

Uso de contenedores Docker en el entorno educativo

Ismael de Fez^a, Pau Arce^a, Román Belda^a y Juan Carlos Guerri^a

^aInstituto de Telecomunicaciones y Aplicaciones Multimedia, Universitat Politècnica de València (mail: isdefez@iteam.upv.es, paarvi@iteam.upv.es, robolor@iteam.upv.es, jcguerri@com.upv.es) .

Cduatcev''

Vj g"uug"qhf kht gpv'uqhy ct g"rtqi tco u"kp"vj g"t gcrk|cvkqp"qhl'rcdqtcvqt { "rtcevkegu"mwcmf"gpvc'ku" rtdqigo u"lqt"dqj "vgcej gtu"cpf "uwf gpu"fwg"vq"vj g"lcev"qhl'ectt {kpi "qw"vj g"kpwcncvkqp"cpf" o ckpvpgpcpeg'qhlj g'rtqi tco u"y j kej 'lqo gvk gu'vgpf 'vq'rt gupv'eqo rcvdkkkl' rtdqigo uf gt kxf 'lqo" wrcvgu'Vj ku'ctvkeng'rtqr qugu'vj g"uug'qhf qemgt. "c"vqql'vj cv'rtqxlf gu'c p'gzgewkqp'gpxkqpo gpv'vj cv' eqpvkpu'cm'vj g'rtqi tco u"vq"dg"uwf "kp" c "rtcevkcg. "vj wu"lcehkcvkpi "vj g"i gpgtcvkqp"qhl'rgctpkpi " gpxkqpo gpu'OO qt gqxt. "vj ku'y qtnl'rt gupvu'c 't gcn'gzr gt kpeg'kp"vj g"uug'qhlj g'clqt go gpvkppgf 'vqql' ecttkgf " qw" qxgt " vj tgg" eqwtugu" kp" c " uwdlgev" y kj " rtcevkcn' gzgtekugu" kp" vj g" lkgf" qh' Vgreqo o wplecvkpu'cpf "kplqt o cvkqp"Vgej pqm qku. "f gckkpi "vj g'cf xcpwi gu'cpf "ftcy dcemu'vj cv' vj g"uug'qhf qemgt 'j cu'o gcpv'lqt "dqj 'lcewml'cpf "uwf gpu'0

Mgfy qtf u<F qengt. "xk wcrk|cvkqp. "go wrcvkqp. "o wnk/uf ugo "gpxkqpo gpu" g'rgctpkpi "

Tgiawo gp''

Nc "wkk|cek>p'f g'f kht gpvu'rtqi tco cu'uqhy ct g'gp"rc 't gcrk|cek>p'f g'rt" evkecu'f g'rcdqtcvqt kq'lwrg" eqpnxct "rtdqigo cu'xpvq"rctc "gn'rtqlguqtcf q'eqo q'rctc "gn'cnwo pcf q. "rat "gn'j gejq'f g'ngxct "c" ecdq"rc" kpwcrc'ek>p" { "o cpvpgko kpvq"fg"nqu"rtqi tco cu. "nqu"ewrgu" c "xgegu"uwrgp" rtdqigo cu'f g'eqo rcvdkkkl'cf "f g'kxf qu'f g'cewrck|cekqp gu'Gp"guw'ct v'wq'ug"rtqr qpg'gn'wq'f g' F qengt. "j gttco kpvx" s w" r gto kq'f kur qpgt "f g'wp" gpvtpq"fg" glgewek>p" s w" eqpvkpg"vqf qu'nqu" rtqi tco cu" c "wkk|ct "gp" wpc"rt" evkec. "lcehkc'pf q"cu'rc "i gpgtc'ek>p"fg" gpvtpqu'f g'crt gpf k'clg' Cuko kuo q. "gp"guw'vtcdcl'ug'rt gupw' wpc"gzr gt kpeg'c' t gcn'f g'wq'f g'f'kej c'j gttco kpvx. "ngxc'c" c" ecdq'c' h'rti q'f g'v'gu'ent uqu'gp'wpc'cuki pcwt c'f g'rt" evkecu'f grlco r q'f g'ru'Vgreqo wplecvkpu' { "rcu'Vgepqm q' f'cu'f g'rc "kplqt o cek>p. "f g'nc'p'f q'rcu'xgpxlcu'g'kpeqpxgpkpvgu's w'gn'wq'f g'F qengt" j c "lwr wguq'xpvq"rctc "gn'rtqlguqtcf q'eqo q'rctc "gn'cnwo pcf q0

Rcndt cu'evxg<F qengt. "xk wcrk|cek>p. "go wrc'ek>p. "g'vtpqu'o wnk/ukwgo c. "g'rgctpkpi "

1. Introducción

1.1. Motivación

El uso de las nuevas tecnologías ha tenido un crecimiento exponencial en los últimos años en todos los ámbitos de la vida cotidiana. Este crecimiento ha facilitado la adopción de nuevas herramientas digitales en el campo de la educación y, especialmente, ha hecho posible otorgar una mayor autonomía a los estudiantes, quienes han visto como se incrementan sus posibilidades de acceso a la información, tanto en la forma de hacerlo como en la rapidez.

Sin embargo, la tecnología no presenta una mejora educativa *rgt"ug* si no viene acompañada de innovaciones docentes. En este sentido, con el uso de la tecnología se debe facilitar la creación de un

entorno colaborativo y personalizado, dando lugar a un proceso de formación bien diseñado, tutorizado, guiado y evaluado por el profesorado, pero cuyo principal protagonista sea el alumnado (Calero, 2019).

A pesar de que los alumnos demuestren soltura con las tecnologías en su uso cotidiano, no se debe generalizar y suponer que serán capaces de utilizarlas en los procesos de aprendizaje, seleccionando y evaluando la información relevante (Barroso, 2015). Por tanto, es importante que la figura del docente aparezca como facilitador, y sirva de guía para que el alumno pueda sacar el máximo provecho a las herramientas y a la información (Rosendo, 2010).

Por tanto, incluso en asignaturas con un fuerte componente tecnológico, es importante preparar los recursos didácticos para que el alumnado pueda avanzar de forma progresiva, haciendo hincapié no sólo en los contenidos sino también en la metodología y el proceso de aprendizaje. No tener las destrezas adecuadas o recibir una formación técnica deficiente son algunos de los principales motivos de frustración de los estudiantes cuando se enfrentan a nuevas situaciones educativas en las que la tecnología y las comunicaciones suponen un factor fundamental, como en la educación en línea (Borges, 2005).

Concretamente, y aunque también es extensible a otros ámbitos, en asignaturas del área de Telecomunicaciones y Ciencias de la Información, es habitual el uso de *uq'hy ctg* y aplicaciones dentro del proceso formativo, no sólo con el fin de aprender su uso, sino como herramientas que facilitan el aprendizaje de conceptos y prácticas que, explicadas únicamente de un modo teórico, no tendrían el impacto suficiente para que los estudiantes adquirieran las destrezas requeridas.

La manera en la que se presenten estas herramientas software a los alumnos puede ser decisiva tanto en su aceptación, la fluidez del proceso de aprendizaje o, inclusive, la falta de motivación. Lo último que quieren los alumnos es tener que lidiar con entornos o programas software que nunca han visto, con una pronunciada curva de aprendizaje y sin entender cómo encaja en su proceso formativo.

Por otro lado, en los casos en los que los alumnos utilicen sus propios ordenadores y dispositivos, la casuística puede ser tan variada que los docentes han de lidiar con errores de instalación y dependencias de librerías que consumen mucho tiempo que estaría mejor empleado en la resolución de dudas relativas a la temática de la asignatura.

En este sentido, la posibilidad de distribuir un software informático en un único formato con el fin de que sea ejecutado en cualquier plataforma ha sido un objetivo ampliamente perseguido a lo largo de la historia de la ingeniería de software, no obstante prácticamente imposible de conseguir.

La imposibilidad se origina principalmente en que los principales actores en el mundo de los sistemas operativos (Microsoft y Apple) no parecen predispuestos a facilitar la tarea de proporcionar entornos de ejecución compatibles entre sus respectivos sistemas operativos (Windows y Mac OS).

El tercer sistema operativo en importancia es Linux. Sin embargo, Linux no está representado por una única organización ni empresa y se ofrece en una multitud de distribuciones que, si bien favorecen la innovación de las propias distribuciones, también genera una diversidad de entornos que dificulta la compilación y ejecución de software en diferentes distribuciones de Linux.

A pesar de que Linux es un sistema operativo minoritario en su uso en estaciones de trabajo, se ha convertido en un elemento imprescindible en los entornos educativos debido a la posibilidad de disponer de él de forma incondicional, a la variedad de software educativo y a su facilidad de distribución.

Ante la imposibilidad de desarrollar y ejecutar un mismo software en las diferentes plataformas existentes, a lo largo de la historia, se han planteado diferentes alternativas que podríamos agrupar en tres grupos: lenguajes interpretados, emulación de sistemas operativos y emulación de entornos de ejecución.

1.2. Lenguajes interpretados

La utilización de lenguajes de programación interpretados está ampliamente extendida en el sector de la educación. Entre los lenguajes más populares están Python y Java. Ambos, al igual que el resto de lenguajes interpretados, necesitan de un entorno de ejecución instalado en el sistema operativo anfitrión.

Los entornos de ejecución suelen ser programas de código abierto que están disponibles en todos los sistemas operativos y distribuciones, en el caso de Linux. A pesar de la ubicuidad de estos lenguajes, su utilización en la enseñanza se ciñe al aprendizaje relacionado con la programación (Peña, 2015) o cuando todo el software que se requiere está disponible en uno de estos lenguajes (Díaz, 2018).

1.3. Emulación de sistemas operativos

Incluso en el caso de que se disponga del código fuente del software que se desea utilizar y de que dicho software esté preparado para ser compilado en los principales sistemas operativos, el propio proceso de compilación puede, y suele, generar una casuística que complica su utilización en los entornos educativos.

En estas situaciones es frecuente recurrir a la forma de una imagen de una máquina virtual de Linux como mecanismo de distribución de un entorno de software complejo. Esta solución tiene la ventaja principal de que es posible ejecutar una máquina virtual de Linux en cualquiera de los principales sistemas operativos, incluido Linux, de forma que todos los alumnos, sin importar su elección en materia de equipos informáticos personales, pueden acceder a un entorno común preparado ex profeso para una actividad educativa determinada.

Su principal desventaja, muchas veces determinante, es la cantidad de recursos necesarios por parte del ordenador anfitrión en relación a los siguientes aspectos: el de distribución, puesto que las imágenes de sistemas operativos suelen estar en el orden de Gigabytes; y el de los recursos de ejecución, debido a que la máquina anfitrión debe contar con la memoria y la capacidad de proceso suficiente para que funcionen con fluidez dos sistemas operativos al mismo tiempo.

1.4. Emulación de entornos de ejecución

La historia de la emulación de los entornos de ejecución es larga pero poco prolífica. Por un lado, están las emulaciones que los sistemas operativos Windows y Mac OS han desarrollado con el fin de facilitar la transición tecnológica de sus propios sistemas. Ejemplos de ello serían los entornos de ejecución para la compatibilidad de aplicaciones para Mac OS 9 en OS X o Rosetta y Rosetta2 empleados en las transiciones de arquitecturas de los procesadores utilizados en los ordenadores Mac. En el caso del sistema operativo Windows, también utiliza emulación de entornos de ejecución como es el caso de Windows para la plataforma ARM.

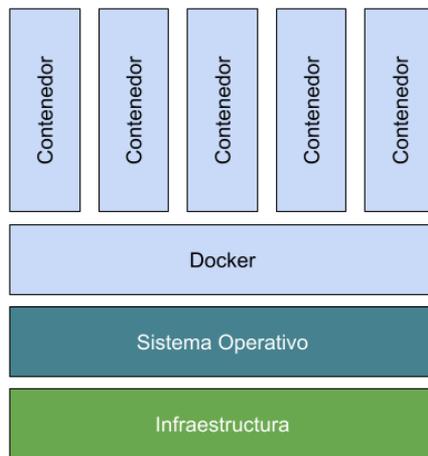
Aparte de estos sistemas de emulación de entornos de ejecución destinados a las transiciones tecnológicas de los sistemas operativos, Docker (Docker, 2021) pretende reducir el esfuerzo que supone pasar del entorno de desarrollo, en diferentes sistemas operativos y versiones, al entorno de despliegue de aplicaciones desarrolladas para Linux en una multitud de distribuciones y versiones que componen los entornos de Cloud Computing.

Con este objetivo, Docker define como único punto de contacto con la máquina anfitrión el espacio de usuario del kernel de Linux. Por encima de éste, todos los recursos de librerías, versiones, datos e interfaces

de red serán de uso exclusivo del proceso ejecutado de este modo. A diferencia con respecto a la emulación de sistemas operativos completos, en Docker no hay emulación de un *jctf y ctg* ni de una arquitectura, por lo que la ejecución es mucho más rápida y necesita de una menor cantidad de recursos. De esta forma, Docker permite disipar la dificultad en la ejecución de software entre versiones y variantes de Linux pero no entre sistemas operativos.

Sin embargo, la popularidad alcanzada por esta solución motivó el desarrollo de versiones de Docker para los sistemas operativos Windows y Mac OS. Docker define un entorno de ejecución de aplicaciones Linux que proporciona un entorno de ejecución de binarios basados en imágenes que se comporta igual, salvando pequeñas diferencias generadas por los sistemas operativos anfitriones.

La Fig. 1 muestra como se ejecutan una serie de contenedores, que es como se denomina a la instancia en ejecución de una imagen de Docker, sobre Docker.



Hkí 03'Gnt wewt c'f g'hwpekqpc o kpvq'f g'F qengt "

Por esta razón, en este artículo se plantea la utilización de herramientas de emulación de entornos de ejecución de software que facilitan la generación de entornos de aprendizaje, permitiendo a los estudiantes focalizarse en los objetivos de aprendizaje diseñados, evitando las fricciones con la interacción de software innecesarias para alcanzar los resultados de aprendizaje.

2. Objetivos

El objetivo principal de este artículo es exponer las principales actividades llevadas a cabo por un grupo de profesores e investigadores de la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València (UPV) para el desarrollo de herramientas en un entorno de ejecución emulado a través de Docker y su utilización en la docencia en asignaturas con una fuerte base tecnológica. Además, se pretende analizar la aceptación por parte del alumnado de estas herramientas y cuantificar el impacto que ha tenido su uso, comparando tres cursos en los que la implantación de las herramientas ha sido diferente y progresiva.

3. Desarrollo de la innovación

Esta experiencia educativa se engloba dentro de la asignatura “Comunicaciones Multimedia”, perteneciente al Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación, impartido en la UPV.

La asignatura tiene una parte de teoría de aula, práctica de aula y práctica de laboratorio. Por lo que respecta a la parte de las prácticas de laboratorio, éstas versan sobre aspectos heterogéneos referentes a las comunicaciones multimedia, concretamente:

- Estudio del efecto de las pérdidas en el vídeo comprimido.
- Transmisión de vídeo entre un servidor y un cliente.
- Compresión y perfil de vídeo comprimido.
- Envío de contenido multimedia a través de protocolos adaptativos.
- Análisis de la trama de televisión MPEG-TS.
- Análisis de protocolos para el transporte de contenidos multimedia en tiempo real.

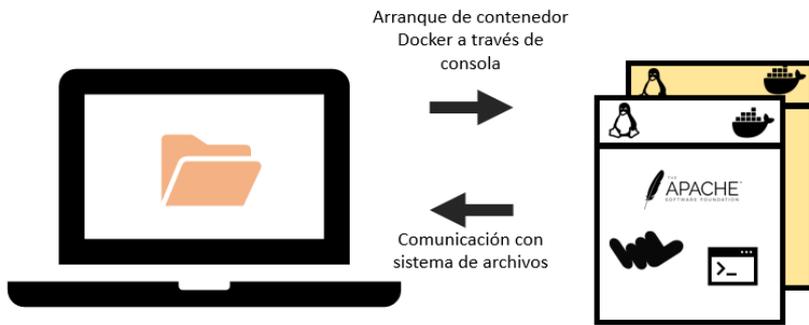
La heterogeneidad de las temáticas y la propia naturaleza de la asignatura hace que, para poder llevar a cabo las prácticas, se necesite la utilización de distintos programas de software específicos que se encargan de determinadas tareas, como la codificación de vídeo, la transmisión de paquetes por IP, la simulación de pérdidas en el canal, la fragmentación de un contenido adaptado a un estándar de vídeo adaptativo, etc.

Así, la innovación consiste en la utilización de una herramienta de emulación de entornos de ejecución, concretamente Docker, empleada como base para la ejecución de distintos programas utilizados en las prácticas, a través de la generación de una imagen de Docker. La imagen de Docker se crea por los profesores de la asignatura al principio del curso y contiene todos (y únicamente) aquellos programas necesarios para la realización de las diferentes prácticas que componen el curso.

A diferencia de otras soluciones basadas en la utilización de entornos virtuales disponibles en servidores remotos, el uso de de la misma herramienta, Docker, tanto en los ordenadores de los laboratorios como en los ordenadores personales de los alumnos, ofrece ventajas como la velocidad de ejecución, la no dependencia de servicios ofrecidos por la universidad y el aprendizaje de una herramienta ampliamente utilizada en entornos profesionales.

De esta forma, ya sea en los equipos disponibles en el laboratorio de la Escuela o en los equipos que dispone el alumnado, únicamente es necesario instalar el entorno de ejecución de Docker (disponible de forma gratuita para Windows, Mac OS y Linux) y, una vez instalado, ejecutar la imagen correspondiente. Para ello basta con escribir un comando sencillo en la consola del sistema.

A modo de ejemplo, la Fig. 2 muestra un esquema de la utilización de Docker, cuyo símbolo es una ballena con contenedores. En primer lugar, desde la consola/terminal se ejecuta un comando de Docker para arrancar el contenedor o contenedores deseados. Se pueden arrancar tantos contenedores como se deseen, pero en las prácticas se utilizarán un máximo de 2 contenedores, los cuales tienen una imagen de Ubuntu. Así, una vez se está ejecutando un determinado contenedor, se puede trabajar en él ejecutando los programas necesarios, por ejemplo FFmpeg, comandos de Linux o Apache, entre otros. Es importante destacar que, el directorio de trabajo del contenedor Docker y el directorio de trabajo local se enlazan de forma que desde el contenedor Docker es posible acceder a ficheros disponibles en el equipo local y viceversa.



Hki 04'Gis wgo c'f'g'w'k'k'c'ek'p'f'g'F'qengt''

Concretamente, la imagen de Docker puesta a disposición de los alumnos está basada en la imagen oficial de Ubuntu 18:04 para Docker, a la que se le han agregado posteriormente los siguientes programas:

- FFmpeg: software para la codificación de audio y vídeo (Ffmpeg, 2021).
- MP4Box: software para la paquetización de contenido y segmentación DASH.
- Evalvid: software para la transmisión de vídeo y generación de trazas.
- Netcat: herramienta de red para la apertura de puertos TCP/UDP.
- Tcpdump: herramienta para analizar el tráfico de la red y llevar a cabo la generación de trazas.
- Apache: servidor web de código libre (Apache, 2021).

4. Resultados

La experiencia presentada en este artículo se ha llevado a cabo durante los cursos 2018-2019 (curso 1) y 2019-2020 (curso 2). Además, se toma como referencia el curso 2017-2018 (curso 0), en el cual no se empleó Docker para la realización de las prácticas de esta asignatura.

En el curso 0, las prácticas se realizaron de forma presencial en un laboratorio habilitado. Los ordenadores utilizaban Windows 10, el sistema operativo con el que está más familiarizado el alumnado. En dichos ordenadores se instalaron los diferentes programas necesarios para la realización de las prácticas.

En una de ellas, en la que se realiza la transmisión de un vídeo, resultaba necesario el uso de dos ordenadores diferentes, ya que uno actuaba como servidor y otro como cliente. Esto causaba que dicha práctica se tuviera que realizar por parejas, al disponer de equipos limitados.

El hecho de necesitar múltiples programas para la realización de las prácticas representaba para los profesores un inconveniente por el hecho de tener que realizar la instalación de dichos programas así como el correspondiente mantenimiento y actualización. El correcto funcionamiento de los programas utilizados va ligado a la actualización de las versiones que los propietarios realicen del software, así como de la compatibilidad con la versión del sistema operativo utilizado. Por la propia experiencia del profesorado, cierto software utilizado durante un curso puede dejar de funcionar correctamente debido a problemas de compatibilidad con actualizaciones del sistema operativo, lo que obliga a cambiar de software o a modificar las tareas de una determinada práctica para adaptarse a las funcionalidades del software utilizado.

Además, la necesidad de utilizar unos equipos concretos habilitados en un laboratorio prácticamente obligaba a los alumnos a realizar las prácticas en dicho laboratorio, por la dificultad de tener que instalar varios programas de software en sus ordenadores. Esto suponía una inquietud añadida a los estudiantes, ya que tenían un tiempo limitado para la ejecución de cada práctica, ante la dificultad de poder llevarla a cabo desde sus propias casas. Por ejemplo, para la práctica de transmisión de contenidos, aparte de la dificultad

añadida de instalación de múltiples programas, requerían de dos equipos (uno actuando como servidor y otro como cliente), lo cual no siempre resulta factible para todos los alumnos.

Debido a dichos problemas, se decidió la utilización de una herramienta de emulación de entornos de ejecución en el curso siguiente. Con el uso de dicho entorno se evitan problemas de compatibilidad y de actualizaciones, ya que cada contenedor representa una imagen “congelada” que contiene una serie de programas que funcionan en ese entorno, y que se puede cargar siempre que se desee, con la seguridad de que va a funcionar.

Además, por el propio funcionamiento de las herramientas de emulación de entornos de ejecución, se pueden simular diferentes entornos de ejecución desde la misma máquina. Así, en el caso comentado anteriormente de la práctica de servidor y cliente, desde el mismo equipo se pueden arrancar dos contenedores independientes (uno actuando como servidor y otro como cliente). Por tanto, no resulta necesario el uso de dos ordenadores diferentes para la realización de la práctica.

Por tanto, en el curso 1 se instaló Docker en todos los ordenadores del aula y se creó una imagen Docker que incluía todos los programas utilizados a lo largo de las prácticas de la asignatura. A dicha imagen se accedía a través de un usuario y contraseña. Para facilitar la labor del alumnado, en todos los ordenadores se realizó inicialmente el acceso a dicha imagen, por lo que el alumnado únicamente tenía que arrancar el contenedor y empezar a utilizarlo, ya que las credenciales se almacenaban.

En la primera sesión de prácticas, el docente explicó al alumnado el concepto de Docker y su utilización como herramienta base para la realización de las prácticas de la asignatura. Asimismo, en la guía de cada práctica se indicaban las instrucciones de Docker a utilizar.

La experiencia de utilización de Docker fue bastante positiva y, tras las dificultades propias de la primera toma de contacto con la herramienta en la primera práctica, los alumnos enseguida se acostumbraron al uso de la herramienta. Hay que tener en cuenta que se hizo un uso bastante básico de Docker (arrancar un Docker, pararlo y consultar los contenedores abiertos). Además, en la primera práctica el profesorado realizó una sesión bastante guiada de la práctica para acompañar a los alumnos en el proceso de aprendizaje.

El curso 2 (2019-2020) estuvo marcado por la pandemia causada por el COVID-19. Teniendo en cuenta que la asignatura objeto de estudio se impartió durante el segundo cuatrimestre, se dio la circunstancia de que parte de las prácticas se llevaron a cabo de forma presencial y otra parte de manera *qprkpg*.

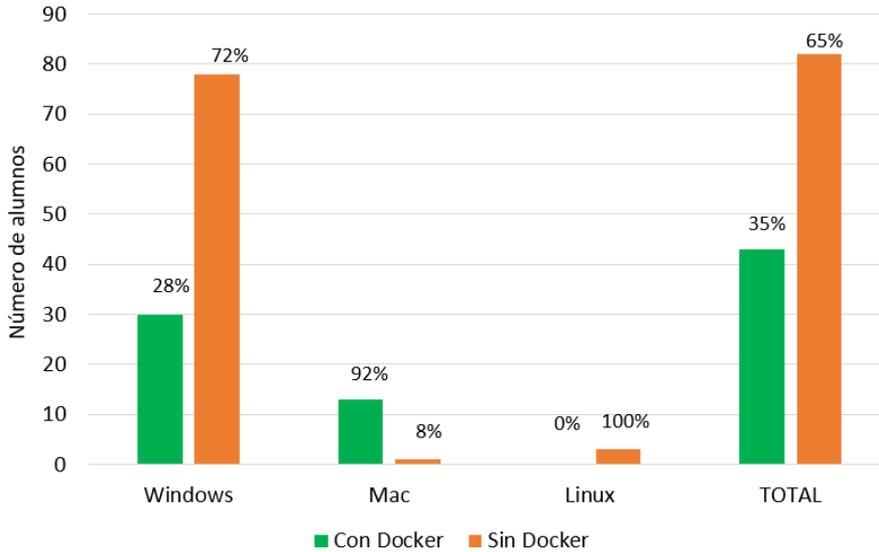
El transcurso de las prácticas presenciales fue similar al del curso anterior. La instalación de los programas a utilizar en las prácticas se realizó de una forma muy rápida gracias a la imagen Docker ya disponible en el curso anterior.

Por el contrario, la realización de prácticas de manera remota tuvo mayores dificultades para la mayoría de alumnado. Para facilitar la instalación de Docker por parte del alumnado, el profesorado realizó una guía en la que se explicaba de forma detallada el proceso para instalar Docker y arrancar el contenedor generado en los principales sistemas operativos: Windows, Mac OS y Linux. Cabe mencionar que de las dos prácticas que se realizaron de forma remota, únicamente una de ellas hacía uso de Docker. La realización de dicha práctica resultó problemática para algunos alumnos, que tuvieron dificultades de instalación de Docker pese a la guía proporcionada.

Una dificultad añadida resultó el hecho de que la instalación de Docker no resultaba compatible con algunos sistemas operativos si no se cumplían ciertas características (por ejemplo, permitir activar la virtualización). Como consecuencia, se facilitó a los alumnos otra posibilidad de realización de la práctica a través de la instalación de los programas necesarios, en vez de emplear Docker. Los alumnos que tuvieron mayores

problemas con la instalación fueron aquellos que disponían de equipos antiguos (en el caso de Windows) o sistemas operativos antiguos (tanto Windows como Mac).

Por lo que respecta a la distribución de sistemas operativos por parte del alumnado, de los 125 que realizaron la práctica, 108 (es decir, un 86,40 %) emplearon Windows, 14 (un 11,20 %) utilizaron Mac OS y únicamente 3 alumnos (un 2,40 %) utilizaron Linux (distribución Ubuntu). La Fig. 3 recoge los resultados de utilización de Docker por parte de los alumnos clasificado según el sistema operativo empleado.



Hli 05'F'kat klwek»p'f'g'c'no pcf q's'wg'wklk »'F qengt "q'pq'gp'hwpe»p'f'gr'itango c'qr gt c'kxq"

Como se aprecia en la gráfica, entre el alumnado que empleó Windows, un 28 % decidió utilizar Docker, mientras que casi tres cuartas partes prefirió instalarse todo el software necesario para poder ejecutar la práctica. En cambio, la mayoría de usuarios de Mac emplearon Docker, existiendo únicamente una persona que no utilizó Docker. Puede sorprender el caso de Linux, pero teniendo en cuenta que la imagen Docker utilizada está basada en Linux y que algunos de los programas utilizados en la práctica ya van instalados en Linux (o su instalación es bastante inmediata), el alumnado que utilizó Linux no consideró necesario emplear Docker para llevar a cabo la práctica.

Así, a nivel global, un 35 % del alumnado decidió emplear Docker para llevar a cabo la práctica. Este resultado se puede considerar bastante positivo, teniendo en cuenta que al inicio del curso el alumnado no había empleado anteriormente Docker (según respondieron a la pregunta formulada en clase), y que suele ser habitual el hecho de que el alumnado prefiera utilizar herramientas con las que está más familiarizado en detrimento de otras nuevas. Además, parte del alumnado no pudo emplear Docker por limitaciones de sus propios equipos, por lo que dicho porcentaje podría haber sido más elevado.

Por lo que respecta al profesorado, la experiencia se considera muy positiva, ya que el uso de Docker permite la generación de un entorno controlado en el que están ya instalados los programas necesarios para ser utilizados en la práctica.

A modo de resumen, la siguiente tabla muestra las principales ventajas e inconvenientes de cada una de las metodologías utilizadas en los diferentes cursos.

Curso	Ventajas	Inconvenientes
Curso 0 (2017-2018): sin Docker	<ul style="list-style-type: none"> El alumnado utiliza un entorno que les resulta más familiar 	<ul style="list-style-type: none"> Instalación de los programas Posibles problemas de compatibilidad de los programas por las actualizaciones Dificultad de realizar la práctica en casa
Curso 1 (2018-2019): Docker presencial	<ul style="list-style-type: none"> Compatibilidad de programas utilizados Facilidad de instalación de software utilizado Facilidad de continuar la práctica en casa por parte del alumnado 	<ul style="list-style-type: none"> Necesidad de crear una imagen Docker por parte del profesorado Nuevo entorno que no es familiar para el alumnado
Curso 2 (2019-2020): Docker online	<ul style="list-style-type: none"> Reutilización de la imagen Docker disponible Compatibilidad de programas utilizados Facilidad de instalación de software utilizado Facilidad de continuar la práctica en casa por parte del alumnado 	<ul style="list-style-type: none"> Nuevo entorno que no es familiar para el alumnado, por lo que el alumnado es más reticente a utilizarlo si no resulta necesario Problemas de utilización y compatibilidad de Docker en equipos del alumnado

Así, tras utilizar Docker en el entorno educativo, se detecta como ventaja principal, en primer lugar, el ahorro en tiempo y recursos que supone utilizar una única herramienta que ya incorpora todos los programas necesarios para la realización de las prácticas. Y lo más importante, que se dispone de la garantía de que los programas integrados en la imagen de Docker creada va a funcionar independientemente del sistema operativo utilizado y de las distintas versiones de sistemas operativos, ya que no hay problemas de compatibilidad con los programas necesarios para la práctica. Esto supone un claro beneficio para el docente de cara a la preparación de las prácticas.

Aunque es cierto que la experiencia de uso de Docker en la docencia online ha presentado problemas de compatibilidad del mismo Docker en algunos estudiantes debido al uso de equipos y sistemas operativos antiguos, en un futuro cercano dichos problemas desaparecerán. Además, la introducción de una nueva herramienta desconocida por los alumnos siempre supone una dificultad añadida. Para eso caso, la explicación del Docker por parte del profesor o incluso la realización de un taller o curso sobre la instalación y configuración de Docker puede representar una buena solución, ya que es una herramienta que se puede emplear de forma transversal en varias asignaturas de un grado tecnológico, además de en el mundo empresarial.

Para un entorno educativo online, Docker representa una solución ideal para el caso en el que se utilizan múltiples programas, como ha sido el caso de la asignatura bajo estudio. El hecho de que los estudiantes tengan que lidiar con la instalación de distintos softwares y los problemas de compatibilidad implícitos supone una dedicación temporal adicional que suele penalizar el tiempo dedicado por los estudiantes a lo

realmente importante como son los contenidos didácticos, como muestran las memorias presentadas de los estudiantes. De esta forma, con el uso de Docker los alumnos pueden aprovechar el tiempo de la práctica en lo realmente importante que es en trabajar los contenidos de la asignatura, no en estar llevando a cabo la instalación de programas. Esto supone una clara mejora en los resultados de aprendizaje. Además, al haber menos problemas referentes a la instalación y compatibilidad de programas, esto se traduce en menos dudas al respecto por parte de los alumnos al profesor. Por tanto, el profesor dispone de más tiempo que dedicar al alumnado para resolver dudas referentes a los contenidos de la asignatura, lo que también se traduce en una mejora del aprendizaje.

Es por ello por lo que, en los siguientes cursos, independientemente de que la modalidad de la docencia sea presencial u online, se seguirá utilizando Docker para la realización de las prácticas. Así, los resultados obtenidos en los sucesivos cursos completará el estudio presentado en este documento.

5. Conclusiones

Con los resultados obtenidos en este trabajo se puede concluir que el uso de herramientas para la emulación de entornos software aporta importantes ventajas en el desarrollo de asignaturas en las que el alumnado tenga que interactuar con aplicaciones software, especialmente aquellas relacionadas con el campo de las Telecomunicaciones y las Tecnologías de la Información.

La creación de entornos mediante contenedores Docker favorece la compatibilidad de los programas utilizados en la asignatura, facilita la instalación y permite una continuidad de la práctica entre el laboratorio de la universidad y el equipo del estudiante con la seguridad de mantener el mismo entorno de forma coherente.

Asimismo, la manera en la que se distribuyen las imágenes a través de un registro de imágenes Docker garantiza la compatibilidad en las versiones de las aplicaciones utilizadas y permite la actualización incremental y la reutilización de los contenedores para proporcionar nuevos entornos de ejecución.

Por otro lado, la introducción de Docker plantea un nuevo entorno para los alumnos, lo cual se debe tener en cuenta a la hora de realizar la guía docente e incluir actividades donde se aprenda su uso básico, para evitar cualquier fricción del alumno con el software, ya que es una de las motivaciones principales del presente trabajo.

Finalmente, sería interesante extender el uso de este tipo de entornos emulados a otras asignaturas afines a las de este trabajo, en las que el alumnado tenga que realizar también actividades relacionadas con la transmisión de vídeo, servidores de streaming, gestión de licencias y derechos digitales, uso de diferentes codificadores de vídeo o la medida de la calidad de experiencia de usuario, entre otras.

6. Referencias

- APACHE. Httpd server project. <<https://httpd.apache.org/>> [Consulta: 29 de marzo de 2021]
- BARROSO, J. y CABERO, J. (2015). "Nuevos retos en tecnología educativa". Editorial Síntesis.
- BORGES, Federico (2005). "La frustración del estudiante en línea. Causas y acciones preventivas". *Digithum*, n.º 7. DOI: <http://dx.doi.org/10.7238/d.v0i7.536>.
- CALERO, C. (2019). "La llegada de las nuevas tecnologías a la educación y sus implicaciones" en *International Journal of New Education*, n.º 4, pp. 21-39.

DÍAZ, J. (2018). “Aplicaciones del lenguaje de programación Python en la docencia de ingeniería de estructuras”, en Nuevos desafíos en la enseñanza superior, Educación Editora, pp. 653-657.

DOCKER. Empowering App Development for Developers: Docker. <<https://www.docker.com/>> [Consulta: 29 de marzo de 2021]

FFMPEG. A complete, cross-platform solution to record, convert and stream audio and video. <<https://www.ffmpeg.org/>> [Consulta: 29 de marzo de 2021]

PEÑA, R. (2015). “Python como primera aproximación a la programación”, en Revista de Investigación en Docencia Universitaria de la Informática (ReVisión), vol. 8, n.º 2, pp. 17-29.

ROSENDO, D. (2010). Las ciencias sociales y las nuevas tecnologías de la educación. Wanceulen Editorial.