- **vagrant file** configuration

```ruby
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure("2") do |config|

  # Web server configuration
  config.vm.define "web_server" do |web|
    web.vm.box = "bento/debian-11"
    web.vm.hostname = "web-server"

    web.vm.provider "vmware_desktop" do |v|
      v.memory = 2048
      v.cpus = 2
      v.gui= true
    end

    # Forwarded port for web server
    web.vm.network "forwarded_port", guest: 80, host: 8080

    # Private network for web server
    web.vm.network "private_network", ip: "192.168.33.10"

    # Synced folder for web server
    web.vm.synced_folder ".", "/vagrant"

    # Provision web server (install Nginx)
    web.vm.provision "shell", inline: <<-SHELL
      apt-get update
      apt-get install -y nginx
    SHELL
  end

  # Database server configuration
  config.vm.define "db_server" do |db|
    db.vm.box = "bento/debian-11"
    db.vm.hostname = "db-server"

    db.vm.provider "vmware_desktop" do |v|
      v.memory = 1024
      v.cpus = 2

    end

    # Private network for database server
    db.vm.network "private_network", ip: "192.168.33.11"

    # Forwarded port for web server
    db.vm.network "forwarded_port", guest: 3306, host: 3306

    # Provision database server (install MySQL)
    db.vm.provision "shell", inline: <<-SHELL
      apt-get update
      apt-get install -y mariadb-server
      sudo systemctl enable mariadb
      sudo systemctl start mariadb
    SHELL
  end

end
```

- Use **vagrant up** to start both machines

```
PS D:\vagrant_project\Servers> vagrant up
Bringing machine 'web_server' up with 'vmware_desktop' provider...
Bringing machine 'db_server' up with 'vmware_desktop' provider...
```

- **vagrant status** to check the status

```
PS D:\vagrant_project\Servers> vagrant status
Current machine states:

web_server                running (vmware_desktop)
db_server                 running (vmware_desktop)
```

- **ssh** both machine

```
PS D:\vagrant_project\Servers> vagrant ssh db_server
Linux db-server 5.10.0-31-amd64 #1 SMP Debian 5.10.221-1 (2024-07-14) x86_64

This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento

Use of this system is acceptance of the OS vendor EULA and License Agreements.

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
vagrant@db-server:~$
```

```
PS D:\vagrant_project\Servers> vagrant ssh web_server
Linux web-server 5.10.0-31-amd64 #1 SMP Debian 5.10.221-1 (2024-07-14) x86_64

This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento

Use of this system is acceptance of the OS vendor EULA and License Agreements.

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
vagrant@web-server:~$
```

- Accessing the running application on the guest machine via the host machine

localhost:8080

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

- **Index.html** file created on the host machine is automatically shared with the VM machine.

```
vagrant@web-server:~$
vagrant@web-server:~$ cd /vagrant
vagrant@web-server:/vagrant$ ls
Vagrantfile  web
vagrant@web-server:/vagrant$ cd web/
vagrant@web-server:/vagrant/web$ ls
index.html
vagrant@web-server:/vagrant/web$
```