**Cách deploy ứng dụng Nodejs lên EC2**

1. **sudo yum update –y**
2. **sudo yum install git**
3. **sudo su –**
4. **curl -sL https://rpm.nodesource.com/setup\_11.x | bash -**
5. **sudo su – ec2-user**
6. **sudo yum install -y nodejs**
7. **node --version**
8. **npm –version**
9. **mkdir thumuc**
10. **git clone** [**https://github.com/NguyenVanHoang19/OLAZ-Module1.git**](https://github.com/NguyenVanHoang19/OLAZ-Module1.git)
11. **ls**
12. **cd thumuc**
13. **npm install**
14. **-- chinh sua cacs file**
15. **sudo nano DangKy.js**
16. **Ctrl O -> Enter -> Ctrl X**
17. **Di chuyển đến thư mục project**
18. **Sudo node index.js**

* **Sử dụng pm2 để quản lý các tiến trình nodejs**

1. **npm install pm2 -g**
2. **pm2 start index.js**

**- xoa 1 tien trinh**

**pm2 delete tên\_process**

**pm2 stop app**

**pm2 start app**

**pm2 restart app**

* **Xoa thu muc trong linux**

**rm -rf dbstore/**

**cd ..**

**Cách upload file lên github**

* **git init**
* **git add .**
* **git commit –m“upload”**
* **git remote add origin https://github.com/NguyenVanHoang19/DeGK\_T5.git**
* **git push –f origin master**

**Cách tạo dự án Nodejs**

**npm init**

**npm install express ejs aws-sdk body-parser**

**Thao tác với DynamoDB**

1. Service
2. Tables
3. Create table
4. Create Table Name.
5. Create Primary key (Chọn kiểu dữ liệu cho khóa).
6. Tick use default setting.
7. Click create.

**Triển khai database lên AWS**

1. Cấu hình AWS phía server Nodejs.
2. Cài đặt thư viện aws-sdk ( npm install aws-sdk )
3. Cấu hình DynamoDB Client

const dynamoDB = {

region: region, // vùng tạo dynamo của bạn

accessKeyId : accessKeyId, // accessKey user IAM

secretAccessKey : secretAccessKey //secretAccessKey user IAM

}

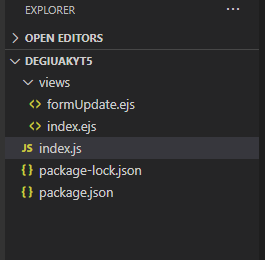
**Sử dụng nodejs/express để lập trình ứng dụng ( mô tả cách làm )**

1. Cài đặt các thư viện liên quan.
   1. Npm install express aws-sdk ejs body-parser
   2. Config AWS DynamoDB.
   3. Viết các api CRUD xử lý các chức năng.
   4. Sử dụng express cho server chạy ở port.
   5. Tạo giao diện hiển thị danh sách view EJS.

**Deploy lên AWS**

1. Tạo EC2 Linux
2. Connect EC2
3. Update máy ảo ( Sudo yum update –y )
4. Cài đặt các thư viện ( git và nodejs )
5. Tạo thư mục chứa project
6. Clone project từ Github.
7. Chỉnh sửa file server ( Thêm key AWS )
8. Chạy server.

Project CRUD



Index.js

**const express = require("express")**

**const app = express();**

**const aws = require("aws-sdk")**

**const bodyParser = require("body-parser")**

**app.use(bodyParser.urlencoded({extended:true}))**

**app.use(express.json({extended:false}));**

**app.set("view engine","ejs")**

**app.set("views","./views")**

//config

const region = "us-east-2";

const accessKeyId = "";

const secretAccessKey = "";

app.listen(5000,(err)=>{

if(err)

console.log("Loi: ",err);

else

console.log("server runing port 5000");

});

**const dynamoDB = new aws.DynamoDB.DocumentClient**({

region: region,

accessKeyId : accessKeyId,

secretAccessKey : secretAccessKey

**})**

**// get danh sach Linh Kien**

app.get("/",(req,res)=>{

const paramsDanhSachLinhKien = {

TableName : "LinhKien",

};

dynamoDB.scan(paramsDanhSachLinhKien,(error,data)=>{

if(error)

console.log(JSON.stringify(error,null,2));

else

res.render("index",{

linhKien : data.Items

});

});

});

**// add linh kien /api/addLinhKien**

app.post("/api/addLinhKien",(req,res)=>{

const {maLinhKien,ten,donViTinh,gia,thongSoKyThuat} = req.body;

console.log(maLinhKien);

const linhKien = {

maLinhKien : maLinhKien,

ten : ten,

donViTinh : donViTinh,

gia : gia,

thongSoKyThuat : thongSoKyThuat

};

const paramsAddLinhKien = {

TableName : "LinhKien",

Item: linhKien

};

dynamoDB.put(paramsAddLinhKien,(error,data)=>{

if(error){

console.log("Loi",error);

return res.json({msg:"Lỗi khi thêm"});

}

else

res.redirect("/");

// return res.json({msg:"Thêm thành công!!!!"});

});

});

**// delete linh kien**

app.post("/deleteLinhKien",(req,res)=>{

const maLinhKien = req.body.maLinhKien;

const paramsDeleteLinhKien= {

TableName : "LinhKien",

Key : {

"maLinhKien" : maLinhKien,

},

};

dynamoDB.delete(paramsDeleteLinhKien,(error,data)=>{

if(error)

console.log(error)

else

res.redirect("/");

});

});

**// render form update**

app.post("/updateForm",(req,res)=>{

const {maLinhKien,ten,donViTinh,gia,thongSoKyThuat} = req.body;

console.log(maLinhKien)

const linhKien = {

maLinhKien : maLinhKien,

ten : ten,

donViTinh : donViTinh,

gia : gia,

thongSoKyThuat : thongSoKyThuat

};

res.render("formUpdate",{

linhKien : linhKien

});

});

**// update Linh Kien**

app.post("/updateLinhKien",(req,res)=>{

const {maLinhKien,ten,donViTinh,gia,thongSoKyThuat} = req.body;

const paramsUpdateLinhKien = {

TableName : "LinhKien",

Key : {

"maLinhKien" : maLinhKien

},

UpdateExpression: "set ten = :ten , donViTinh = :dvt , gia = :gia , thongSoKyThuat = :thongSoKyThuat",

ExpressionAttributeValues:{

":ten": ten,

":dvt": donViTinh,

":gia": gia,

":thongSoKyThuat": thongSoKyThuat

},

ReturnValues: "UPDATED\_NEW",

};

dynamoDB.update(paramsUpdateLinhKien,(error,data)=>{

if(error)

console.log(error);

else

res.redirect("/");

});

});

**/// tim kiem ten linh kien theo ten**

app.post("/timKiemLinhKienTheoTen",(req,res)=>{

const ten = req.body.ten;

console.log(ten);

const paramsTimKiemTheoTen = {

TableName : "LinhKien",

FilterExpression: "contains(ten, :t)",

ExpressionAttributeValues:{

":t" : ten

},

};

dynamoDB.scan(paramsTimKiemTheoTen,(error,data)=>{

if(error)

console.log(error);

else

res.render("index",{

linhKien: data.Items

});

});

});

**// tim kiem tuyet doi**

app.post("/timKiemTuyetDoiTheoTen",(req,res)=>{

const ten = req.body.ten;

const paramsTimKiemTheoTenTuyetDoi = {

TableName : "LinhKien",

FilterExpression: "ten = :t",

ExpressionAttributeValues:{

":t":ten

},

};

dynamoDB.scan(paramsTimKiemTheoTenTuyetDoi,(error,data)=>{

if(error)

console.log(error);

else

res.render("index",{

linhKien : data.Items

});

});

})

**File index.ejs**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Linh Kiện</title>

</head>

<body>

<p>Thêm Linh Kiện</p>

<form action="/api/addLinhKien" method="POST">

<input type="text" name="maLinhKien" placeholder="Nhập mã linh kiện">

<br/><br/>

<input type="text" name="ten" placeholder="Nhập tên linh kiện">

<br/><br/>

<input type="text" name="donViTinh" placeholder="Nhập đơn vị tính">

<br/><br/>

<input type="text" name="gia" placeholder="Nhập giá">

<br/><br/>

<input type="text" name="thongSoKyThuat" placeholder="Nhập thông số kỹ thuật">

<br/><br/>

<input type="submit" value="Thêm sinh viên">

</form>

<P>Tìm kiếm linh kiện theo tên</P>

<form action="/timKiemLinhKienTheoTen" method="POST">

<input type="text" name="ten" placeholder="Nhập vào tên cần tìm kiếm">

<input type="submit" value="Tìm kiếm">

</form>

<p>Danh Sách Linh Kiện</p>

<table border="1">

<tr>

<th>Mã Linh Kiện</th>

<th>Tên linh kiện</th>

<th>Đơn vị tính</th>

<th>Giá</th>

<th>Thông số kỹ thuật</th>

<th>Cập nhật</th>

<th>Xóa</th>

</tr>

<% for(let i=0 ; i< linhKien.length; i++){%>

<tr>

<td><%= linhKien[i].maLinhKien %></td>

<td><%= linhKien[i].ten %></td>

<td><%= linhKien[i].donViTinh %></td>

<td><%= linhKien[i].gia %></td>

<td><%= linhKien[i].thongSoKyThuat %></td>

<td>

<form action="/updateForm" method="POST">

<input type="hidden" name="maLinhKien" value="<%= linhKien[i].maLinhKien %>">

<input type="hidden" name="ten" value="<%= linhKien[i].ten %>">

<input type="hidden" name="donViTinh" value="<%= linhKien[i].donViTinh %>">

<input type="hidden" name="gia" value="<%= linhKien[i].gia %>">

<input type="hidden" name="thongSoKyThuat" value="<%= linhKien[i].thongSoKyThuat %>">

<input type="submit" value="Cập nhật">

</form>

</td>

<td>

<form action="/deleteLinhKien" method="POST">

<input type="hidden" name="maLinhKien" value="<%= linhKien[i].maLinhKien %>">

<input type="submit" value="Xóa">

</form>

</td>

</tr>

<%}%>

</table>

</body>

</html>

**File form update**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Update</title>

</head>

<body>

<p>Cập nhật thông tin linh kiện có mã <%= linhKien.maLinhKien %></p>

<form action="/updateLinhKien" method="POST">

<input type="text" name="ten" value="<%= linhKien.ten%>">

<br/><br/>

<input type="text" name="donViTinh" value="<%= linhKien.donViTinh %>">

<br/><br/>

<input type="text" name="gia" value="<%= linhKien.gia %>">

<br/><br/>

<input type="text" name="thongSoKyThuat" value="<%= linhKien.thongSoKyThuat %>">

<br/><br/>

<input type="hidden" name="maLinhKien" value="<%= linhKien.maLinhKien %>">

<input type="submit" value="Cập nhật linh kiện">

</form>

</body>

</html>

**// Sử dụng S3**

**Config policy**

{

"Version": "2012-10-17",

"Statement": [

{

"Sid": "PublicRead",

"Effect": "Allow",

"Principal": "\*",

"Action": [

"s3:GetObject",

"s3:GetObjectVersion"

],

"Resource": "arn:aws:s3:::sinhvien-giuaky/\*"

}

]

}

<a href="/test/12/13">Click vao toi</a>

app.get("/test/:id/:j",(req,res)=>{

const id = req.params.id;

const j = req.params.j;

})