|  |  |
| --- | --- |
| **Práctica Build 1** | Entregable: Este documento y el histórico bash |
| Fecha Entrega: 19/2/20 |
| Equipo: Individual |
| Herramientas: IDE Android y Arduino IDE |
| Paula Estanga Blasco |

**Introducción**

La construcción de un entregable puede hacerse empleando un IDE pero en entornos con múltiples plataformas y diferentes equipos especializados esto no es recomendable y debe automatizarse con procesos independientes del IDE.

Acceso a máquina virtual: <https://eim-laboratoriovirtual.unavarra.es/laboratorio/#/>

1. Acceso con [alumno@ldap](mailto:alumno@ldap)
2. En cada contenedor: alumno/P6dJbyA2020

Para obtener histórico desde CLI: >$ *history*

**Desarrollo de la práctica**

1.-Instale en su contenedor dos IDE específicos:

Android Studio

Arduino IDE

2.-Prepare la automatización de dos *entregables*:

2.1.-Android

Crear proyecto con una actividad ActivityAxxxxx (donde xxxxx es el número de alumno) que compile. No es necesario añadir funcionalidad alguna.

Como *Android Studio* instala *Gradle*, buscar un *target* adecuado y ejecútelo desde CLI.

El histórico de ambos ejercicios esta guardado en un archivo: “ejercicio2.txt”

2.2.-Sensor basado en Arduino.

Proyecto referencia: https://github.com/ladislas/Bare-Arduino-Project

Según lo familiarizado que esté el alumno con el entorno, hay dos opciones:

1. Crear un sketch propio y compilarlo desde CLI con un *makefile* (recomendado ***Bare-Arduino-Project***).

b) Emplear el proyecto ***FooProject*** incluido en ***Bare-Arduino-Project*** siguiendo las instrucciones para compilar desde CLI.

3.- Comentar cómo afecta tener diferentes plataformas de compilación en un mismo proyecto y qué características debería tener un servicio SaaS que permitiera compilar aplicaciones para diferente hardware.

Tener diferentes plataforms almacenar versiones diferentes de las propiedades de solución y de proyecto que se aplican a distintas plataformas de destino. Por ejemplo, puede crear una configuración de depuración destinada a una plataforma x86 y una configuración de depuración destinada a una plataforma x64. Puede cambiar rápidamente la plataforma activa para poder compilar fácilmente varias configuraciones.

Resulta lógico que las empresas y desarrolladores necesiten recursos ágiles que les ayuden a acelerar los procesos de desarrollo y gestión de proyectos, y para solucionar estos problemas, una buena alternativa son los servicios “cloud” como SAAS.

La mejor manera de trabajar con aplicaciones en estos momentos viene sin duda de las plataformas de [cloud computing](http://www.cloudhispano.com/tag/cloud-computing/). Si un programa necesita unos recursos determinados, se puede aumentar la capacidad del servidor desde el propio software en la nube, añadiendo más RAM o más núcleos para el procesador según las necesidades del momento.

Los procesos de complicación de aplicaciones, dependiendo del programa que hayamos creado, pueden ser largos y complicados. La alternativa a los entornos cloud para compilar una aplicación es disponer de un servidor propio o alojamiento web con unas características determinadas.

Todos los programas bajo licencia exigen determinados requerimientos de sistema operativo, CPU, RAM... Aunque la mayoría de ordenadores pueden soportar un software desarrollado para uso común, no todos los programas funcionan con un buen rendimiento, o algunos sólo funcionan para determinados sistemas operativos.