**Universidad de las Américas**

**Facultad de Ingeniería y Ciencias Aplicadas**

**Ingeniería en Sistemas de Computación e Informática**

**Implementación de una guía de buenas prácticas de Integración Continua en proyectos de desarrollo en el área de tecnología de una Empresa Privada.**

Edison P. Mosquera León, Carlos X. Ruiz Águila, David F. Martínez Bravo

**DOCKER**

1. **Historia.**

En un principio, lo que ahora llamamos contenedores surgió como idea en los años 2000 de una tecnología que permite la partición de un sistema FreeBSD en varios subsistemas. Los cuales permitían desarrollos en entornos seguros que un administrador de sistemas podía compartir con diferentes usuarios dentro o fuera de una empresa. Cada vez se fueron combinando más tecnologías con mayor rapidez, creando grupos de control y limitación del uso de recursos para distintos procesos.

En 2008 es donde aparece docker en escena con su tecnología de contenedores, la cual fue incorporando diferentes series de conceptos y herramientas nuevas como, una línea de comandos sencilla de ejecutar y el diseño de imágenes nuevas en capas, un daemon de servidor, una biblioteca de imágenes en contenedores prediseñados y por último el concepto de un servidor de registros. Permitiendo que usuarios diseñaran rápidamente nuevos contenedores de capas, concediendo la interacción con otros usuarios sin dificultades.

Docker representa una evolución de la tecnología patentada de dotCLoud, la cual fue creación de Solomon Hykes por lo cual a su vez fue construida sobre proyectos de código abierto anteriores como lo fue Cloudlets.  
Por último, un análisis sobre esta tecnología mostro que las principales organizaciones contribuyentes de Docker son: Red Hat, Google, Cisco Systems y Amadeus TI Group.

1. **Descripción.**

Docker es una herramienta que permite el beneficio de desarrolladores, testers, administradores de sistemas que trabajan con entornos donde se ejecutan aplicaciones de software, procesos de despliegues, entre otros.

En la mayoría de las empresas el desarrollo, despliegues y entrega de software es un proceso con pasos bien diferenciados como lo son:

* Diseño de la aplicación.
* Desarrollo de la misma.
* Montar entornos de pruebas.
* Empaquetar la aplicación, desplegarla y entregarla.

Visto desde un punto de vista de desarrollo y despliegue los contenedores pueden realizar las mismas acciones que realizan máquinas virtuales, pero mejor.

Los contenedores no suponen un cambio en el diseño o construcción, ni permite la variación de la forma fundamental de pruebas, lo que propone es facilitar el hecho de mantener un entorno consistente. Además de ayudar a la hora de empaquetar y desplegar aplicaciones, permitiendo que los desarrolladores coloquen su aplicación dentro de un contenedor, el cual vendría a ser un entorno definido como portable.

1. **Características.**

* Instancias de docker ligeras, el proceso de montaje es sencillo a la hora de desplegar una aplicación.
* Contenedores consistentes, su consistencia se basa dentro del entorno de pruebas y el entorno de producción de una aplicación.
* Tiene contenedores gratuitos y de codigo abierto, docker es totalmente gratuito para que cualquiera lo descargue e instale, pero puedes pagar por ciertos servicios adicionales y soportes.

1. **Beneficios.**

* El retorno de la inversión y ahorro de costes, permitiendo a una empresa minimizar los costes de una solución reduciendo los recursos de infraestructura.
* Estandarización y productividad, la cual se enfoca a lo largo del proceso de desarrollo y despliegue de la aplicación permitiendo compilaciones, pruebas y entornos de producciones repetibles y replicables sin la necesidad de compartir gran cantidad de archivos.
* Compatibilidad y mantenimiento más fácil, esto permite beneficiar a todo el equipo de desarrollo permitiendo la ejecución en paridad, igualmente independiente del servidor en donde se ejecuta solucionando los problemas del entorno, configuración más fácil y una base de código más portable.
* Pruebas continuas, se garantizan entornos consistentes desde la fase de desarrollo hasta la de producción; asegurando que la configuración de sus contenedores de mantenga junto con sus dependencias internas con la seguridad de que no haya discrepancia ni intervención manual.

1. **Conclusiones.**

En conclusión, docker comparte sus sistemas operativos para ser ejecutados como procesos aislados independientemente del sistema operativos de la maquina HOST. Los contenedores docker pueden ser ejecutados en cualquier máquina, infraestructura y tecnologías de nube, permitiendo la portabilidad, flexibilidad y simplicidad a la hora de programar aplicaciones con sus instancias.