

APÉNDICE B. MANUAL DE INSTALACIÓN

B.1 MANUAL DE INSTALACIÓN APLICACIÓN WEB

Los siguientes pasos son para instalar el entorno donde va a funcionar la aplicación Web, se utiliza Ubuntu 14.04 LTS y la instalación corresponde a un entorno llamado LAMP (Linux, Apache, MySQL y PHP).

Antes de iniciar la instalación del entorno se debe actualizar el sistema mediante el siguiente comando en la terminal y presione la tecla Enter: **sudo apt-get update**

B.1.1 Instalar Apache

Para instalar Apache se debe abrir la terminal e insertar el siguiente comando y presione la tecla Enter:

sudo apt-get install apache2

Luego de finalizar la instalación, abrir el navegador y escribir **http://localhost** para verificar que la instalación de Apache se realizó con éxito. La Figura B.1 indica algo similar a lo que debe aparecer en el navegador.

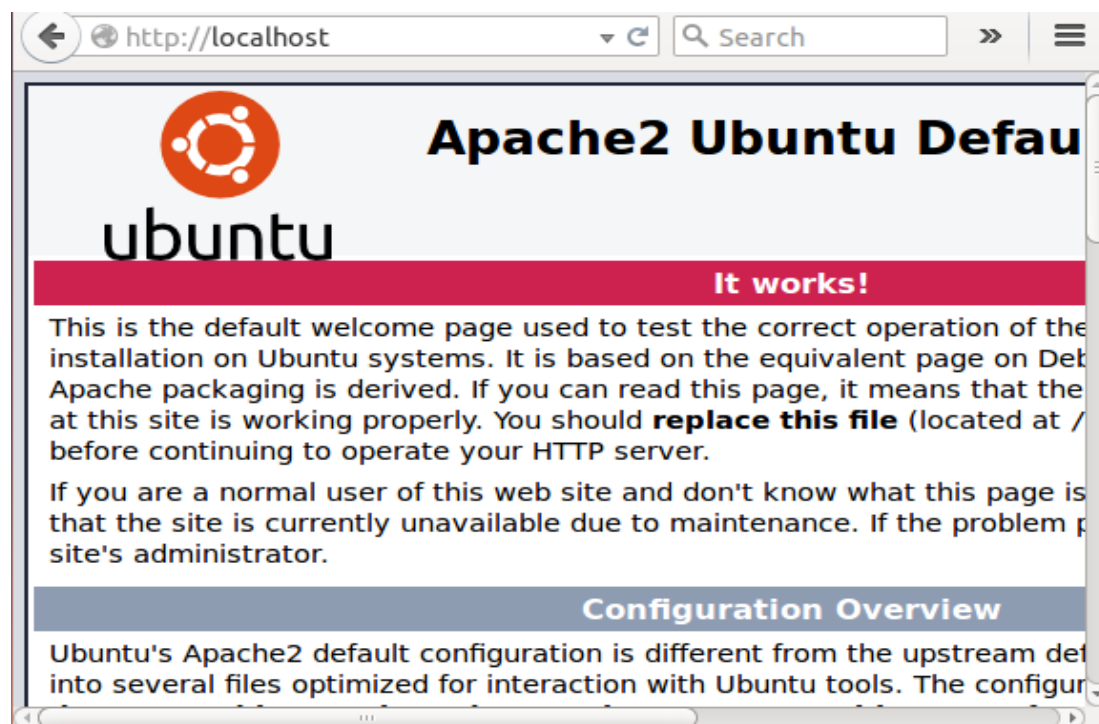


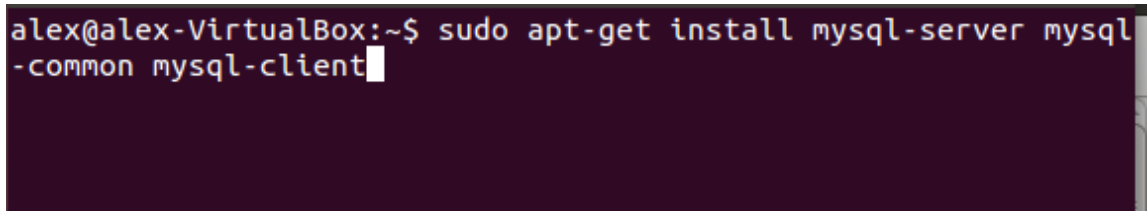
Figura B.1: Instalación Apache correcta

En caso que no aparezca lo indicado en la figura anterior, colocar el siguiente comando para reiniciar Apache y presione la tecla Enter: **sudo /etc/init.d/apache2 restart**

B.1.2 Instalar MySQL

El sistema ocupa una base de datos relacional MySQL, por lo que es necesario contar con este motor de base de datos. Para instalarlo colocar en la terminal el siguiente comando y presione la tecla Enter. Presionar la tecla **S** cuando pida la opción: **sudo apt-get install mysql-server mysql-common mysql-client**

La Figura B.2 muestra el comando indicado.

A terminal window with a dark purple background. The prompt is 'alex@alex-VirtualBox:~\$'. The command entered is 'sudo apt-get install mysql-server mysql-common mysql-client'. The cursor is at the end of the command.

```
alex@alex-VirtualBox:~$ sudo apt-get install mysql-server mysql-  
-common mysql-client
```

Figura B.2: Comando para instalar MySQL

Cuando se le indique debe colocar una contraseña para el usuario root (debe ser una contraseña que recuerde). Luego debe repetir la contraseña.

Luego de eso finaliza la instalación de MySQL. Para verificar que se instala correctamente debe ingresar el siguiente comando en la terminal y presionar Enter: **mysql -u root -p** e ingresar la contraseña. Debe aparecer algo similar a lo que indica la Figura B.3.

```
alex@alex-VirtualBox:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 42
Server version: 5.5.50-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

Figura B.3: Instalación de MySQL correcta

B.1.3 Instalar librerías de PHP para integración con Apache y MySQL

Para que se integre correctamente el servidor Apache con PHP y MySQL se deben instalar algunas librerías. Estas librerías se instalan con el siguiente comando en la terminal y presionar la tecla **S** cuando pida la opción:

```
sudo apt-get install php5 libapache2-mod-php5 php5-mysql
```

Luego de finalizar la instalación, debe reiniciar Apache colocando el siguiente commando y presionar Enter: **sudo /etc/init.d/apache2 restart**

Luego debe situarse en la carpeta **html** mediante la terminal de la siguiente manera y presionar Enter: **cd /var/www/html/**

Luego de estar situado en la carpeta **html** debe dar permiso de administrador a ésta, mediante el siguiente comando por la terminal y presionando Enter (debe reemplazar el nombre de usuario "alex" por el nombre de usuario correspondiente): **sudo chown alex:adm /var/www/html**

En la Figura B.4 se aprecia cómo debe aparecer el archivo **index.html** si se ha asignado correctamente permisos de administrador a la carpeta **html**.

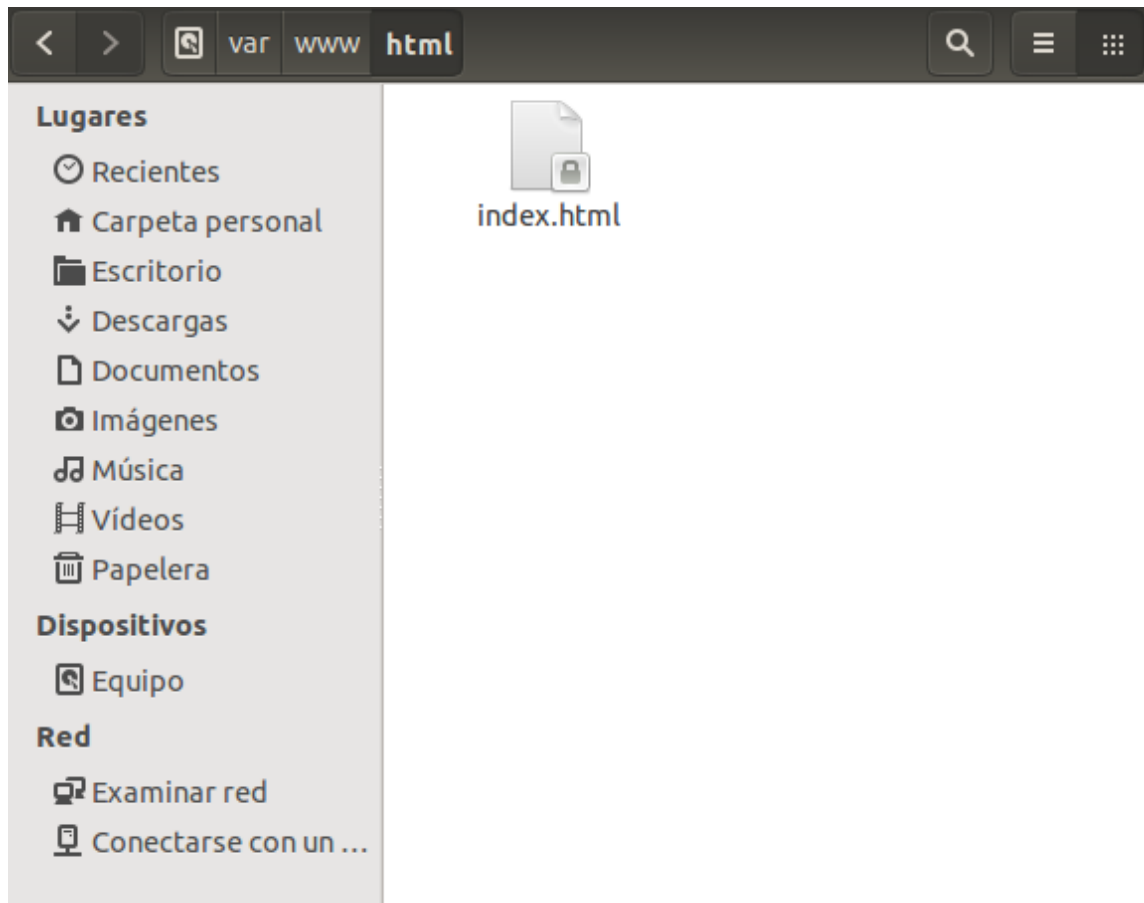


Figura B.4: Carpeta html con permisos de administrador

Es necesario dar permisos de administrador a esta carpeta, ya que en ésta se va a situar el archivo donde estará la aplicación (más adelante se aprecia este paso).

Luego debe cerrar la terminal y abrirla nuevamente para agregar la librería CURL de PHP, la cual es indispensable para que el sistema funcione. Debe colocar el siguiente comando en la terminal, presionar Enter y la tecla **S** cuando de la opción: **sudo apt-get install php5-curl**

Luego de finalizar la instalación debe reiniciar el servidor Apache con el siguiente comando en la terminal: **sudo /etc/init.d/apache2 restart**

B.1.4 Instalar phpMyAdmin

Para visualizar de mejor manera las bases de datos de MySQL y la base de dato del sistema, se procede a la instalación del entorno gráfico de este motor de base de datos llamado phpMyAdmin.

Cierre y abra nuevamente la terminal y coloque el siguiente comando, presione Enter y presione **S** cuando de la opción: **sudo apt-get install phpmyadmin**

Luego se le pide elegir el servidor para configurar automáticamente phpMyAdmin, debe elegir Apache2 y presionar Enter, tal como se aprecia en la Figura B.5.

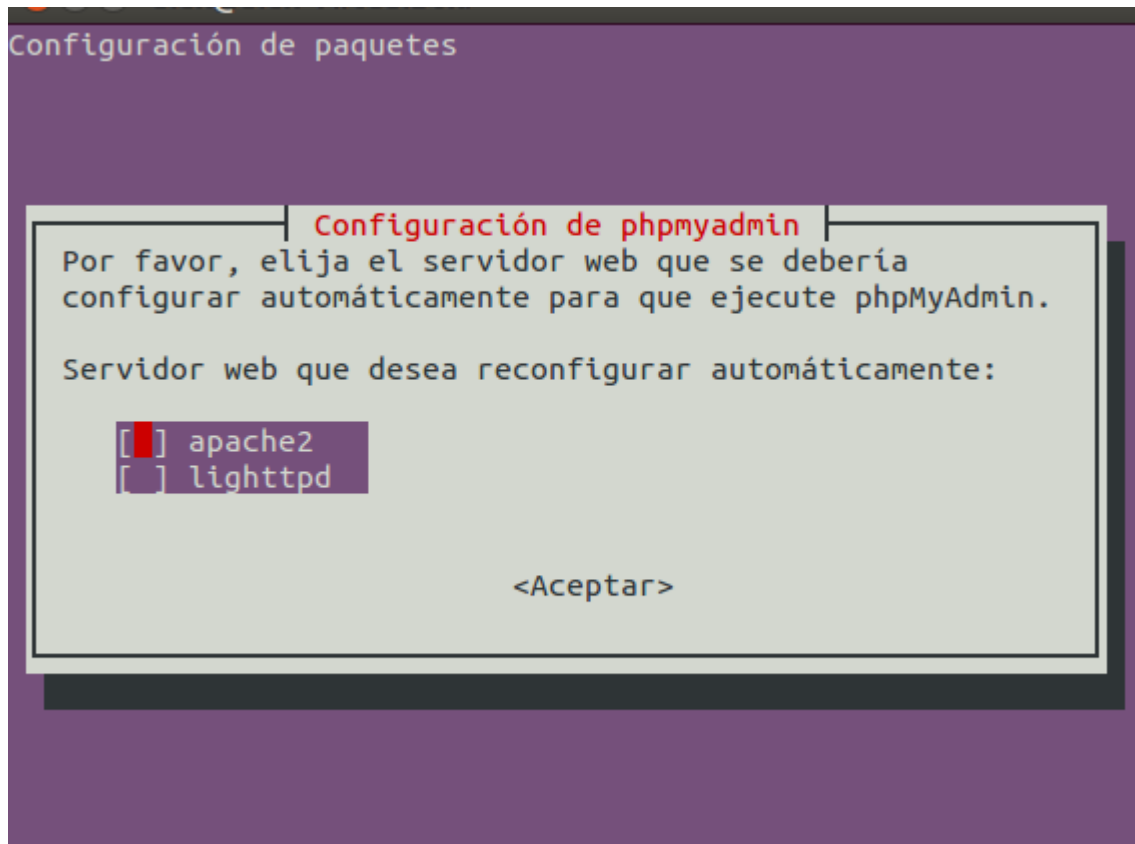


Figura B.5: Elección de Apache2 para phpMyAdmin

Luego se pide configurar phpMyAdmin, y debe dar la opción que **SI** y presionar Enter, tal como muestra la figura B.6.

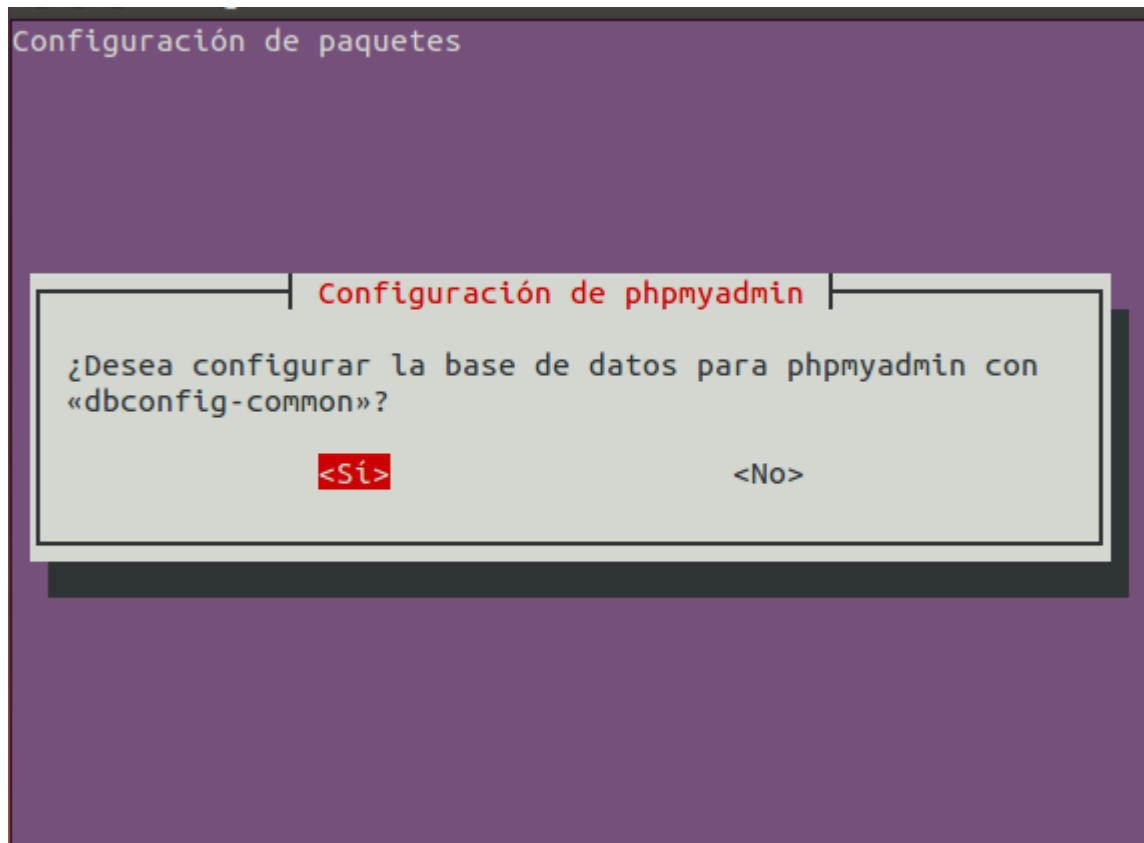


Figura B.6: Configurar phpMyAdmin

Luego debe ingresar la contraseña de root ingresada cuando se instaló MySQL, tal como se aprecia en la Figura B.7 y presionar Enter.

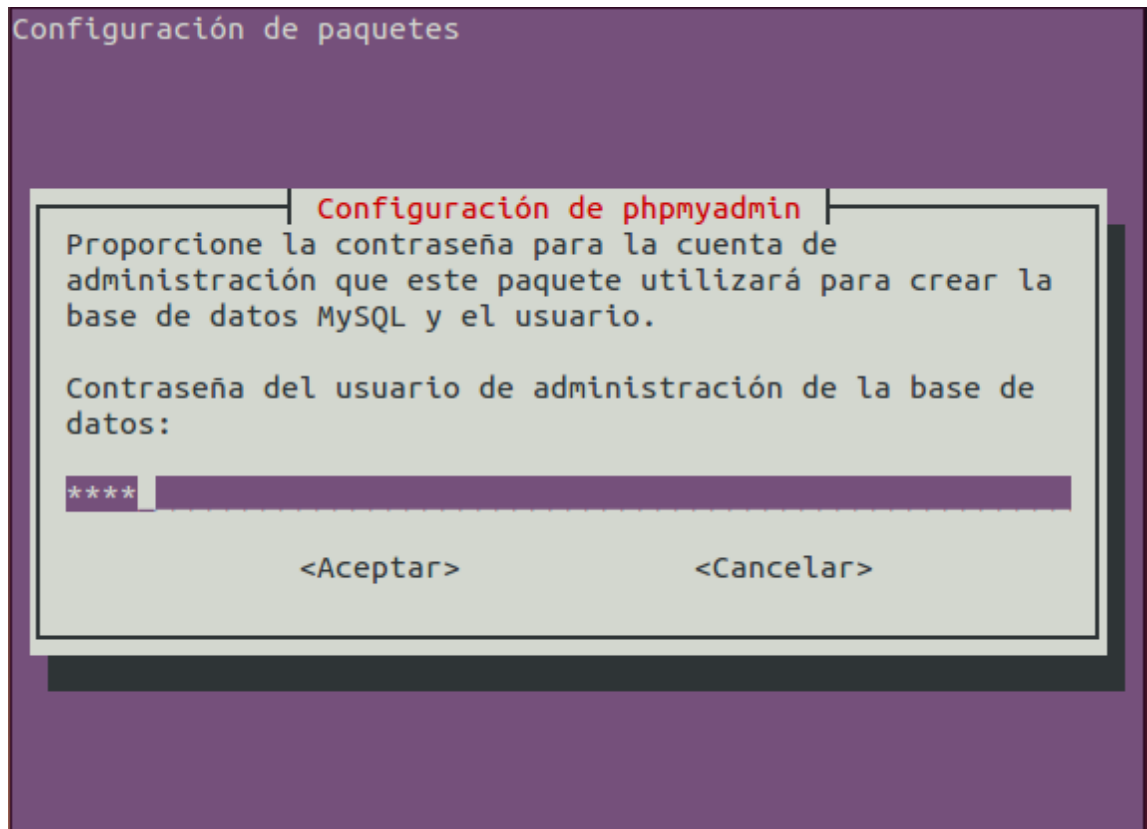


Figura B.7: Contraseña de usuario administrador de MySQL

Luego debe ingresar una contraseña para administrar phpMyAdmin (se sugiere colocar la misma que para MySQL por simpleza), tal como se aprecia en la Figura B.8 y presionar Enter.

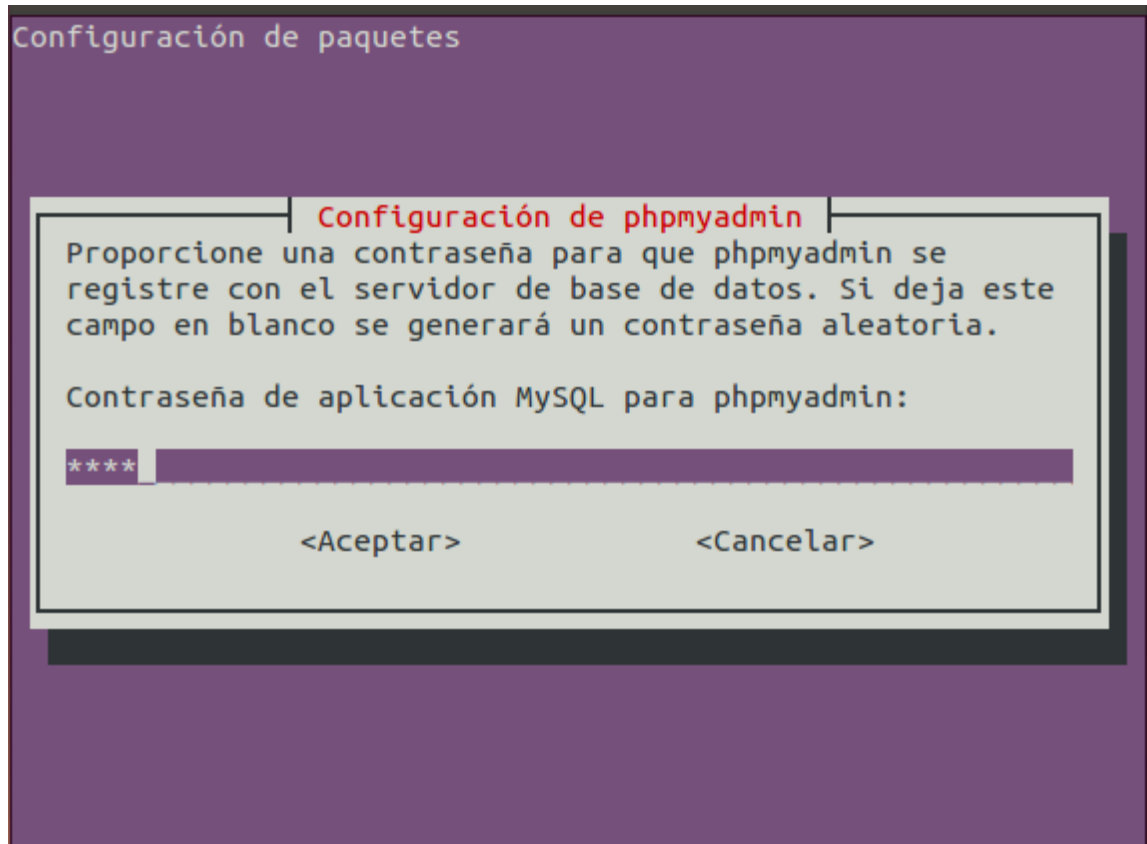


Figura B.8: Contraseña para ingresar a phpMyAdmin

Luego debe ingresar nuevamente la contraseña y presionar Enter.

Con esto ya debe tener instalado phpMyAdmin para gestionar la base de datos MySQL.

Para colocar phpMyAdmin en localhost (raíz del servidor) debe colocar el siguiente comando en la terminal (aunque no es necesario pero sirve para localizar de mejor manera phpMyAdmin y abrirlo desde el navegador mediante localhost): **ln -s /usr/share/phpmyadmin /var/www/html**

Luego para verificar que phpMyAdmin está correctamente instalado, debe abrir el navegador y escribir en la barra de direcciones lo siguiente: **http://localhost/phpmyadmin**

En la Figura B.9 se aprecia phpMyAdmin desde el navegador.

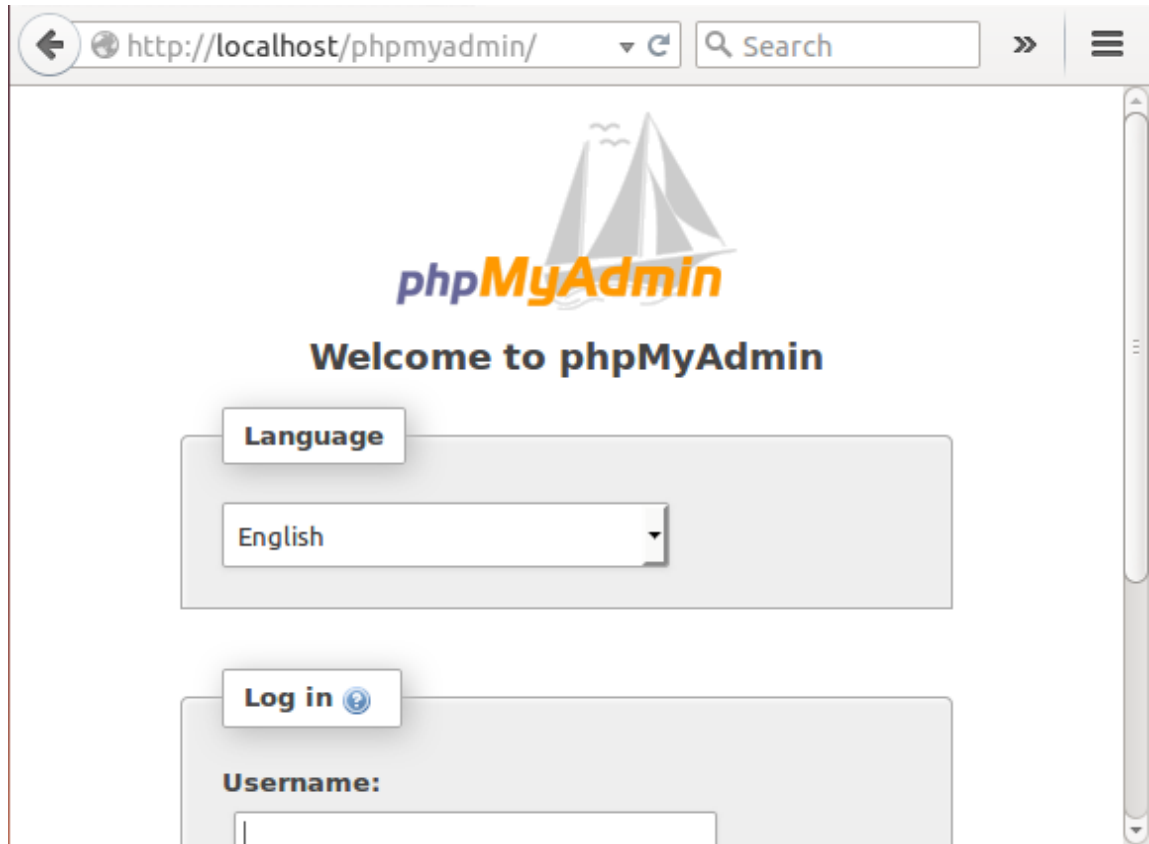


Figura B.9: phpMyAdmin correctamente instalado

B.1.5 Descargar proyecto desde GitHub

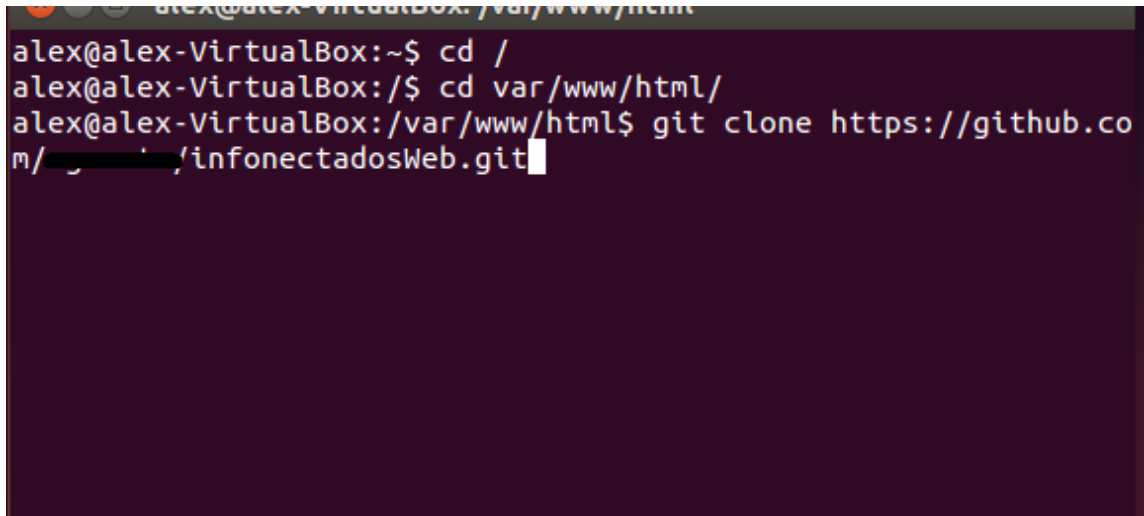
A continuación se dan los pasos para descargar el proyecto de la aplicación Web desde GitHub.

Lo primero es situarse en la carpeta principal desde la terminal, mediante el siguiente comando: **cd var/www/html** (estas carpetas se instalaron en los primeros pasos), tal como muestra la Figura B.10.

```
alex@alex-VirtualBox:~$ cd /  
alex@alex-VirtualBox:/$ cd var/www/html/
```

Figura B.10: Comando para situarse en la carpeta html

Luego debe clonar el repositorio para guardarlo en la carpeta **html** con el siguiente comando y luego presionar Enter: **git clone https://github.com/nombre_usuario/infonectadosWeb.git** donde **nombre_usuario** es su usuario de Github, tal como muestra la figura B.11.

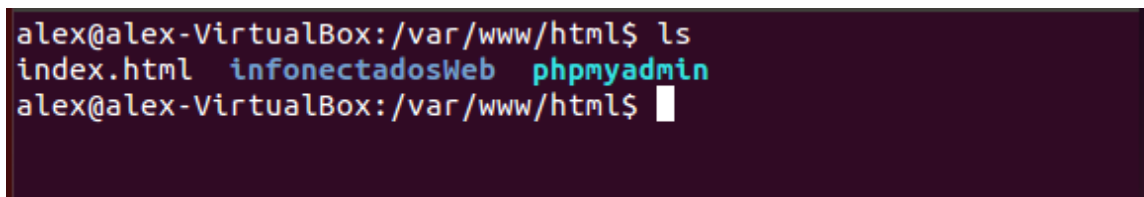
A terminal window with a dark purple background. The prompt is 'alex@alex-VirtualBox:~\$'. The user enters 'cd /', the prompt changes to 'alex@alex-VirtualBox:/\$'. The user enters 'cd var/www/html/', the prompt changes to 'alex@alex-VirtualBox:/var/www/html\$'. The user enters 'git clone https://github.com/alex-101/infonectadosWeb.git'.

```
alex@alex-VirtualBox:~$ cd /
alex@alex-VirtualBox:/$ cd var/www/html/
alex@alex-VirtualBox:/var/www/html$ git clone https://github.com/alex-101/infonectadosWeb.git
```

Figura B.11: Clonando proyecto desde Github

Luego pedirá ingresar su nombre de usuario y contraseña de Github.

Finalmente se descarga la carpeta llamada **infonectadosWeb** donde se encuentra el proyecto, tal como muestra la Figura B.12 mediante el comando **ls**.

A terminal window with a dark purple background. The prompt is 'alex@alex-VirtualBox:/var/www/html\$'. The user enters 'ls'. The output shows 'index.html', 'infonectadosWeb', and 'phpmyadmin' in different colors.

```
alex@alex-VirtualBox:/var/www/html$ ls
index.html  infonectadosWeb  phpmyadmin
alex@alex-VirtualBox:/var/www/html$
```

Figura B.12: Carpeta infonectadosWeb descargada exitosamente

B.1.6 Agregando la base de datos a phpMyAdmin

Para agregar la base de datos MySQL a phpMyAdmin debe seguir los siguientes pasos:

Ingresa a phpMyAdmin desde el navegador y sitúate en la sección “Base de datos” e ingresa un nombre a la base de datos que va a crear (se recomienda colocar **infonectados**, aunque puede ser cualquiera), luego presionar el botón “Crear”, tal como muestra la Figura B.13.

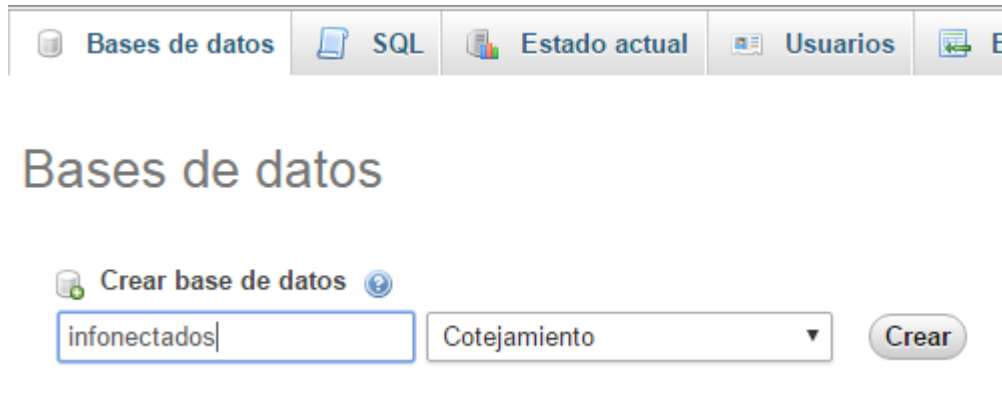


Figura B.13: Creando una base de datos desde phpMyAdmin

Luego debe situarse en la base de datos creada y en la parte superior presionar “Importar”, tal como indica la Figura B.14.

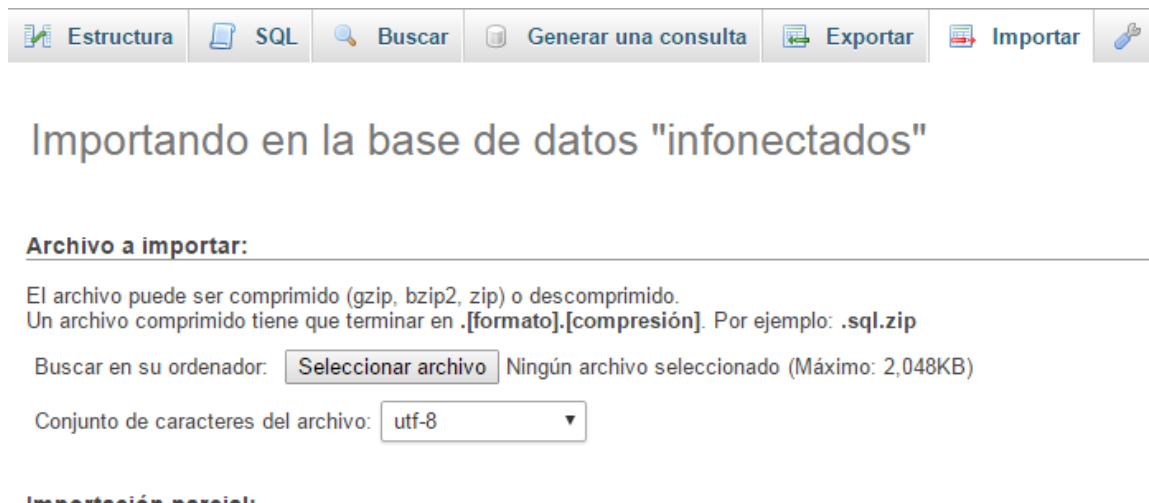


Figura B.14: Importando la base de datos desde phpMyAdmin

Luego debe buscar el archivo llamado **infonectados.sql** (Figura B.15) que se encuentra ubicado en la carpeta **html** donde se descargó el proyecto desde Github. La dirección donde se encuentra el archivo es **var/www/html/infonectadosWeb/** y presionar el botón “Continuar” que se encuentra en la parte inferior. Si todo sale bien aparece un mensaje indicando que se crearon las tablas de manera correcta.

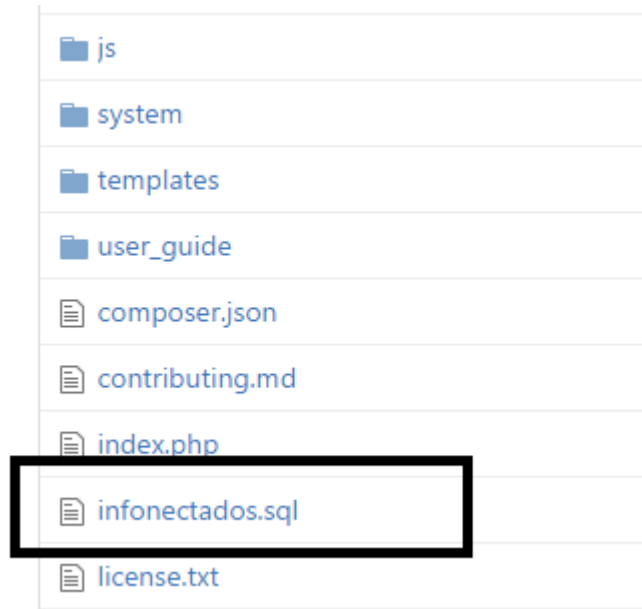


Figura B.15: Archivo infonectados.sql ubicado en la carpeta infonectadosWeb

B.1.7 Configurando el proyecto

Para que la aplicación funcione correctamente se deben realizar algunas configuraciones al proyecto. Lo primero es configurar la conexión de Codeigniter a la base de datos creada anteriormente, para esto se deben seguir los siguientes pasos.

Debe situarse en la carpeta **config** que se encuentra en la carpeta **infonectadosWeb** en la dirección **var/www/html/infonectadosWeb/application/config** y abrir el archivo llamado **database.php**.

Luego debe editar los campos correspondientes a la conexión a la base de datos, como **host**, **username**, **password y database**, donde host es el nombre del dominio del servidor (para este ejemplo es localhost, servidor local), username es el nombre de usuario de la base de datos de MySQL, password es la contraseña de la base de datos de MySQL y database es el nombre de la base de datos a utilizar (la cual se creó en el paso anterior, llamada para el ejemplo infonectados), tal como muestra la Figura B.16.

```
database.php x
| the query builder class.
*/
$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => '...',
    'password' => '...',
    'database' => '...',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
```

Figura B.16: Configurando conexión a la base de datos

Lo segundo que se debe configurar es la URL del servidor donde se harán las peticiones. Para ello debe editar la constante `URL_SERVIDOR` del archivo llamado **app.js** que se encuentra dentro de la carpeta **js** de **infonectadosWeb** en la dirección **var/www/html/infonectadosWeb/js**. Se debe colocar la URL del servidor donde estará alojado el sistema. La Figura B.17 muestra este paso.

```
app.js x
angular.module('app', ['ngRoute', 'ngResource', 'angular-md5'])

.constant('CONFIG', {
    URL_SERVIDOR: "http://localhost/infonectadosWeb/index.php"
    KEY_LOGIN: '...'
})
.run(function($rootScope, $location){
```

Figura B.17: Configurando URL del servidor

B.1.8 Ingresando a la aplicación

Luego de haber realizado todos los pasos descritos anteriormente, para verificar que el sistema funciona correctamente debe abrir el navegador e ingresar en la barra de direcciones la URL correspondiente (para el caso del ejemplo es **http://localhost/infonectadosWeb**) pudiendo ver la página principal de la aplicación, tal como muestra la Figura B.18.

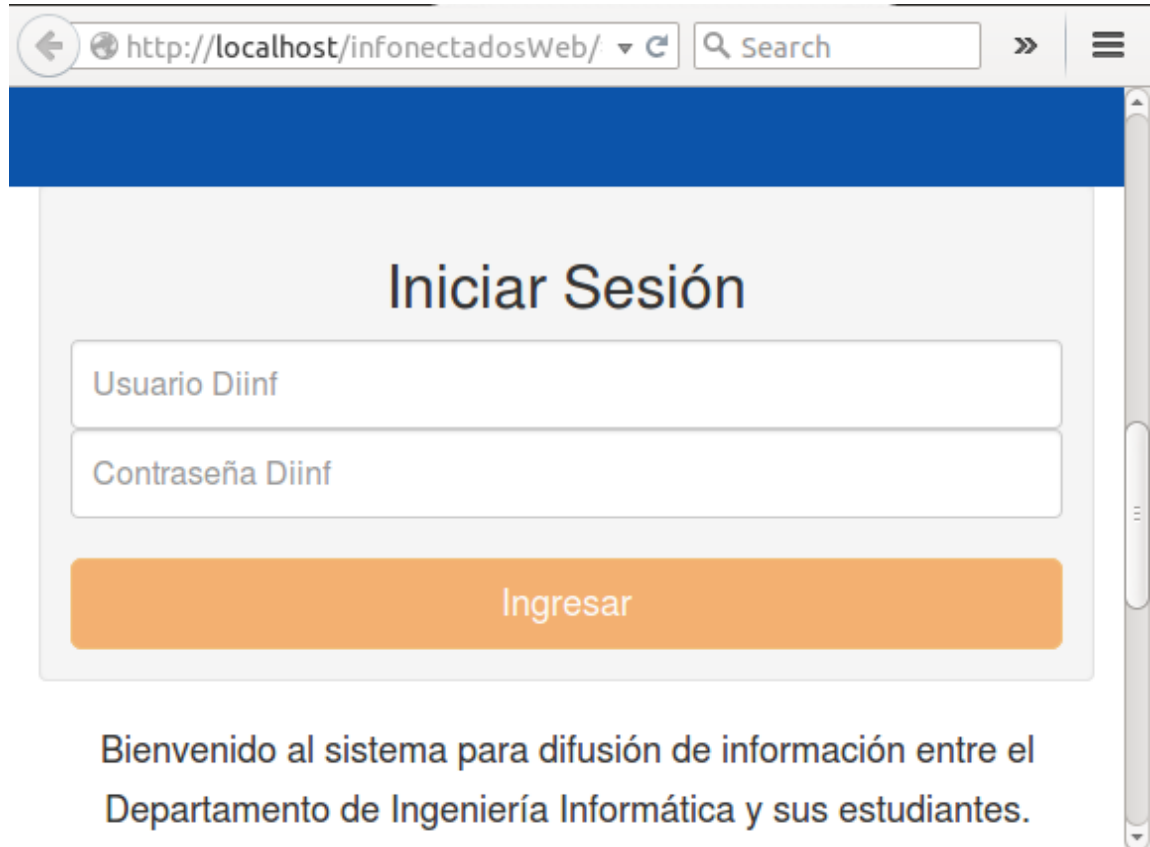


Figura B.18: Página principal de infonectadosWeb

B.2 MANUAL DE INSTALACIÓN APLICACIÓN MÓVIL

La aplicación móvil es desarrollada en Cordova para dispositivos móviles Android, por lo que para compilarla y generar el apk para su posterior publicación en Google Play se deben instalar ciertas dependencias, como Java, node.js, Ionic y el SDK de Android.

B.2.1 Instalando dependencias

Para instalar todas las dependencias mencionadas anteriormente, se debe descargar el archivo `instalador.sh` desde el siguiente link y descomprimir el archivo descargado:

https://github.com/agarate/instalador_ionic/archive/master.zip

Posteriormente debe abrir la terminal y situarse donde se encuentra el archivo **instalador.sh** descomprimido anteriormente y colocar el siguiente comando **sudo ./instalador.sh** y comenzará la instalación.

Luego de finalizar la instalación debe reiniciar el sistema.

Para verificar que está todo instalado debe colocar los siguientes comandos en la terminal:

java -version: verifica que se instaló Java correctamente

node -v: verifica que se instaló nodejs correctamente

ionic -v: verifica que se instaló Ionic correctamente

Luego debe instalar algunas APIs necesarias para que el SDK de Android funcione de manera correcta, para eso debe abrir una terminal e ingresar el siguiente comando y presionar Enter, tal como indica la Figura B.19: **/opt/android-sdk/tools/android**

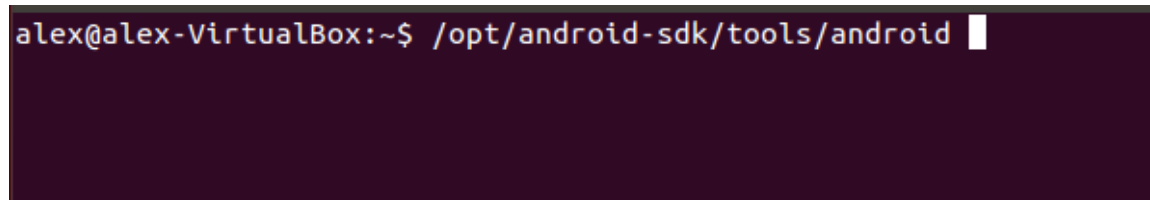


Figura B.19: Abrir panel de APIs de SDK de Android

Luego se abrirá un panel donde aparecen todas las APIs para el SDK de Android, en ésta se deben elegir los paquetes que se indican en la Figura B.20 y B.21 para posteriormente descargarlas e instalarlas apretando el botón **“Install packages...”**.

SDK Path: /opt/android-sdk

Packages

Name	API	Rev.	Status
<input checked="" type="checkbox"/> Tools			
<input checked="" type="checkbox"/> Tools (Preview Channel)			
<input checked="" type="checkbox"/> Android 7.0 (API 24)			
<input checked="" type="checkbox"/> Android 6.0 (API 23)			
<input type="checkbox"/> Android 5.1.1 (API 22)			
<input type="checkbox"/> Android 5.0.1 (API 21)			
<input type="checkbox"/> Android 4.4W.2 (API 20)			
<input checked="" type="checkbox"/> Android 4.4.2 (API 19)			
<input type="checkbox"/> Android 4.3.1 (API 18)			
<input type="checkbox"/> Android 4.2.2 (API 17)			
<input type="checkbox"/> Android 4.1.2 (API 16)			
<input type="checkbox"/> Android 4.0.3 (API 15)			
<input type="checkbox"/> Android 4.0 (API 14)			
<input type="checkbox"/> Android 3.2 (API 13)			
<input type="checkbox"/> Android 3.1 (API 12)			
<input type="checkbox"/> Android 3.0 (API 11)			
<input type="checkbox"/> Android 2.3.3 (API 10)			
<input type="checkbox"/> Android 2.3.1 (API 9)			

Figura B.20: Paquetes del SDK a instalar (parte 1)

Name	API	Rev.	Status
<input type="checkbox"/> Android 2.3.3 (API 10)			
<input type="checkbox"/> Android 2.3.1 (API 9)			
<input type="checkbox"/> Android 2.2 (API 8)			
<input type="checkbox"/> Android 2.1 (API 7)			
<input type="checkbox"/> Extras			
<input type="checkbox"/> GPU Debugging tools		3.1	<input type="checkbox"/> Not installed
<input type="checkbox"/> GPU Debugging tools		1.0.3	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Android Support Repository		35	<input type="checkbox"/> Not installed
<input type="checkbox"/> Android Auto Desktop Head Unit emulator		1.1	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Google Play services		32	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Google Repository		32	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Google Play APK Expansion library		1	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Google Play Licensing Library		1	<input type="checkbox"/> Not installed
<input checked="" type="checkbox"/> Google Play Billing Library		5	<input type="checkbox"/> Not installed
<input type="checkbox"/> Android Auto API Simulators		1	<input type="checkbox"/> Not installed
<input type="checkbox"/> Google USB Driver		11	<input checked="" type="checkbox"/> Not compatible with Linux
<input type="checkbox"/> Google Web Driver		2	<input type="checkbox"/> Not installed
<input type="checkbox"/> Intel x86 Emulator Accelerator (HAXM inste		6.0.3	<input checked="" type="checkbox"/> Not compatible with Linux

Show: ☒ Updates/New ☒ Installed Select [New](#) or [Updates](#) Install 43 packages...

Figura B.21: Paquetes del SDK a instalar (parte 2)

La descarga de todos esos paquetes de demoran, por lo que se debe tener paciencia.

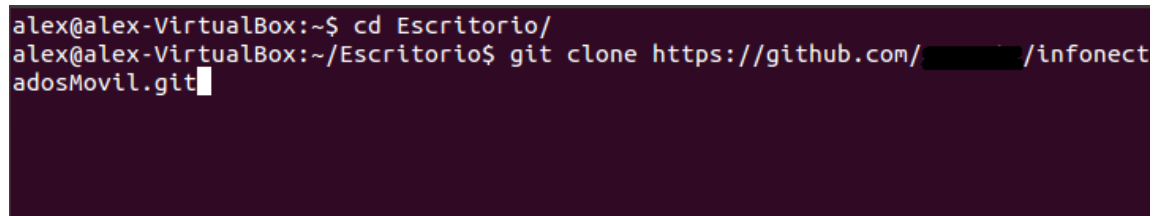
B.2.2 Descargando proyecto móvil desde Github

Luego de tener todo instalado correctamente, se procede a descargar el proyecto **infonectadosMovil** desde Github.

Primero debe abrir la terminal y situarse en algún directorio (por ejemplo el Escritorio) con el comando **cd Escritorio**.

Luego clonar el repositorio mediante el siguiente comando:

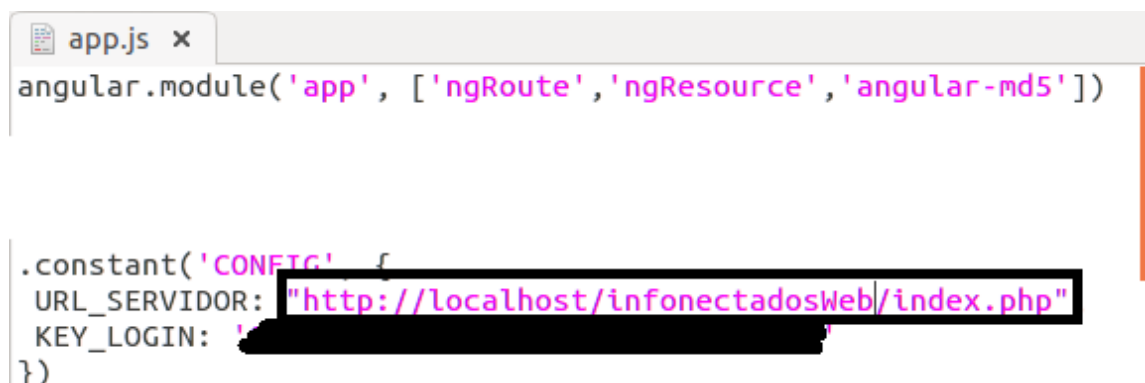
git clone https://github.com/nombre_usuario/infonectadosMovil.git donde **nombre_usuario** es su usuario de Github, tal como muestra la figura B.22 y colocar su nombre de usuario y contraseña.



```
alex@alex-VirtualBox:~$ cd Escritorio/  
alex@alex-VirtualBox:~/Escritorio$ git clone https://github.com/[redacted]/infonectadosMovil.git
```

Figura B.22: Clonar proyecto infonectadosMovil

Luego se debe configurar la URL del servidor donde se harán las peticiones. Para ello debe editar la constante **URL_SERVIDOR** del archivo llamado **app.js** que se encuentra dentro de la carpeta **js** de **infonectadosMovil** en la dirección para el ejemplo sería **Escritorio/infonectadosMovil/js**. Se debe colocar la URL del servidor donde está alojado el sistema Web (**infonectadosWeb**). La Figura B.23 muestra este paso.



```
app.js x  
angular.module('app', ['ngRoute', 'ngResource', 'angular-md5'])  
  
.constant('CONFIG', {  
  URL_SERVIDOR: "http://localhost/infonectadosWeb/index.php"  
  KEY_LOGIN: "[redacted]"  
})
```

Figura B.23: Configurando URL para realizar peticiones al servidor

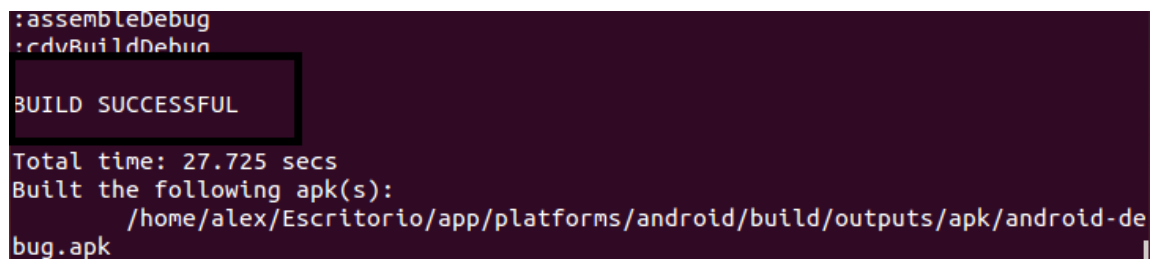
Para verificar que se puede compilar la aplicación exitosamente, se debe ingresar al directorio donde se encuentra el proyecto clonado, lo cual para el ejemplo sería:

cd Escritorio/infonectadosMovil

Dentro del proyecto infonectadosMovil se debe ingresar el siguiente comando en la terminal:

ionic build android

Y si está todo bien debe aparecer el mensaje “**BUILD SUCCESSFUL**”, tal como muestra la Figura B.24. En caso contrario, significa que faltan paquetes SDK que instalar o algún otro problema.

A terminal window with a dark background and light-colored text. The text shows the execution of 'ionic build android' and the resulting output. The 'BUILD SUCCESSFUL' message is highlighted with a white rectangular box. Below it, the total time and the path to the generated APK file are displayed.

```
:assembleDebug
:cdvBuildDebug
BUILD SUCCESSFUL
Total time: 27.725 secs
Built the following apk(s):
   /home/alex/Escritorio/app/platforms/android/build/outputs/apk/android-de
bug.apk
```

Figura B.24: Aplicación compilada exitosamente

B.2.3 Generando APK de infonectadosMovil

Lo primero que se debe realizar es eliminar los plugins que no son necesarios, para eso debe abrir la terminal, situarse en el proyecto infonectadosMovil e ingresar el siguiente comando:

cordova plugin rm cordova-plugin-console

Luego se debe colocar el siguiente comando: **cordova build --release android** con el cual se genera una versión de producción llamada **infonectados-release-unsigned.apk** (o similar) que se encuentra según el ejemplo en el directorio **/home/alex/Escritorio/infonectadosMovil/platforms/android/build/outputs/apk/infonectados-release-unsigned.apk** siempre y cuando se haya hecho la compilación satisfactoriamente apareciendo el mensaje “**BUILD SUCCESSFUL**”.

Luego se necesita generar una clave a la aplicación para poder subirla posteriormente a Google Play, por lo que se debe abrir una nueva terminal, situarse en el escritorio e ingresar el siguiente comando:

```
keytool -genkey -v -keystore my-release-key.keystore -alias alias_name -keyalg RSA -  
keysize 2048 -validity 10000
```

donde **my-release-key** es algún nombre que quiera darle a la clave privada y **alias_name** es su nombre que quiera dar (colocar mismo nombre tanto en **my-release-key** como en **alias_name** para mayor simplicidad). Además, debe ingresar los datos que se solicitan como una contraseña, ciudad y otros tipos de preguntas, y para terminar debe colocar **si** o **yes**. Finalmente se genera la clave llamada **algun_nombre.keystore** en el escritorio (o en el lugar donde se situó en la terminal anteriormente). Esta clave debe guardarla en algún lugar seguro, ya que en caso que se pierda no se podrán realizar actualizaciones a la aplicación.

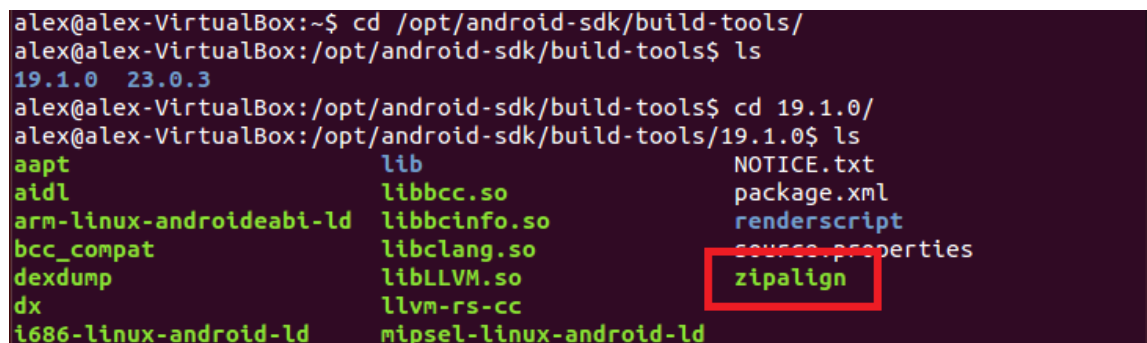
Luego hay que situarse en el directorio donde se encuentra el apk **infonectados-release-unsigned.apk** generado anterior a la generación de la clave e ingresar el siguiente comando:

```
jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore  
algun_nombre.keystore infonectados-release-unsigned.apk algun_nombre
```

donde **algun_nombre** es el nombre de la clave generada con anterioridad e **infonectados-release-unsigned.apk** es el apk generado.

Se pedirá que ingrese su contraseña, ésta es la que se introdujo cuando se genera la clave para el archivo **algun_nombre.keystore** en los pasos anteriores.

Luego se debe buscar la carpeta donde se encuentre un archivo llamado **zipalign** que viene con los paquetes SDK de Android. Puede estar en la carpeta **/opt/android-sdk/build-tools/VERSION** donde VERSION es el número de algún paquete, como por ejemplo 19.1.0, 20.1.0, etc, tal como muestra la Figura B.25.



```
alex@alex-VirtualBox:~$ cd /opt/android-sdk/build-tools/  
alex@alex-VirtualBox:/opt/android-sdk/build-tools$ ls  
19.1.0  23.0.3  
alex@alex-VirtualBox:/opt/android-sdk/build-tools$ cd 19.1.0/  
alex@alex-VirtualBox:/opt/android-sdk/build-tools/19.1.0$ ls  
aapt          lib          NOTICE.txt  
aidl          libbcc.so    package.xml  
arm-linux-androideabi-ld libbccinfo.so renderscript  
bcc_compat    libclang.so  source.properties  
dexdump       libLLVM.so   zipalign  
dx            llvm-rs-cc  
i686-linux-android-ld mipsel-linux-android-ld
```

Figura B.25: Carpeta que contiene el archivo zipalign

Se debe buscar en alguno de esos paquetes el archivo **zipalign**. En caso que no se encuentre, debe instalar algún paquete extra, por lo que debe ingresar por la terminal a **/opt/android-sdk/tools/android** presionar Enter e instalar los paquetes que aparecen como no instalados, como en el ejemplo de la Figura B.26.

SDK Path: /opt/android-sdk

Packages

Name	API	Rev.	Status
<input checked="" type="checkbox"/> Tools			
<input checked="" type="checkbox"/> Tools (Preview Channel)			
<input checked="" type="checkbox"/> Android 7.0 (API 24)			
<input checked="" type="checkbox"/> Android 6.0 (API 23)			
<input type="checkbox"/> Android 5.1.1 (API 22)			
<input type="checkbox"/> Android 5.0.1 (API 21)			
<input type="checkbox"/> Android 4.4W.2 (API 20)			
<input checked="" type="checkbox"/> Android 4.4.2 (API 19)			
<input type="checkbox"/> Android 4.3.1 (API 18)			
<input type="checkbox"/> Android 4.2.2 (API 17)			
<input type="checkbox"/> Android 4.1.2 (API 16)			
<input type="checkbox"/> Android 4.0.3 (API 15)			
<input type="checkbox"/> Android 4.0 (API 14)			
<input type="checkbox"/> Android 3.2 (API 13)			
<input type="checkbox"/> Android 3.1 (API 12)			
<input type="checkbox"/> Android 3.0 (API 11)			
<input type="checkbox"/> Android 2.3.3 (API 10)			
<input type="checkbox"/> Android 2.3.1 (API 9)			

Figura B.25: Paquetes SDK de Android

Luego de situarse en la carpeta que contiene el archivo **zipalign** desde la terminal, se debe ingresar el siguiente comando: **zipalign -v 4 infonectados-release-unsigned.apk Nombre_aplicacion.apk** donde **Nombre_aplicacion** es el nombre que se quiere dar a la aplicación (el cual sería **Infonectados.apk**) e **infonectados-release-unsigned.apk** es el apk que se generó en los pasos anteriores y presionar Enter.

Finalmente se genera el apk final que debe ser subido a la tienda de aplicaciones de Google Play.