



DDNE
Inventory

DDNE Inventory

Tabla de contenido

Tabla de Ilustraciones.....	3
1. Configuración de creación Máquina Virtual.....	4
2. Configuración de puertos y llaves de acceso.....	5
3. Confirmación de creación.....	5
4. Conectar.....	6
6. GIT.....	7
7.NodeJS - Backend	7
8.React - Frontend	8
9.NodeJS Dependencias	8
10. ReactJS Depencias	9



DDNE
Inventory

Tabla de Ilustraciones

ILUSTRACIÓN 1 - CONFIGURACIÓN MÁQUINA VIRTUAL.....	4
ILUSTRACIÓN 2 - CONFIGURACIÓN LLAVES Y PUERTOS.....	5
ILUSTRACIÓN 3 - CONFIRMACIÓN DE CONFIGURACIÓN.....	5
ILUSTRACIÓN 4 - INSTALACIÓN MYSQL.....	6
ILUSTRACIÓN 5 - INSTALACIÓN GIT	7
ILUSTRACIÓN 6 - CLONAR REPOSITORIO NODEJS	7
ILUSTRACIÓN 7 - CLONAR REPOSITORIO REACT	8
ILUSTRACIÓN 8 - INSTALAR DEPENDENCIAS NODEJS.....	8
ILUSTRACIÓN 9 - INSTALAR DEPENDENCIAS REACT.....	9



DDNE
Inventory

1. Configuración de creación Máquina Virtual

En este recuadro se debe elegir las configuraciones básicas de la máquina virtual, como detalles de la suscripción de Azure, como la ubicación del servidor (País/Región).

Ilustración 1 - configuración máquina virtual

Crear configuración de máquina virtual

Crear una máquina virtual

Detalles del proyecto

Seleccione la suscripción para administrar recursos implementados y los costes. Use los grupos de recursos como carpetas para organizar y administrar todos los recursos.

Suscripción *

Azure for Students

Grupo de recursos *

DDNE

Crear nuevo

Detalles de instancia

Nombre de máquina virtual *

ddne

Región *

(US) West US 3

Opciones de disponibilidad

Zona de disponibilidad

Zona de disponibilidad *

Zona 1

Ahora puede seleccionar varias zonas. Si selecciona varias zonas, se creará una VM por zona. [Más información](#)

Tipo de seguridad

Máquinas virtuales de inicio seguro

Configurar características de seguridad

Imagen *

Ubuntu Server 22.04 LTS - x64 gen. 2

Ver todas las imágenes | Configurar la generación de máquinas virtuales

Arquitectura de VM

Arm64

☒ x64

Ejecución de Azure Spot con descuento

☐

Tamaño *

Standard_DS2_v2_Promo - 2 vcpu, 7 GiB de memoria (Precio no disponible)

Ver todos los tamaños

Habilitar hibernación

☐

Actualmente, Hibernar no admite el inicio de confianza y las máquinas virtuales confidenciales para imágenes de Linux. [Más información](#)

Cuenta de administrador

Tipo de autenticación

☒ Clave pública SSH

☐ Contraseña

Ahora, Azure genera automáticamente un par de claves SSH y le permite almacenarlo para usarlo en el futuro. Es una forma rápida, sencilla y segura de conectarse a la máquina virtual.

Nota: Imagen generada de manera local (Equipo DDNE – SENA ADSO 2617551)

2. Configuración de puertos y llaves de acceso

En este formulario se piden las credenciales y puertos que van a estar configurados en la máquina virtual.

Ilustración 2 - configuración llaves y puertos

Nombre de usuario *	azureuser ✓
Origen de clave pública SSH	Generar un par de claves nuevo ✓
Tipo de clave SSH	<input checked="" type="radio"/> Formato RSA SSH <input type="radio"/> Formato Ed25519 SSH <p>Ed25519 ofrece mejor rendimiento y seguridad con un tamaño de clave más pequeño, mientras que RSA todavía se usa ampliamente, especialmente para sistemas y aplicaciones heredados.</p>
Nombre de par de claves *	ddne_key ✓
Reglas de puerto de entrada Seleccione los puertos de red de máquina virtual que son accesibles desde la red Internet pública. Puede especificar acceso de red más limitado o granular en la pestaña Red.	
Puertos de entrada públicos *	<input type="radio"/> Ninguno <input checked="" type="radio"/> Permitir los puertos seleccionados
Seleccionar puertos de entrada *	HTTP (80), SSH (22) ✓

i Se bloquea todo el tráfico de Internet de forma predeterminada. Puede cambiar las reglas del puerto de entrada en la página VM > Redes.

Nota: Imagen generada de manera local (Equipo DDNE – SENA ADSO 2617551)


3. Confirmación de creación



En este recuadro confirma la creación de la máquina virtual con todas las configuraciones realizadas, donde se darán a mostrar en pantalla los atributos más importantes de la misma, donde daremos *clic* en ir al recurso (Enmarcado en un recuadro rojo) para seguir con el flujo normal.

Ilustración 3 - confirmación de configuración

Recuadro de confirmación creación máquina virtual

[Eliminar](#)
[Cancelar](#)
[Volver a implementar](#)
[Descargar](#)
[Actualizar](#)

 **Se completó la implementación**


 Nombre de implementación: CreateVm-canonical.0001-com-ubunt...
 Hora de inicio: 21/6/2024, 13:26:43
 Suscripción: [Azure for Students](#)
 Id. de correlación: 0b36f22c-bf54-471a-989a-37d3038b3106 
 Grupo de recursos: [DDNE](#)

[▼ Detalles de implementación](#)

[^ Pasos siguientes](#)

[Configurar el apagado automático](#) Recomendado
[Supervisar el estado, el rendimiento y las dependencias de red de la máquina virtual](#) Recomendado
[Ejecutar un script dentro de la máquina virtual](#) Recomendado

[Ir al recurso](#)
[Crear otra VM](#)

Enviar comentarios
[Cuéntenos su experiencia con la implementación](#)

Nota: Imagen generada de manera local (Equipo DDNE – SENA ADSO 2617551)

4. Conectar

Al dar clic en *conectar* aparecerá la consola donde podremos instalar todo lo relacionado con el proyecto de formación.

5. MySQL

Ejecutar en la consola ***sudo apt-get install MySQL-server***, para instalar MySQL server.

Ilustración 4 - instalación MySQL

Comando de instalación MySQL

```
jhoan.pea01@soy.sena.edu.co@ddne:~$ sudo apt-get install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Some packages could not be installed. This may mean that you have
requested an impossible situation or if you are using the unstable
distribution that some required packages have not yet been created
or been moved out of Incoming.
The following information may help to resolve the situation:
```

Nota: Imagen generada de manera local (Equipo DDNE – SENA ADSO 2617551)

6. GIT

Ejecutar en la consola `sudo apt-get install git` para instalar git.

Ilustración 5 - instalación git

Comando de instalación git

```
jhoan.pea01@soy.sena.edu.co@ddne:~$ sudo apt-get install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.34.1-1ubuntu1.11).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Nota: Imagen generada de manera local (Equipo DDNE – SENA ADSO 2617551)

7. NodeJS - Backend

Ejecutar en la consola `git clone https://github.com/Esteban1543/ddne-react`, y así clonar el repositorio que contiene la API.

Ilustración 6 - clonar repositorio nodejs

Comando de clonar repositorio Backend

```
jhoan.pea01@soy.sena.edu.co@ddne:~$ git clone https://github.com/Esteban1543/ddne-nodejs
Cloning into 'ddne-nodejs'...
Username for 'https://github.com': Esteban1543
Password for 'https://Esteban1543@github.com':
remote: Enumerating objects: 6512, done.
remote: Counting objects: 100% (841/841), done.
remote: Compressing objects: 100% (277/277), done.
remote: Total 6512 (delta 522), reused 839 (delta 522), pack-reused 5671
Receiving objects: 100% (6512/6512), 14.58 MiB | 21.83 MiB/s, done.
Resolving deltas: 100% (2205/2205), done.
```

Nota: Imagen generada de manera local (Equipo DDNE – SENA ADSO 2617551)

8.React - Frontend

Ejecutar en la consola **git clone** <https://github.com/Esteban1543/ddne-nodejs>, para clonar el repositorio del frontend.

Ilustración 7 - clonar repositorio react

Comando de clonar repositorio react

```
jhoan.pea01@soy.sena.edu.co@ddne:~$ git clone https://github.com/Esteban1543/ddne-react
Cloning into 'ddne-react'...
Username for 'https://github.com': Esteban1543
Password for 'https://Esteban1543@github.com':
remote: Enumerating objects: 3893, done.
remote: Counting objects: 100% (168/168), done.
remote: Compressing objects: 100% (83/83), done.
remote: Total 3893 (delta 106), reused 114 (delta 84), pack-reused 3725
Receiving objects: 100% (3893/3893), 4.41 MiB | 10.78 MiB/s, done.
Resolving deltas: 100% (1896/1896), done.
```

9.NodeJS Dependencias

Ilustración 8 - instalar dependencias nodejs

Comando de instalación dependencias nodejs

En la carpeta donde se clono el repositorio de NodeJS – Backend ejecutar el comando **npm install** para instalar las dependencias.

```
jhoan.pea01@soy.sena.edu.co@ddne:~$ cd ddne-nodejs/
jhoan.pea01@soy.sena.edu.co@ddne:~/ddne-nodejs$ npm i

added 296 packages, and audited 297 packages in 5s

53 packages are looking for funding
  run `npm fund` for details

5 vulnerabilities (3 moderate, 1 high, 1 critical)
To address issues that do not require attention, run:
```


10. ReactJS Dependencias

*En la carpeta donde se clono el repositorio de React – Backend ejecutar el comando **npm install** para instalar las dependencias.*

Ilustración 9 - instalar dependencias react

Comando de instalación react

```
jhoan.pea01@soy.sena.edu.co@ddne:~/ddne-nodejs$ cd ..
jhoan.pea01@soy.sena.edu.co@ddne:~$ cd ddne-react/
jhoan.pea01@soy.sena.edu.co@ddne:~/ddne-react$ npm i
npm warn deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm warn deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.
npm warn deprecated stable@0.1.8: Modern JS already guarantees Array#sort() is a stable sort, so this library is deprecated. See the compatibility table on
  ists/Array/sort#browser_compatibility
npm warn deprecated source-map-url@0.4.1: See https://github.com/lydell/source-map-url#deprecated
npm warn deprecated source-map-resolve@0.5.3: See https://github.com/lydell/source-map-resolve#deprecated
npm warn deprecated sourcemap-codec@1.4.8: Please use @jridgewell/sourcemap-codec instead
npm warn deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use @rollup/plugin-terser
npm warn deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
```



DDNE
Inventory