a diferencia de los ordenadores. Al ingresar con el cliente html5 lo primero que nos pedirá es escoger la opción de micrófono o solo escuchar.

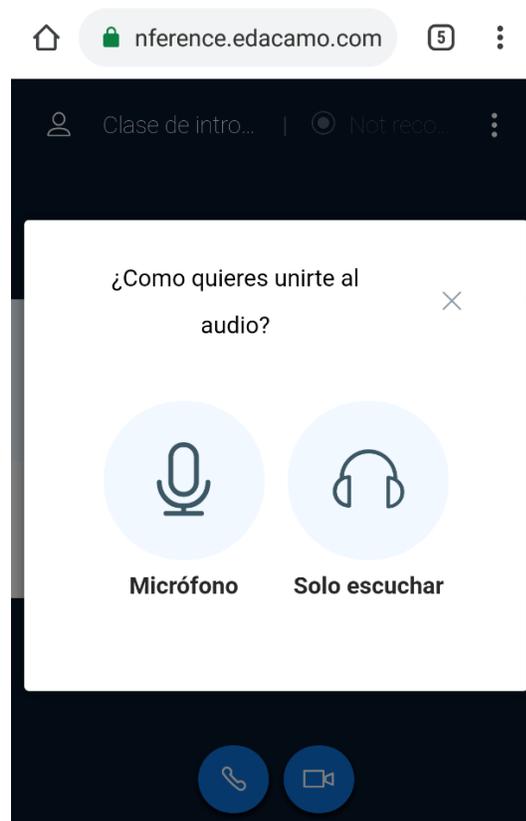


Figura 73. Cliente Html5

Otras de las funcionalidades que ofrece el cliente html5 son la de establecer un chat privado y público durante la videoconferencia.

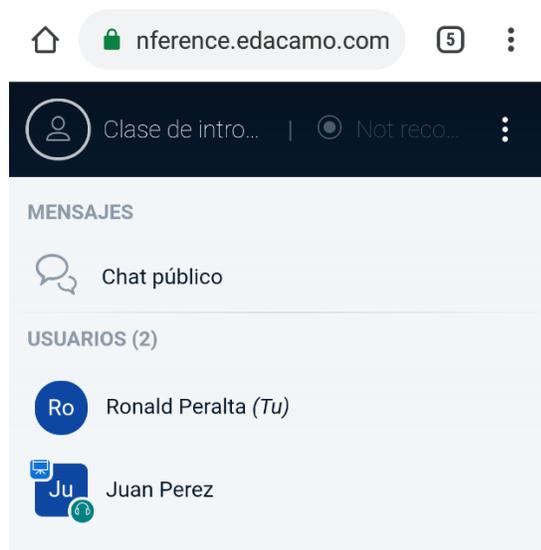


Figura 74. Chat Cliente html5

Al utilizar el chat público todos los usuarios conectados en ese momento podrán ver los mensajes además de que también ellos puedan interactuar.



Figura 75. Chat público, cliente html5

También se ofrece la opción de chat privado mediante el cual se puede enviar mensajes solo a una persona determinada sin que los demás participantes lo vean.

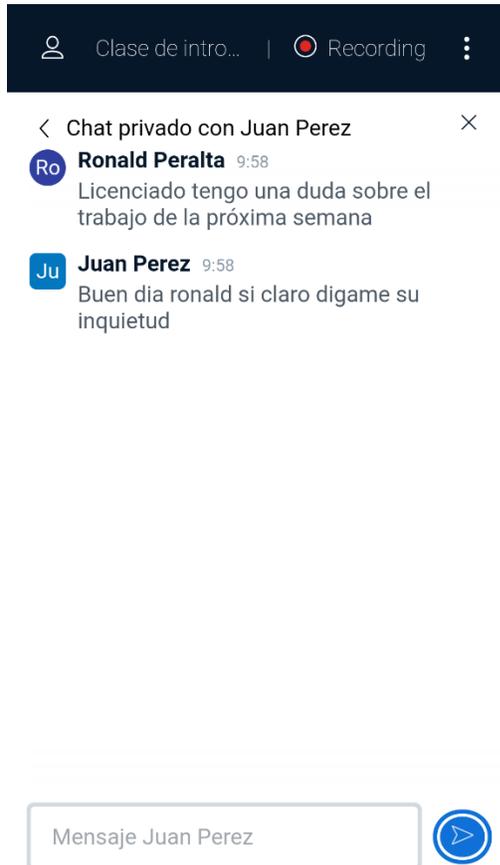


Figura 76. Chat privado, cliente html5

## 10 PRUEBAS Y RESULTADOS

La ejecución de las pruebas tiene como objetivo establecer las métricas adecuadas sobre el uso de la plataforma MOOC, las cuales nos permitan evaluar el correcto funcionamiento con una cierta cantidad de usuarios concurrentes, sin perder la funcionalidad de la plataforma. Las pruebas a ejecutar se consolidan en el rendimiento de memoria, procesamiento y red, el cual se ejecutará sobre los servidores WEB y el Sistema de video conferencia BBB (BigBlueButton) evaluando medidas significativas como son: latencia, jitter, ancho de banda y concurrencia.

## 10.1 PRUEBAS DEL SERVIDOR WEB (MOOC)

Para evaluar el rendimiento de un servidor y determinar las características de hardware necesarios para desplegar la aplicación web (MOOC), y que a su vez este brinde su total funcionalidad a un determinado número de usuarios. Existen diversas métricas que se tienen que medir para garantizar el correcto funcionamiento de nuestro servidor de manera que nos enfocaremos en la evaluación de los recursos de; memoria RAM, ancho de banda y el uso del CPU.

La medición de estas métricas se determinará con el uso de dos herramientas; la primera Apache JMeter que es una herramienta de código abierto desarrollada en java, mayormente utilizada para medir y analizar el rendimiento de diferentes aplicaciones, servidores y tipos de protocolos; la segunda es Zabbix otra herramienta de código abierto considerado como un sistema de Monitorización de Redes; diseñada explícitamente para el monitoreo y recopilación de métricas como (CPU, Memoria, Ancho de banda entre otros) todo esto sobre servicios de red, servidores y hardware de red.

Las características más relevantes que nos proporciona al usar esta herramienta son:

- ✓ Recolección de métricas desde cualquier dispositivo, sistema o aplicación a través de agentes SNMP e IPMI, scripts personalizados y módulos.
- ✓ Envío de notificaciones personalizadas al administrador mediante correo electrónico o mensajes de texto en caso de surgir anomalías en nuestra infraestructura de red.
- ✓ Seguridad y protección en todos los niveles
- ✓ Uso de plantillas predefinidas para mayor agilidad de monitoreo.

Una vez definida las herramientas a usarse y las métricas a evaluar, se estableció el plan de pruebas que está dividido en dos partes; la primera es la creación de los cursos con sus respectivas actividades y recursos, segundo es la concurrencia de usuarios navegando en los diferentes módulos que posee la plataforma MOOC, seguidamente

se detallan los módulos con su respectivo método de solicitud GET (obtiene información del servidor) y POST (envió de información desde el cliente).

- ✓ Inicio de sesión a la plataforma (Login) con tipo de solicitud GET.
- ✓ Revisar el curso con sus diferentes actividades y recursos, recuperando la información con solicitud GET.
- ✓ Revisar y comentar foros los cuales realizan peticiones al servidor de tipo GET y POST respectivamente.
- ✓ Visualizar el listado de los participantes matriculados en el curso, ejecutando el tipo solicitud GET.
- ✓ Finalmente, el módulo de cerrar sesión (Logout) con tipo de solicitud GET.

### 10.1.1 CREACIÓN DE CURSOS

La ejecución de este plan de pruebas, es el de crear cursos de prueba estándar en el que se incluyan diversas secciones de actividades como: archivos, tareas, páginas y foros, además la creación e inscripción de usuarios en el curso todo esto con la finalidad de verificar la confiabilidad y el rendimiento de varios componentes del sistema (como el respaldo y la restauración). Empezando con un tamaño mínimo de curso 10 KB hasta llegar a 1 GB, mientras de mayor tamaño sea el curso más actividades y recursos serán incluidos. A continuación, se presenta la Tabla 5 con los diferentes cursos y la cantidad de actividades que serán creados en cada uno de ellos. Además, se indica el tiempo que se tardó en crear cada uno de los cursos con sus respectivas actividades.

Tamaño del curso	Actividades							Tiempo (segundos)
	Tareas	Páginas	Archivos pequeños	Ficheros grandes	Creación de usuarios	Inscripción de usuarios	Creación de foros (posts)	
10 KB	1	1	1	1	1	1	2	2.6 s
10 MB	10	50	64	2	100	100	20	28.9 s
100 MB	100	200	128	5	1000	1000	500	378.5 s (6 min)
1 GB	500	1000	1024	10	10000	10000	5000	9426.7 (2 hrs)

Tabla 5. Tamaño del curso con sus diferentes actividades

### Tráfico de red entrante – MOOC

En el caso del tráfico de entrada referente a la figura 77, se puede observar que, al crear, inicialmente un curso con un tamaño de 10 KB se genera un tráfico mínimo de 2,63 kbps por el hecho de que se realiza una cantidad mínima de actividades y tareas en el curso, esta es la creación de; (1 tarea, 1 página de actividad, 1 fichero más la creación e inscripción de un usuario como se muestra en la (Tabla 5). En comparación con el curso de 1GB el tráfico generado es de 57,66 kbps siendo mayor por el hecho de que genera un mayor número de actividades.

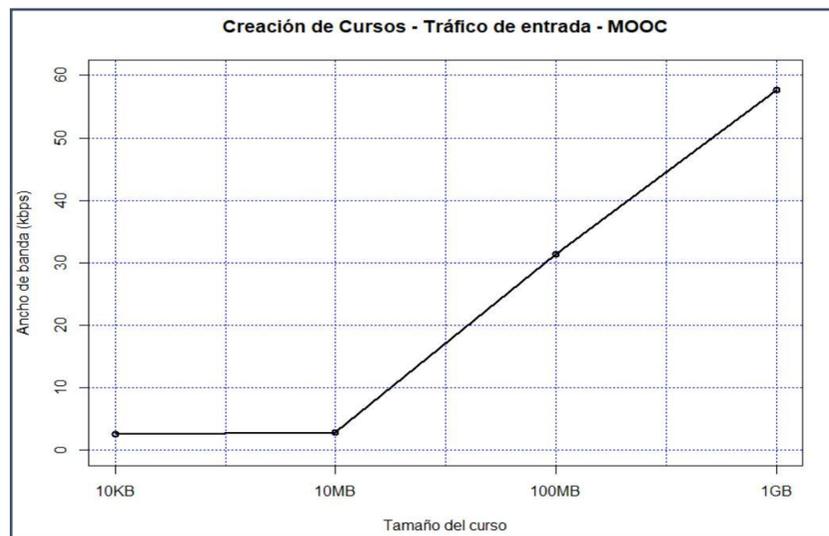


Figura 77. Tráfico de entrada, creación de cursos

### Tráfico de red saliente – MOOC

Por otro lado, el tráfico de red saliente, tomando como referencia a la tabla 5 para la ejecución de las pruebas, se puede notar que al ser un tipo de solicitud GET produce mayor tráfico ya que el servidor nos devuelve toda la información referente a cada uno de las tareas realizadas en los diferentes cursos. Primero el curso de 10KB dando como resultado un consumo de 4,02 kbps, segundo el curso de 10MB con 8,06 kbps, tercero el curso de 100MB con 1420 kbps y finalmente el curso de un 1GB con 1090 kbps, dando como resultado que estos dos últimos cursos son los que mayor tráfico generan al

responder la petición del usuario con toda la información generada al ejecutar las pruebas.

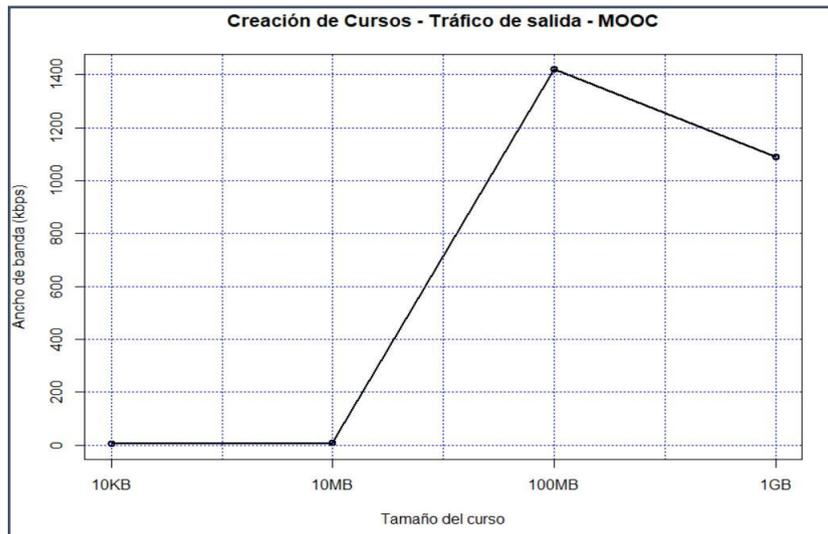


Figura 78. Tráfico de salida, creación de cursos

### Uso de memoria RAM – MOOC

En cuanto a las pruebas de simular la creación de cursos, en la figura 79 se puede observar que el consumo de memoria con 10KB es de 0MB deduciendo que este comportamiento se debe al corto tiempo de 2.6s que se tarda al crear este curso llegando a consumir 123MB con el curso de 1GB.

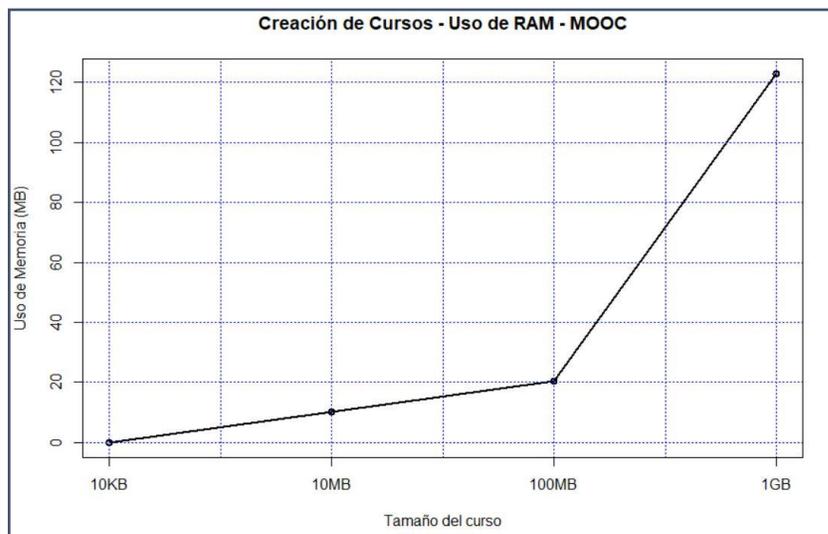


Figura 79. Consumo de memoria RAM, creación de cursos

## Uso de CPU – MOOC

Referente al uso del CPU figura 80, al crear un curso de 10KB el consumo es mínimo dando como resultado el consumo del 3% por el hecho de ejecutarse en un corto lapso de tiempo de 2.6 segundos como se muestra en la tabla 4 a diferencia del curso con un tamaño de 1GB que llega a consumir un máximo de 93%.

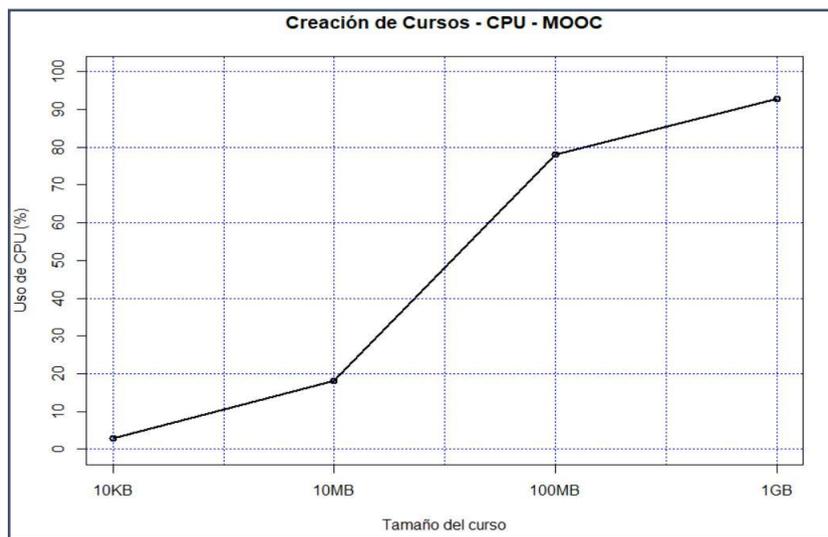


Figura 80. Consumo del CPU, creación de cursos

### 10.1.2 ACCESO CONCURRENTES DE USUARIOS

Mediante esta prueba se simuló el acceso concurrente de usuarios a la plataforma, con el fin de medir el rendimiento del servidor, para encontrar el punto de equilibrio con el cual el funcionamiento de la aplicación WEB no se vea afectada si este excede el número de usuarios determinados en la prueba. A su vez Moodle permite realizar pruebas a desarrolladores mediante un script que es ejecutado en JMeter, en el que se integran las diferentes actividades que los estudiantes pueden realizar dentro de un curso como, por ejemplo; ingresar al sitio, ver el curso con sus diferentes actividades y recursos, revisar y comentar los foros.

Con el software JMeter se realizó el siguiente plan de pruebas que se detalla en la Tabla 6. Donde inicialmente parte con un número mínimo de 5 usuarios hasta llegar a un máximo de 150 usuarios concurrentes cada una de ellas ejecutándose en tiempo de 1 segundo con un tiempo de repetición (5 veces), de este modo probar el rendimiento del servidor hasta encontrar el punto de equilibrio, dicho punto de equilibrio más adelante será determinado según las pruebas establecidas.

NÚM USUARIOS	TIEMPO	BUCLES
5	1	5
30	1	5
60	1	5
90	1	5
120	1	5
150	1	5

*Tabla 6. Plan de pruebas (usuarios concurrentes)*

#### Tráfico de entrada – MOOC

Según las pruebas obtenidas al realizar la prueba con determinados usuarios concurrentes al sitio se puede observar que el tráfico de entrada (Figura 81) varía muy poco ya que en esta prueba los usuarios solo ingresan al sitio sin realizar ninguna actividad y dichos resultados son que al simular el acceso concurrente mínimo de 5 usuarios se genera un tráfico de 8,8 kbps y con un máximo de 150 usuarios un tráfico de 23,43 kbps.

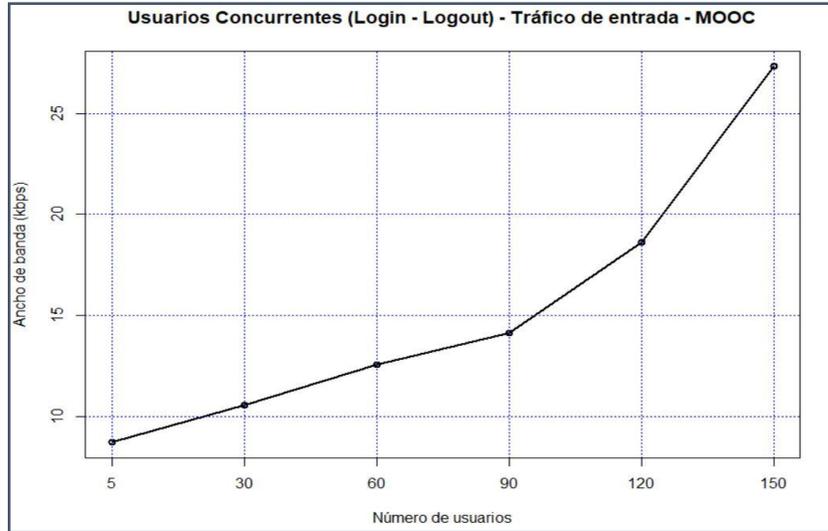


Figura 81. Tráfico de entrada de usuarios concurrentes (Login – Logout)

### Trafico de salida – MOOC

En cambio, el tráfico de salida figura 82, en relación con el tráfico de entrada es mayor por el hecho de que tiene que corroborar si el usuario tiene credenciales para acceder a la plataforma; una vez comprobada si el usuario existe el siguiente paso es comparar los datos para verificar que el usuario y contraseña ingresados sean correctos y finalmente notificarle al usuario que ha accedido satisfactoriamente a la plataforma. Todo este proceso conlleva mayor tráfico de salida dando como resultado que al ingresar 5 usuarios genera un tráfico de 229,22 kbps y al ejecutar una cantidad máxima de 150 usuarios consume 611,61 kbps.

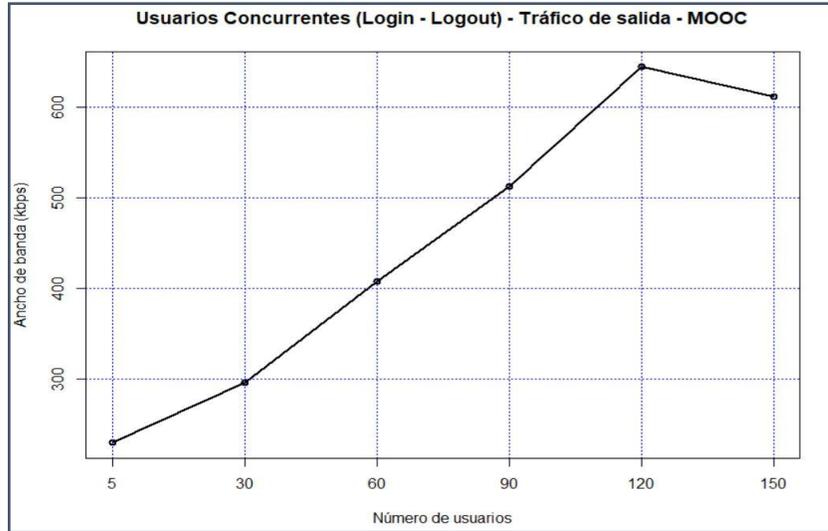


Figura 82. Tráfico de salida de usuarios concurrentes (Login – Logout)

### Uso de memoria RAM – MOOC

Por lo que respecta a la prueba sobre el uso de la Memoria RAM, se empezó con la ejecución de 5 usuarios concurrentes consumiendo 61,44 MB hasta un máximo de 150 con 1533,44 MB, con el cual se logró determinar que a partir de 120 usuarios el servidor empieza a rechazar las peticiones.

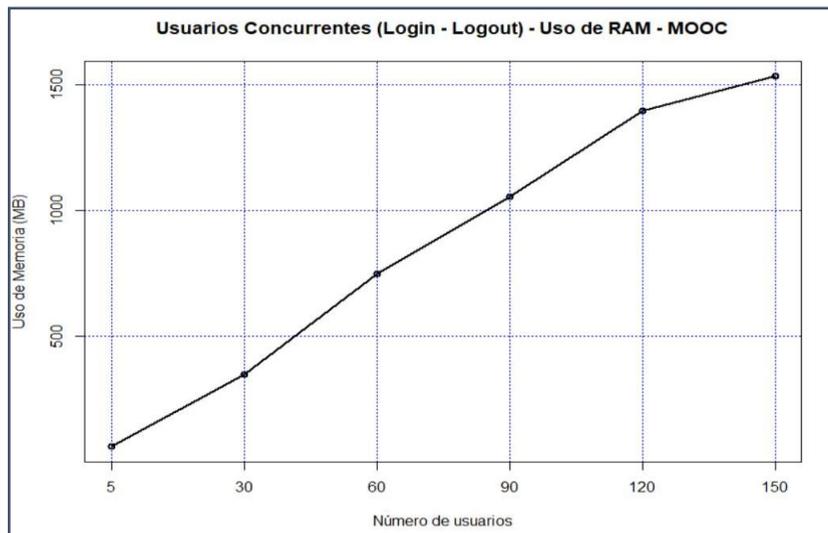


Figura 83. Consumo de RAM, usuarios concurrentes (Login – Logout)

### Uso de CPU – MOOC – Usuarios concurrentes

A diferencia del consumo de memoria el CPU no se sobrecarga al ejecutar las peticiones que los usuarios solicitan entre los diferentes módulos descritos anteriormente de tal manera que al ejecutar la prueba con 5 usuarios el consumo del CPU mínimo es de 11% llegando a un máximo del 20% cuando se evalúa con 150 usuarios concurrentes en comparación a lo que sucede al crear un curso de 1GB su consumo máximo es del 93% como se muestra en la figura 80.

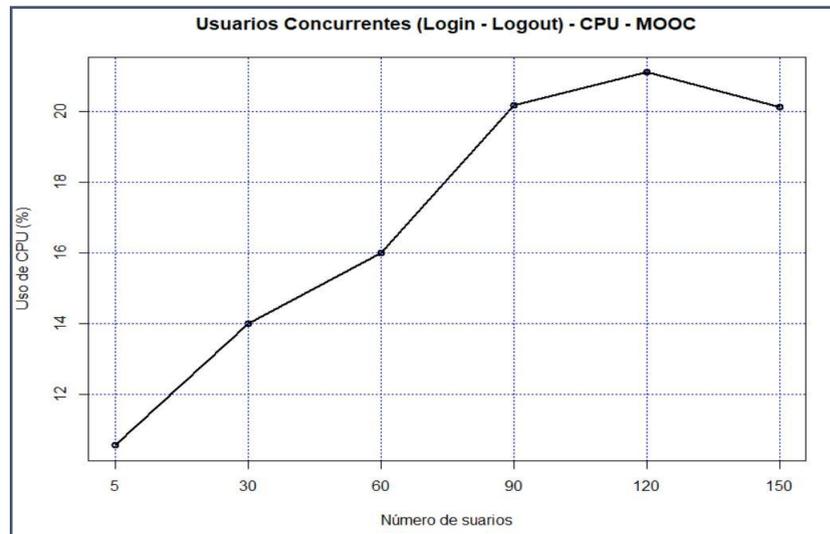


Figura 84. Consumo de CPU, usuarios concurrentes (Login – Logout)

### Latencia generada, usuarios concurrentes

Referente a la latencia generada por usuarios concurrentes en la Figura 85, se puede observar que con 5 usuarios la latencia generada es de 0.9 segundos, esto se debe a que se realiza pocas tareas en el servidor, mientras con 120 usuarios se genera una latencia de 23 segundos ya que se realizan múltiples tareas consecutivamente y la latencia generada es mayor.

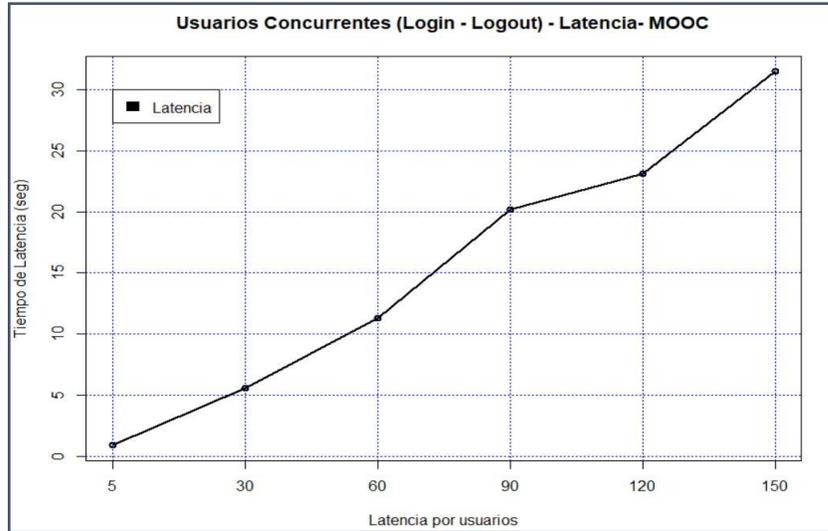


Figura 85. Latencia, usuarios concurrentes (Login – Logout)

### 10.1.3 DISPONIBILIDAD DE LA PLATAFORMA (ACTIVIDADES)

Por lo que se refiere a la disponibilidad de la plataforma, en el cual los usuarios realizan diferentes actividades sobre el servidor MOOC, se implementó la simulación de las siguientes tareas.

1. Ver curso
2. Ver página de actividades
3. Ver actividad del foro
4. Ver discusión del foro
5. Llenar formulario para responder el foro
6. Finalmente ver los participantes del curso

En la tabla 7 se detallan los parámetros que se utilizaron para realizar la ejecución de las pruebas. En cuanto a los tiempos dependiendo el número de usuarios se incrementó el periodo de subida (Ramp-up period), es decir para 5 usuarios que puedan navegar en los diferentes módulos que ofrece la plataforma se asignó un periodo de 1 segundo con un bucle de 5 repeticiones; seguidamente para 30 usuarios el periodo de subida fue de 6 segundos así mismo con un bucle de 5 y así sucesivamente. La razón de aumentar el periodo, permite determinar en cuanto

tiempo se ejecutan n cantidad de hilos (usuarios), por ejemplo, para 30 usuarios, teniendo un periodo de 6 segundos; JMeter inicializa 5 usuarios en un intervalo de 6 segundos, ejecutando las tareas indicadas anteriormente, todo esto hasta completar el número de hilos. En efecto esta prueba permite probar el rendimiento del servidor al realizar múltiples tareas con un determinado número de usuarios. Inicialmente se ejecuta la prueba con 5 usuarios luego con 30 y de ahí en adelante con una escala de 30 hasta llegar a 150 usuarios, según los resultados obtenidos con la ejecución de las pruebas, se determina el punto de equilibrio con 120 usuarios puesto que al ejecutar 150, varias peticiones emitidas por parte de los usuarios hacia el servidor estas son rechazadas.

NÚM USUARIOS	PERIODO DE SUBIDA	BUCLES
5	1	5
30	6	5
60	10	5
90	20	5
120	30	5
150	40	5

*Tabla 7. Plan de pruebas, concurrencia de usuarios (actividades)*

### Tráfico de entrada MOOC

El tráfico generado al ejecutar las pruebas de concurrencia tanto como en la navegabilidad en los diferentes módulos que presenta la plataforma, no existe mayor diferencia por el simple hecho que le estamos agregando un periodo de subida diferente a cada grupo de usuarios para agregar mayor carga y de esta forma comprobar hasta qué cantidad de usuarios la plataforma responde a todas las solicitudes. Inicialmente al ejecutar 5 usuarios con un periodo de subida de 1 llega a generar un tráfico entrante de 15,54 kbps llegando a 26 kbps con 120 usuarios que su periodo de subida es de 30. Ya que al probar con más de 120 usuarios el servidor empieza a rechazar peticiones y por ende empieza a descender el tráfico generado tanto en el de entrada como de salida, tal como se muestra en la figura 86.

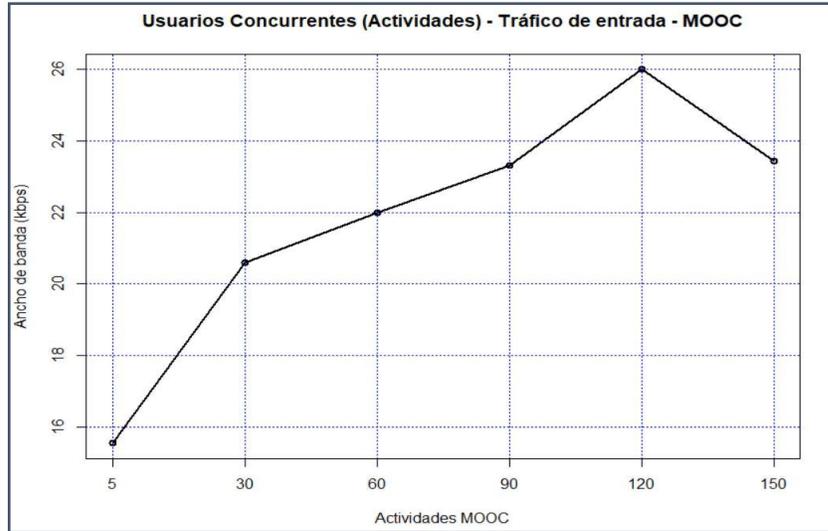


Figura 86. Tráfico de entrada, usuarios concurrentes (Actividades)

### Tráfico de salida MOOC

En cuanto al tráfico de salida, referente a la disponibilidad de la plataforma, como anteriormente ya se menciona, no existe mayor diferencia, tanto así que el tráfico generado con 5 usuarios es de 496,58 kbps y con 120 usuarios 827,56 kbps. De igual forma con el tráfico de salida figura 87, concurrencia de usuarios, a partir de los 120 usuarios el servidor empieza a rechazar peticiones de los usuarios, que a tal punto llega a descender a 727,07 kbps con 150 usuarios.

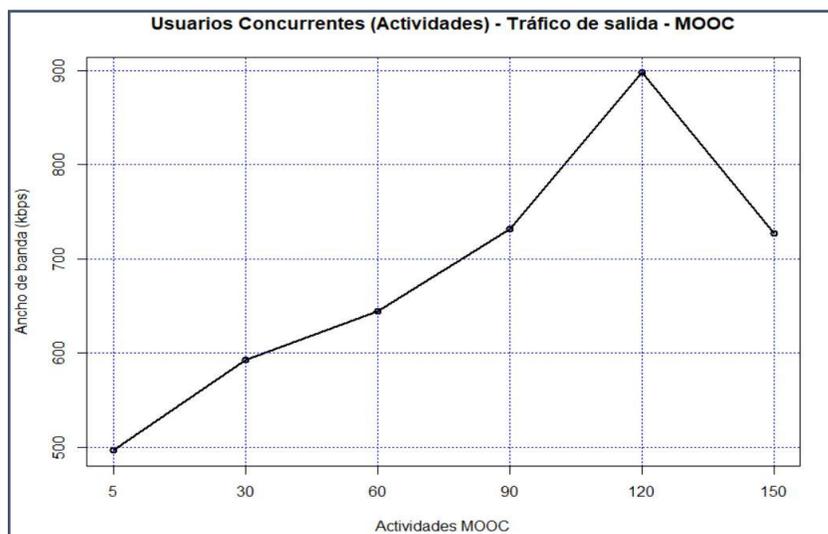


Figura 87. Tráfico de salida, usuarios concurrentes (Actividades)

### Consumo del CPU, disponibilidad de la plataforma

Referente al uso del CPU, se puede observar en la Figura 88 que las actividades realizadas en el servidor MOOC hacen que el consumo del CPU sea mínimo, por el hecho de que las solicitudes de los usuarios son de tipo GET, en el cual el recurso más afectado es la memoria RAM como se detalla posteriormente.

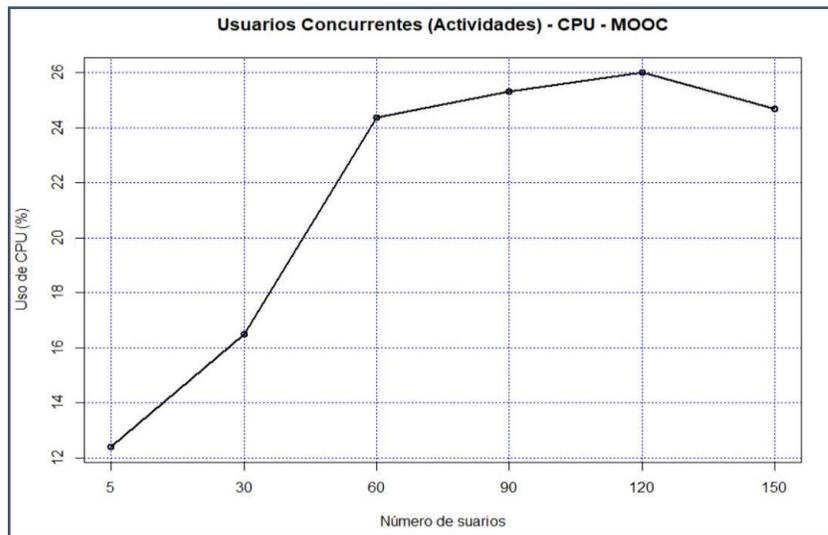


Figura 88. Uso de CPU, usuarios concurrentes (Actividades)

### Uso de memoria RAM

Por lo que se refiere al consumo de memoria RAM, al realizar las actividades mencionadas anteriormente, se observa un gran uso de la memoria, ejecutando desde los 5 usuarios con un consumo de 10,24 MB llegando a consumir 931.84 MB con 120 usuarios que es donde se encontró el punto de equilibrio, ya que al tener más usuarios realizando tareas, el servidor empieza a quedarse sin memoria RAM y los procesos se vuelven más lentos, provocando que la plataforma deje de responder, lo que sucede al ejecutar 150 usuarios tal como se muestra en la Figura 89.

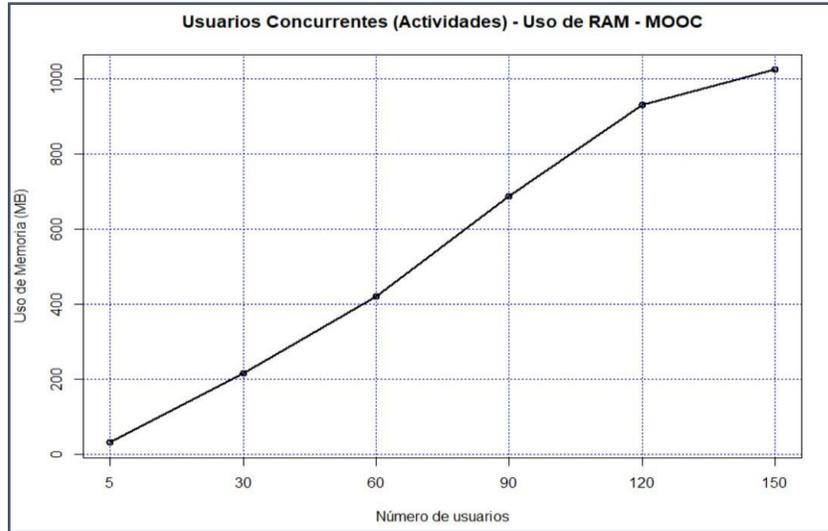


Figura 89. Consumo de Memoria RAM, usuarios concurrentes (Actividades)

### Latencia generada, disponibilidad de la plataforma

Referente a la latencia generada en la Figura 90, se empezó con un número de 5 usuarios realizando tareas en el servidor MOOC, hasta llegar a 150 usuarios. La latencia generada es de 31,47 segundos con 120 usuarios ya que con 150 usuarios el servidor MOOC empieza a rechazar las peticiones (perdida de paquetes).

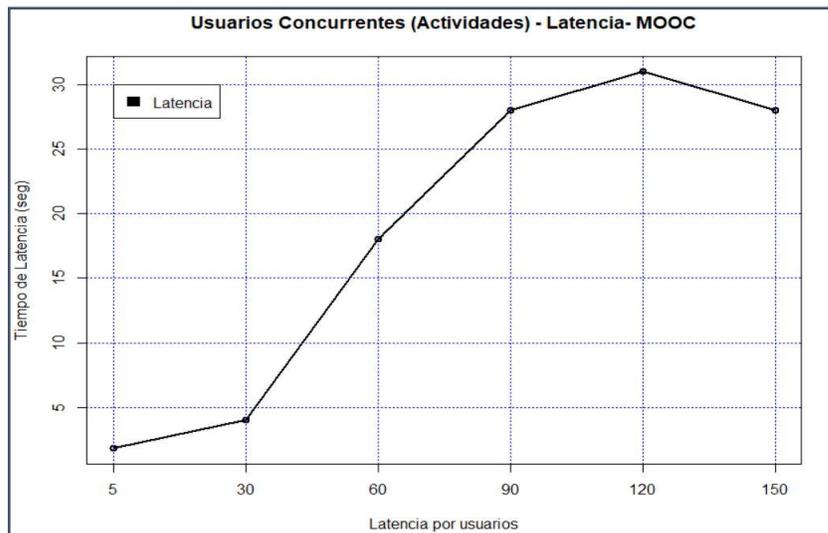


Figura 90. Latencia, usuarios concurrentes (Actividades)

## 10.2 PRUEBAS DE SERVIDOR WEBRTC

Para evaluar el rendimiento del servidor WebRTC es necesario determinar el ancho de bando y el retardo para garantizar que el servicio de videoconferencia integrado a la plataforma MOOC estén activos y funcionales. De igual manera para evaluar los recursos de Memoria RAM, CPU y Ancho de banda se utilizó Zabbix; adicionalmente se utilizó la herramienta de Iperf.

**Iperf** es una herramienta cliente – servidor multiplataforma disponible para Windows y Linux que se utiliza para hacer pruebas en redes informáticas, permitiendo medir la velocidad máxima que alcanzan dos equipos conectados en una red local. Además de poder ajustar ciertos parámetros de una red para optimizar o ajustar.

Las pruebas de medición se ejecutaron en el servidor alojado en Openstack como en Microsoft Azure, de los cuales la velocidad máxima de cada uno de ellos se presenta en las siguientes gráficas.

En la figura 91 se expone la medición realizada en Openstack, donde inicialmente 0 segundos con una tasa de transferencia de 128 KBytes nos da como resultado un ancho de banda de 97.8 Mbits/sec y en un intervalo de 0 – 10 segundos una tasa de transferencia de 2.75 GBytes con un ancho de banda de 2.36 Gbits/sec.

```
root@srv-bbb:/home/carper# iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)

4] local 192.168.100.26 port 5001 connected with 192.168.100.17 port 53768
ID] Interval      Transfer      Bandwidth
4]  0.0- 0.0 sec   128 KBytes   97.8 Mbits/sec
5] local 192.168.100.26 port 5001 connected with 192.168.100.17 port 53770
5]  0.0-10.0 sec  2.75 GBytes  2.36 Gbits/sec
```

Figura 91. Medición de ancho de banda en Openstack

A comparación con la medición en Openstack los resultados arrojados en Microsoft Azure son sumamente menor debido a que el cliente al conectarse al servidor WebRTC alojado en la nube pública, este al responder dicha solicitud lo hace a través de varios

saltos y por ende pierde ancho de banda. Los resultados se presentan en la figura 92, donde con un delay (retardo) de 1 segundo se tiene un ancho de banda de 1.03 Mbits/sec y con retardo de 12 segundos nos da 9.32 Mbits/sec de ancho de banda.

```
root@Server-BBB:~# iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
 4) local 167.99.6.73 port 5001 connected with 181.198.204.66 port 38894
ID] Interval      Transfer    Bandwidth
 4]  0.0- 1.0 sec   128 KBytes  1.03 Mbits/sec
 5) local 167.99.6.73 port 5001 connected with 181.198.204.66 port 38896
 5]  0.0-12.0 sec  13.4 MBytes  9.32 Mbits/sec
```

Figura 92. Medición de ancho de banda en Microsoft Azure

### Tráfico de entrada Servidor WebRTC

En cuanto al tráfico de entrada generado en el servidor WebRTC figura 93, inicialmente con 3 usuarios es de 0.2 Mbps llegando a consumir hasta 5 Mbps aproximadamente con 15 usuarios, esto solo establecido Videoconferencias sin compartir el escritorio por parte de Docente.

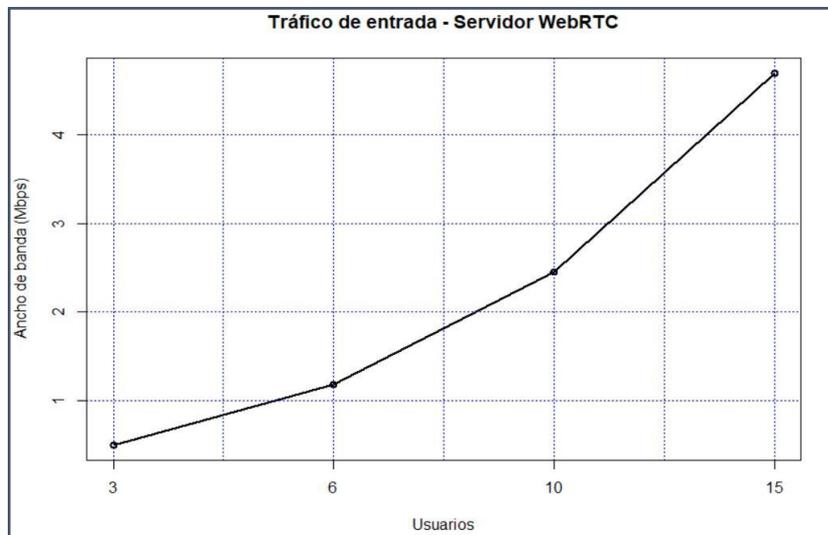


Figura 93. Tráfico de entrada de Videoconferencias establecidas

### Tráfico de salida Servidor WebRTC

Referente al tráfico de salida figura 94, se puede notar que inicialmente con 3 usuarios consume un ancho de banda de 1 Mbps, con 10 usuarios 6 Mbps; oscilando alrededor de 10 Mbps cuando están conectados 15 usuarios al mismo tiempo. El mayor consumo

se produce al momento de realizar el compartimiento de escritorio, tanto así que llega a consumir hasta 21 Mbps aproximadamente, esto puede variar en relación al número de usuarios que se encuentren en una sala de videoconferencia.

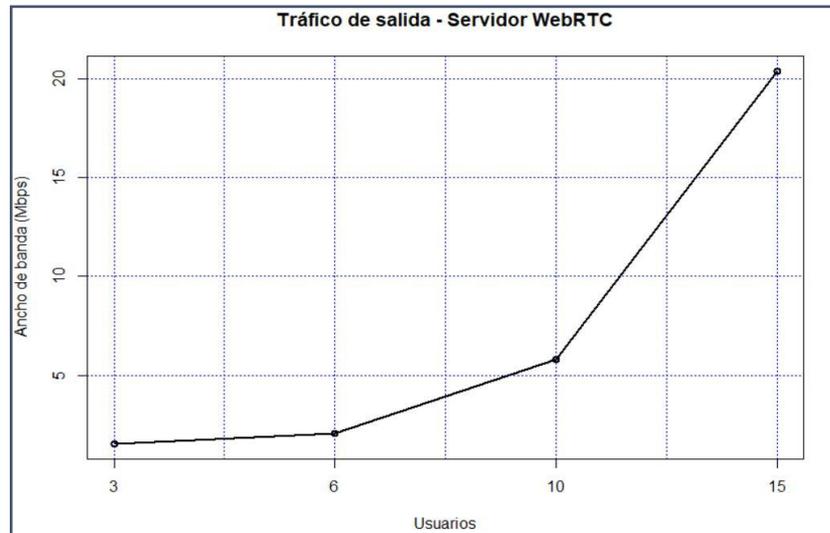


Figura 94. Tráfico de salida de Videoconferencias establecidas

### Consumo del CPU, Servidor WebRTC

El consumo del CPU inicialmente al establecerse una videoconferencia de varios usuarios es mayor, oscilando alrededor del 78%, pero una vez establecida la videoconferencia cambia dicho valor donde inicialmente con 3 usuarios se genera un consumo del 5% llegando al 69% con 15 usuarios, el mismo caso que en el tráfico de red saliente el consumo del CPU se incrementa al compartir el escritorio, tal como se muestra en la figura 95.

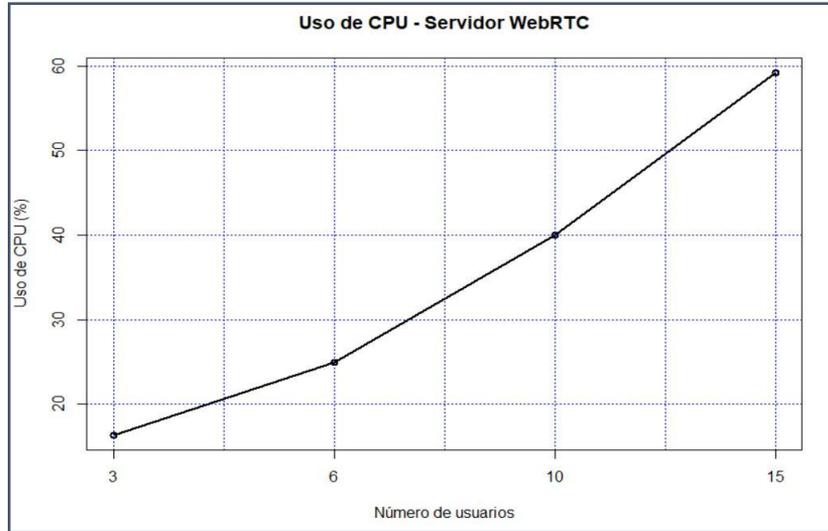


Figura 95. Consumo de CPU, servidor WebRTC

### Consumo de Memoria RAM, Servidor WebRTC

Por lo que se refiere al consumo de memoria RAM, al establecer una videoconferencia, se observa que el consumo inicial con 3 usuarios es de 0.2 GB llegando a consumir 2 GB con 15 usuarios, tal como se muestra en la figura 96.

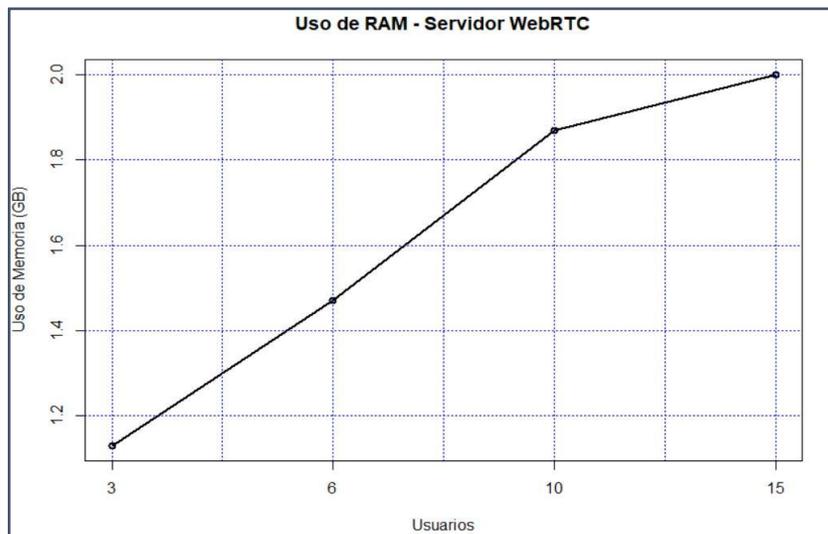


Figura 96. Consumo de Memoria RAM, servidor WebRTC

## 11 CONCLUSIONES

Los sistemas MOOC se han convertido en el primer escalón que habilita el uso de las TICS para la educación superior, remota y ubicua permitiendo de esta forma mejorar la calidad de vida de los ecuatorianos y alineándose con la nueva matriz productiva.

La plataforma MOOC es compatible con los estándares de accesibilidad, no solo en relación con la interfaz web, sino también para apoyar a los estudiantes en la configuración del entorno y el contenido de aprendizaje de acuerdo a sus necesidades.

El Uso de la nube de Azure nos permite características de escalabilidad y alta disponibilidad de la plataforma para poder agregar número de sesiones concurrentes, cursos, actividades y funcionalidades de la plataforma.

El uso de herramientas colaborativas como WEBRTC, requieren una mayor capacidad de ancho de banda y minimizar la latencia aplicando métodos de calidad de servicio (QoS) más aún se trata de sistemas infocentros/telecentros comunitarios rurales de

nuestro país, que usan sistemas VSAT satelitales o sistemas celulares de tercera generación GSM.

Dentro de las funcionalidades validamos la grabación de las clases o cursos (VideoMark), lo que habilita a los alumnos una herramienta de consulta y revisión, para consolidar y aclarar conceptos de aprendizaje específicos en un entorno de aprendizaje basado en video.

## 12 RECOMENDACIONES

Para la etapa de pruebas en el servidor MOOC se recomienda que principalmente sean realizadas en un servidor explícitamente para pruebas y no en un servidor en producción, ya que podrían dañar el correcto funcionamiento de la plataforma.

En lo que respecta al servicio WebRTC, si opta por almacenar las grabaciones de las respectivas clases dentro de la plataforma, se recomienda calcular el tamaño de almacenamiento.

Es necesario considerar que al usar la nube pública de Microsoft Azure, lo más recomendable es abrir solo los puertos a utilizar en la plataforma para ofrecer mayor seguridad y confidencialidad sobre los servicios desplegados.

Previo a realizar una Videoconferencia se debe tener instalado Adobe Flash en el computador del cliente, ya que se necesita este complemento para que funcione correctamente. Además, de este complemento es necesario configurar seguridad en los servicios WebRTC ya que, si la página no tiene certificados, Google Chrome no permitirá acceder al micrófono y cámara para lo cual se recomienda usar Let's Encrypt para configurar HTTPS.

Los dispositivos móviles no soportan Flash y por ende no se puede acceder directamente a la Videoconferencia, para ello es recomendable que accedan mediante el cliente HTML5, el mismo que aporta mayor fluidez y se adapta a cualquier dispositivo móvil.

### 13 REFERENCIAS BIBLIOGRÁFICAS

- [1] J. Salinas, «Redes y educación: Tendencias en educación flexible y a distancia,» Oviedo, 1998, pp. 141-151.
- [2] J. F. Guerra, J. . M. Jaque, A. . O. Jiménez y M. E. Álvarez Hernández, «Servicios de soporte en una plataforma de tele-educación,» de *Servicios de soporte en una plataforma de tele-educación*, 2007, pp. 3-16.
- [3] F. Forero Avella y A. A. Hernández Rodríguez, *EDUCACION EN LA NUBE UN NUEVO ESPACIO PARA LA EDUCACION A DISTANCIA*, 2013.
- [4] V. . R. Curran, «Tele-education,» *Journal of telemedicine and telecare*, vol. 12, nº 2, pp. 57-63, 1 3 2006.
- [5] A. Avilés López-Sepúlveda, *Teleeducación comienzos y futuro*, 2001.
- [6] I. Ros Martínez de Lahidalga, «Moodle, la plataforma para la enseñanza y organización escolar,» 2008.
- [7] M. d. L. Hernández Rodríguez, *Método de migración al modelo Cloud Computing Software as a Service de un LMS Open Source*, México, 2016.
- [8] J. M. Palacios Bayas, *Sistema de tele-educación como servicio sobre una infraestructura de computación en la nube para la unidad educativa del milenio*

Angel Polibio Chavez, 2016.

- [9] J. E. Caballero González y O. Ivañez González, «La educación a distancia como modalidad de estudio en el Proyecto " Mais Médicos para Brasil",» Brasil, 1999, Editorial Ciencias Médicas, 2015, pp. 191-198.
- [10] E. R. Muñoz Vara, *Implementación de enlaces inalámbricos aplicado a teleeducación en zonas rurales*, Perú, 2014.
- [11] W. S. Potosí Sanipatín, *Servicio de Tele-Educación para la modalidad semipresencial del Instituto Técnico Imbabura.*, 2017.
- [12] L. Y. Maza, C. F. Espinosa, V. P. Damián Molina y E. Martínez Aguilar, *SLOR: Simple Learning Object Repository, diseñando un repositorio de objetos de aprendizaje para la plataforma de teleeducación EducAD-UNACH (basada en MOODLE)*, 2006.
- [13] M. Carracedo, C. Pérez, P. Ramírez y B. Salazar, *Implantación coordinada del entorno virtual Moodle y su utilización en la Escuela Universitaria de Informática de la Universidad Politécnica de Madrid*, Jornadas Internacionales de Innovación Educativa (INECE), 2009.
- [14] A. H. Gallego, R. T. Gimeno, F. M. Gutiérrez y J. Velázquez Saornil , *TELE ENSEÑANZA: NUEVOS MÉTODOS PARA EL APRENDIZAJE DE LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA APLICADOS A PROYECTOS DE INGENIERÍA*, Madrid, 2010.
- [15] C. F. Meléndez Tamayo, *Plataformas virtuales como recurso para la enseñanza en la universidad: análisis, evaluación y propuesta de integración de moodle con herramientas de la web 2.0*, Madrid: Universidad Complutense de Madrid, 2013.
- [16] P. Colás Bravo, *Internet y aprendizaje en la sociedad del conocimiento*, 2003.
- [17] R. C. Clark y R. E. Mayer, *E-learning and the science of instruction: Proven guidelines for consumers and designers of multimedia learning*, John Wiley & Sons, 2016.
- [18] J. G. Ruiz, M. J. Mintzer y R. M. Leipzig, *The impact of e-learning in medical education*, vol. 81, LWW, Ed., 2006, pp. 207-212.
- [19] M. Area y J. Adell, «E-learning: enseñar y aprender en espacios virtuales,» de *Tecnología Educativa. La formación del profesorado en la era de Internet*, 2009, pp. 391-424.
- [20] F. J. García Peñalvo, «Estado actual de los sistemas e-learning,» de *Teoría de la*

*Educación. Educación y Cultura en la Sociedad de la Información*, 2005.

- [21] W. Watson y S. L. Watson, «An Argument for Clarity: What are Learning Management Systems, What are They Not, and What Should They Become., » vol. 51, nº 2, p. 7, 2007.
- [22] . L. F. Ortiz, «Campus Virtual: la educación más allá del LMS,» vol. 4, p. 3, 2007.
- [23] A. Marinos y G. Briscoe, «Community cloud computing, » de *IEEE International Conference on Cloud Computing*, 2009.
- [24] P. Mell y T. Grance, «The NIST definition of cloud computing, » vol. 14, p. 7.
- [25] L. F. Barrios Espino, «Cloud computing como una red de servicios,» de *Instituto Tecnológico de Costa Rica*, 2009.
- [26] L. Aguilar Joyanes, «COMPUTACIÓN EN LA NUBE: Notas para una estrategia española en cloud computing,» de *Revista del Instituto Español de Estudios Estratégicos*, 2013.
- [27] N. R. Rodríguez, M. A. Murazzo, S. B. Chávez y M. J. Guevara, «Arquitectura de cloud computing híbrida basada en tecnología open source,» de *XX Congreso Argentino de Ciencias de la Computación (Buenos Aires, 2014)*, 2014.
- [28] Wikipedia. [En línea]. Available: [https://upload.wikimedia.org/wikipedia/commons/thumb/b/b5/Cloud\\_computing.svg/1200px-Cloud\\_computing.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/b/b5/Cloud_computing.svg/1200px-Cloud_computing.svg.png). [Último acceso: 10 06 2018].
- [29] K. Jackson, C. Bunch y E. Sigler, *OpenStack cloud computing cookbook*, Packt Publishing Ltd, 2015.
- [30] B. Galarza, G. Zaccardi, D. Encinas y D. M. Morales, «Análisis de despliegue de una IaaS utilizando Openstack,» de *XXI Congreso Argentino de Ciencias de la Computación (Junín, 2015)*, Argentina, 2015.
- [31] Moodle. [En línea]. Available: [https://docs.moodle.org/all/es/Acerca\\_de\\_Moodle](https://docs.moodle.org/all/es/Acerca_de_Moodle). [Último acceso: 10 06 2018].
- [32] BigBlueButton, «BigBlueButton Open Source Web Conferencing, » [En línea]. Available: <http://docs.bigbluebutton.org/install/install.html>. [Último acceso: 10 06 2018].
- [33] NGINX, «NGINX,» [En línea]. Available: <https://nginx.org/en/>. [Último acceso: 10 06 2018].

- [34] FreeSWITCH, «What is FreeSWITCH?,» [En línea]. Available: <https://freeswitch.com/>. [Último acceso: 10 06 2018].
- [35] Apache Tomcat, «What is Apache Tomcat?,» [En línea]. Available: <http://tomcat.apache.org/>.
- [36] redis, «What is about Redis?,» [En línea]. Available: <https://redis.io/>.