



# WEBAPP PROMOTING CLEAN ENERGY DEPLOYED ON AWS Elastic Kubernetes Service (EKS)

## 1. Contents

### Contents

1. Contents .....	2
2. Abstract .....	3
3. Acknowledgments .....	3
4. Introduction .....	4
a. Aims of the project .....	5
b. Scope .....	5
5. Architecture .....	6
a. Infrastructure Design Specification .....	6
b. Continuous Integration and Continuous Deployment .....	7
- Buildspec.yml .....	8
- Deployment.yaml .....	9
- Dockerfile .....	10
c. Environment information's .....	10
- AWS Elastic Kubernetes Service (EKS) .....	11
- AWS CodeCommit .....	13
- AWS Parameter Store .....	14
- AWS CodePipeline .....	14
- AWS CodeBuild .....	15
- AWS Elastic Container Repository (ECR) .....	16
- AWS CloudWatch .....	16
- AWS Identity Access Management (IAM) .....	17
- AWS RDS Database .....	19
- AWS Security Groups .....	20
d. Application Design and Specification .....	21
- Home page .....	22
- About page .....	22
- Contact page .....	23
- Calculator page .....	23
- Records .....	23
- Solar Panels .....	24
- Electric Cars .....	25
- Solar Water Panels .....	25

-	Fence turbines.....	26
-	Wind Turbines.....	26
-	River Turbines .....	27
-	_Layout.cshtml file to show CSS implementation for the website .....	31
6.	Application Development environment .....	34
7.	Infrastructure Development Environment .....	34
8.	Database Design.....	35
9.	Application Testing.....	35
a.	Performance testing .....	35
b.	Manual testing .....	37
10.	Infrastructure testing .....	39
11.	Conclusions .....	41
12.	References / Bibliography.....	41

## 2. Abstract

The main project aim was to build webapp, deploy in cloud and build continuous Integration/continuous deployment around this system. The app provides some information's about clean energy possibilities for homeowners in Ireland, links to companies which can help with it and links for grants from government if technology apply for such grant. Webapp also has solar panels calculator which provide average energy generation according to the size of panels in Ireland, all entries from calculator are stored in database (AWS RDS SQL) for future use and are visible for admins on this site. Once the application was ready, docker container image was built and stored in AWS ECR (Elastic Container Service), another step was to deploy in AWS EKS (Elastic Kubernetes Service) cluster. Last step was to build continuous Integration/continuous deployment system.

## 3. Acknowledgments

I would like to thank my teachers who gave me opportunity to work on this project. During the project I have learned a lot about all framework needed to finish project like ASP.NET core and AWS Elastic Kubernetes Service, SQL, and other useful info about clean energy technologies. Big thanks to my supervisor Mehran Rafiee who give me golden points which help me a lot in my project.

For ASP.NET core I have used these videos from YouTube channel, and tutorial site:

<https://www.youtube.com/watch?v=iKmUMgZj-cE> (Industrial and IT Automation, n.d.)

[https://www.youtube.com/watch?v=YUPg41kG\\_kw](https://www.youtube.com/watch?v=YUPg41kG_kw) (BoostMyTool, n.d.)

<https://www.w3schools.com/asp/default.asp> (W3Schools, n.d.)

For AWS EKS I have used these videos from YouTube channel, and tutorial sites:

<https://www.youtube.com/watch?v=LlisKI-gN5w> (Sandip Das, n.d.)

<https://www.youtube.com/watch?v=nEK7e0QUVio> (Sandip Das, n.d.)

<https://docs.aws.amazon.com/eks/latest/userguide/getting-started.html> (AWS, n.d.)

<https://docs.aws.amazon.com/eks/latest/userguide/add-user-role.html> (AWS, n.d.)

<https://docs.aws.amazon.com/eks/latest/userguide/metrics-server.html> (AWS, n.d.)

For Calculator formula used these websites:

<https://www.saurenergy.com/solar-energy-blog/here-is-how-you-can-calculate-the-annual-solar-energy-output-of-a-photovoltaic-system> (Saur Energy, n.d.)

<https://solargis.com/maps-and-gis-data/download/ireland> (Solargis, n.d.)

For information about AWS tools used these websites:

<https://docs.aws.amazon.com/whitepapers/latest/overview-deployment-options/amazon-elastic-kubernetes-service.html> (AWS, n.d.)

<https://aws.amazon.com/codecommit/> (AWS, n.d.)

<https://docs.aws.amazon.com/systems-manager/latest/userguide/systems-manager-parameter-store.html> (AWS, n.d.)

<https://aws.amazon.com/codepipeline/> (AWS, n.d.)

<https://aws.amazon.com/codebuild/> (AWS, n.d.)

<https://aws.amazon.com/ecr/> (AWS, n.d.)

<https://aws.amazon.com/ecr/> (AWS, n.d.)

<https://aws.amazon.com/cloudwatch/> (AWS, n.d.)

<https://aws.amazon.com/iam/> (AWS, n.d.)

## 4. Introduction

My Project is online Web Portal wrote in ASP.NET core framework, which contains information about clean energy technologies that can be used by homeowners, links for fitting companies in Ireland and links for grants if they apply for that technology also in Ireland. Is deployed on AWS cloud EKS (AWS Kubernetes) and contain CI/CD (Continuous Integration/Continuous Deployment) system which use AWS CodeCommit, CodePipeline, ParameterStore, CloudWatch, Elastic Container Repository and CodeBuild. It is Highly Available, Secure, Redundant and a Scalable system, deployed on two availability zones on North Virginia region us-east-1a and us-east-1b.

#### a. Aims of the project

1. Implement what I have already learned during the Higher Diploma
2. Learn new tools and frameworks needed by current DevOps engineers which I am planning to be in future
3. Be helpful for people looking for clean energy technologies
4. Be my portfolio for future employer

Ad. 1 During my Higher Diploma studies I have learned a lot about the AWS Cloud tools and wasn't difficult to design and implement VPC (virtual private network) for the project as we had modules which covered this technology, also creating networking, subnetting for the system was simple. ASP.NET core was also in one of modules and material help me a lot in this project. On one of assignment, I did AWS ECS (Elastic Container Service) deployment and had some knowledge about Docker and containerization on AWS.

Ad.2. One of main tools I had to learn was ASP.NET layout of the page and how CSS is implemented in such framework, how you use Razor pages and the model cooperating with view of the page. Also seeing how to save, retrieve data from SQL database.

AWS EKS how to deploy and maintain cluster where most of work is done through AWS CLI with kubectl plugin commands.

Ad.3. Hope this web application will be useful and helpful for users, having information of clean energy technologies explained, have links to fitting companies around Ireland and links for possible grants from Government institutions like Sustainable Energy Authority of Ireland.

#### b. Scope

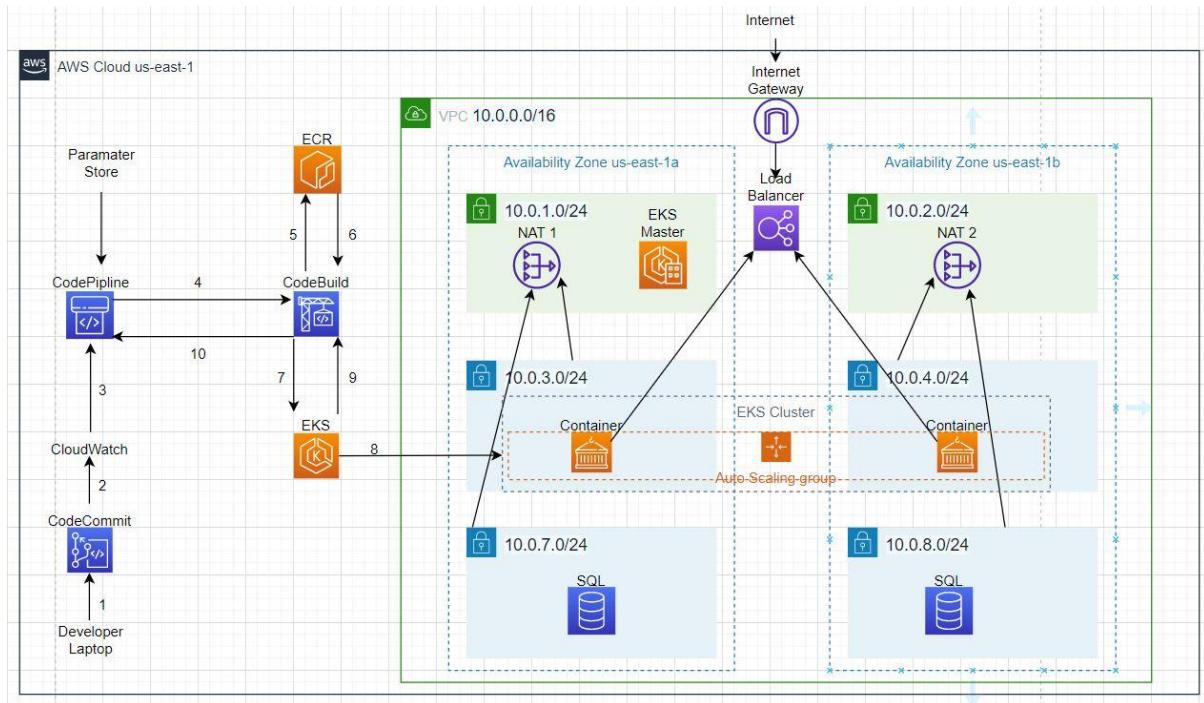
Scope of this project follow:

- Webapp in ASP.NET core
  - a. Website form and calculations
  - b. Database connection for storing results
  - c. Display all records after logging in
- Continuous Integration and Continuous deployment system on AWS by using CodePipeline, CodeBuild, CodeCommit, Parameter store from System Manager, CloudWatch events
- AWS CodeCommit setup, where we store our webapp code

- AWS ECR setup, where we store our webapp docker container
- AWS EKS service setup, orchestrator tool that our containers run on
- Acquiring all necessary clean energy details and formula to make calculations for users
- AWS VPC setup for our infrastructure
- Setup AWS Security groups and IAM roles to achieve highly secure system
- AWS VPC Subnetting to separate frontend and backend services
- AWS Networking to make sure packet traffic flow desired way
- AWS Autoscaling group setup so our webapp scale when needed

## 5. Architecture

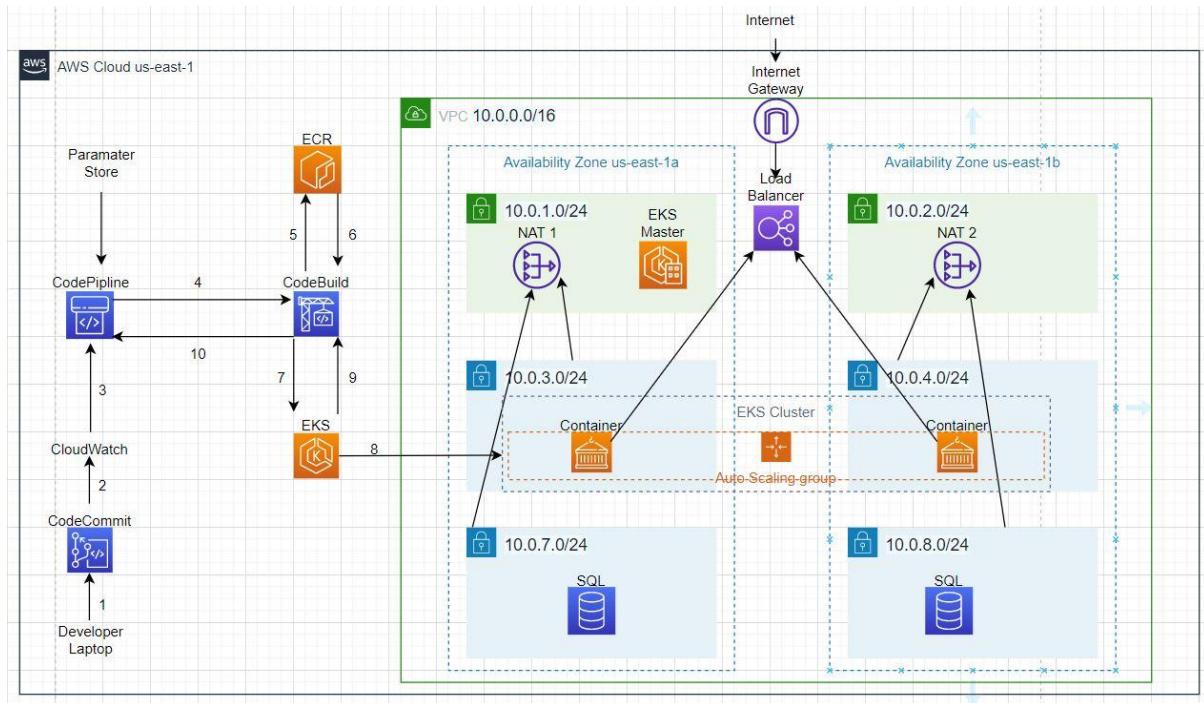
### a. Infrastructure Design Specification



EKS cluster is deployed on Private Subnet 1 and 2, backend database in Private Subnet 3 and 4 on EC2 nodes and EKS master is deployed across all subnets. There are 2 NAT gateways one on each availability zone. Containers accept traffic just from the Database security group on port 1433, load balancer security group all traffic and EKS API master security group all traffic. All outbound traffic from containers, nodes and database are going through NAT gateways. There are 2 EC2 nodes deployed across the 2 availability zones where containers run. Autoscaling group is set to 1 minimum node, desired 2 nodes and if spot needed up to 3 nodes can be deployed. EKS cluster pods autoscaling group is called Horizontal Pod Autoscaler and is set to minimum 2 pods and maximum 3 pods, is deployed by watching metrics of CPU if CPU usage is more than 70% than another pod is deployed. For this to work metrics-server must be deployed on EKS cluster which is not by default. SQL Server database on AWS RDS is used with multizone deployment. System is designed with 6 pillars of Well-Architected framework for operational excellence, security, reliability, performance efficiency, cost optimization, and sustainability also system is Highly Available by spanning through 2 availability zones. It scales on demand.

## b. Continuous Integration and Continuous Deployment

Continuous Integration (CI) is a DevOps practice where developers regularly merge their code changes into a central repository after which automated builds and tests are run. The key goal of continuous integration is to reduce the time of software updates. And Continuous deployment is automating deployment, so production happens automatically. For this project to achieve this I have used tools from AWS CodeCommit, CodePipeline, CodeBuild, Elastic Container Repository(ECR), Parameter store from System Manager and CloudWatch events.



When a developer updates the code to central repository(CodeCommit), CloudWatch event is triggered and send information to CodePipeline which starts a CI/CD pipeline, it is verifying changes first on CodeCommit, getting variables from CodePipeline project and from parameter store if that pass instruction are send to CodeBuild to start test and build docker container which is send to ECR(Elastic Container Repository), once this stage is done, CodeBuild sends deployment instruction to EKS(Elastic Kubernetes Service), and if all stages pass on CodeBuild, than CodeBuild sends information of finished process to CodePipeline and process is done. Changes done by developer are applied to production in 2-3 min.

CodeBuild require 3 files to run buildspec.yml, deployment.yaml that run with kubectl(Kubernetes CLI) and dockerfile for building docker container:

- Buildspec.yml

```

version: 0.2

phases:
  install:
    commands:
      - echo Installing app dependencies...
      - curl -o kubectl https://s3.us-west-2.amazonaws.com/amazon-
eks/1.22.6/2022-03-09/bin/linux/amd64/kubectl
      - chmod +x ./kubectl
      - mkdir -p $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export
PATH=$PATH:$HOME/bin
      - echo 'export PATH=$PATH:$HOME/bin' >> ~/.bashrc
  
```

```

- source ~/.bashrc
- mkdir -p $HOME/.kube
- aws eks --region us-east-1 update-kubeconfig --name projectEKS
- echo 'Check kubectl version'
- kubectl version --short --client
- echo check connection to claster
- kubectl get svc
pre_build:
  commands:
    - echo Logging in to docker...
    - docker login -u $docker_username -p $docker_password
build:
  commands:
    - echo Build started on `date`
    - echo Building the Docker image...
    - docker build -t $IMAGE_REPO_NAME .
    - docker tag $IMAGE_REPO_NAME:$IMAGE_TAG
$AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$IMAGE_TAG
post_build:
  commands:
    - echo Build completed on `date`
    - echo Pushing the Docker image to ECR...
    - aws ecr get-login-password --region $AWS_DEFAULT_REGION | docker login --username AWS --password-stdin
$AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com
    - docker push
$AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$IMAGE_TAG
    - kubectl apply -f eks/deployment.yaml
    - kubectl rollout restart -f eks/deployment.yaml
  "
    - Deployment.yaml
  "

```

```

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app.kubernetes.io/name: cleanenergy
    app.kubernetes.io/instance: cleanenergy-instance
    app.kubernetes.io/version: "1.0.0"
    app.kubernetes.io/manage-by: kubectl
  name: cleanenergy
spec:
  replicas: 2
  selector:
    matchLabels:

```

```

    app: cleanenergy
template:
  metadata:
    labels:
      app: cleanenergy
spec:
  containers:
    - image: 953941695125.dkr.ecr.us-east-
1.amazonaws.com/cleanenergycontainer:latest
      imagePullPolicy: Always
      name: cleanenergy
      resources:
        requests:
          cpu: "250m"
        limits:
          cpu: "500"

      ports:
        - containerPort: 80
    "
  "

```

- Dockerfile

```

FROM mcr.microsoft.com/dotnet/sdk:6.0 AS build-env
WORKDIR /app

# Copy everything
COPY . .
# Restore as distinct layers
RUN dotnet restore
# Build and publish a release
RUN dotnet publish -c Release -o out

EXPOSE 80

# Build runtime image
FROM mcr.microsoft.com/dotnet/aspnet:6.0
WORKDIR /app
COPY --from=build-env /app/out .
ENTRYPOINT [ "dotnet", "CleanEnergy.dll" ]
"

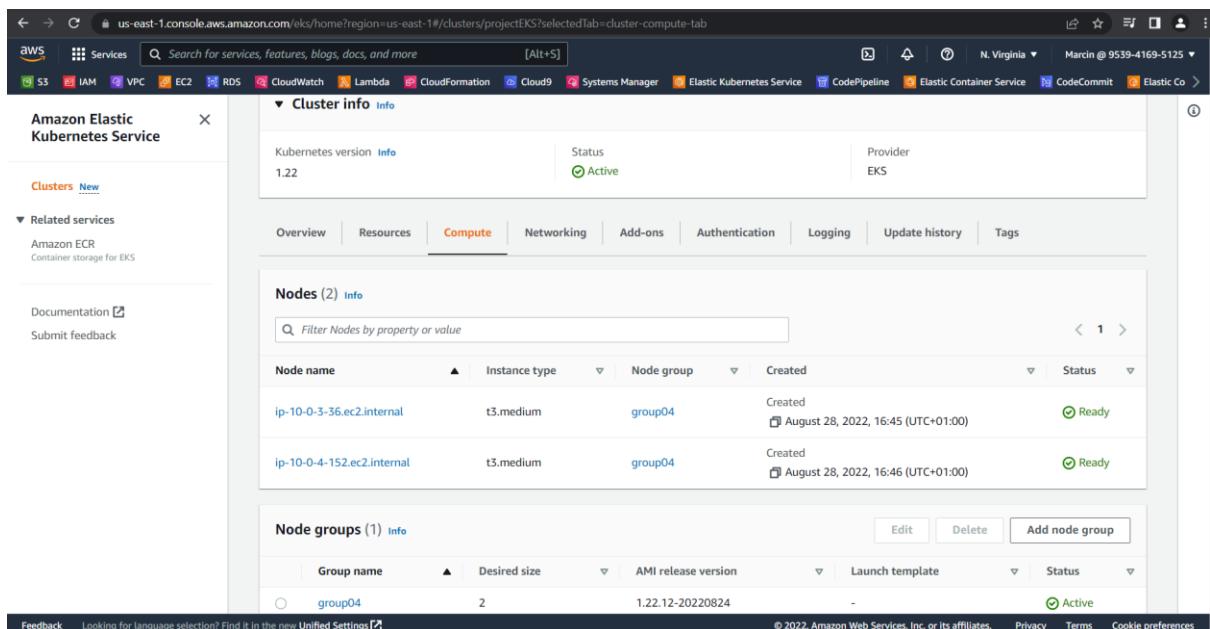
```

c. Environment information's

- AWS Elastic Kubernetes Service (EKS)

Amazon Elastic Kubernetes Service is fully managed service where you can run Kubernetes on AWS without needing to install, operate and maintain Kubernetes control plane or nodes. It is integrated with the core AWS services like Auto Scaling Groups, VPC (Virtual Private Network), ECR (Elastic Container Repository), CloudWatch, IAM, load balancer for your containerized applications. As the orchestrator tool is very popular in modern businesses that can automate configuration, management, and coordinate of computer systems, applications, and services. When running applications on EKS you can choose underlying computer power for containers from EC2 or Fargate.

For this project EC2 was chosen as underling computer power for containers:



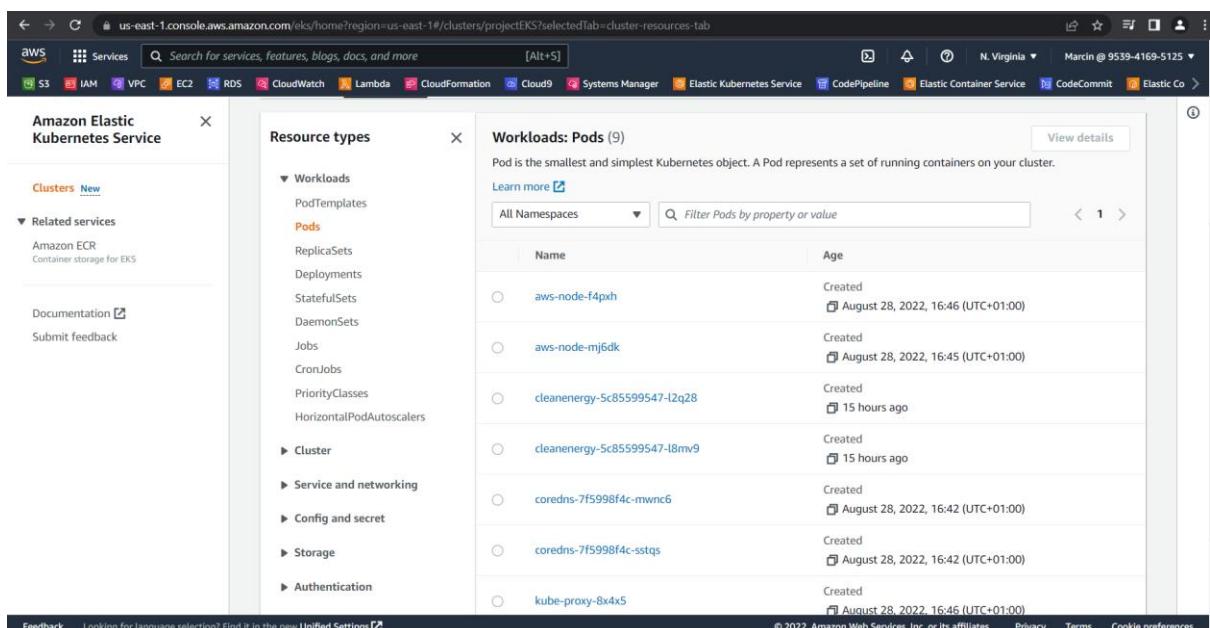
The screenshot shows the AWS EKS Cluster Compute tab. It displays the following information:

- Cluster info:** Kubernetes version 1.22, Status Active, Provider EKS.
- Nodes (2) info:** Two EC2 nodes are listed:
 

Node name	Instance type	Node group	Created	Status
ip-10-0-3-36.ec2.internal	t3.medium	group04	Created August 28, 2022, 16:45 (UTC+01:00)	Ready
ip-10-0-4-152.ec2.internal	t3.medium	group04	Created August 28, 2022, 16:46 (UTC+01:00)	Ready
- Node groups (1) info:** One node group is listed:
 

Group name	Desired size	AMI release version	Launch template	Status
group04	2	1.22.12-20220824	-	Active

Have 2 pods deployed for clean energy application:



The screenshot shows the AWS EKS Cluster Resources tab. It displays the following information:

- Resource types:** Workloads, PodTemplates, Pods, Deployments, StatefulSets, DaemonSets, Jobs, CronJobs, PriorityClasses, HorizontalPodAutoscalers.
- Workloads: Pods (9):** A list of pods:
 

Name	Age
aws-node-f4pxh	Created August 28, 2022, 16:46 (UTC+01:00)
aws-node-mj6dk	Created August 28, 2022, 16:45 (UTC+01:00)
cleanenergy-5c85599547-l2q28	Created 15 hours ago
cleanenergy-5c85599547-l8mv9	Created 15 hours ago
coredns-7f5998f4c-mwnic6	Created August 28, 2022, 16:42 (UTC+01:00)
coredns-7f5998f4c-sstq5	Created August 28, 2022, 16:42 (UTC+01:00)
kube-proxy-8x4x5	Created August 28, 2022, 16:46 (UTC+01:00)

Have 1 pod deployed for metric server that helps manage Horizontal Auto Scaling:

Workloads: Pods (9)

Name	Type	Age
metrics-server-64cf6869bd-8st8l		Created 15 hours ago

Have 3 deployments running clean energy application, metric-server and dns:

Workloads: Deployments (3)

Name	Namespace	Type	Age	Pod count	Status
cleanenergy	default	deployments	Created August 21, 2022, 16:39 (UTC+01:00)	2	2 Ready   0 Failed   2 Desired
coredns	kube-system	deployments	Created August 21, 2022, 13:52 (UTC+01:00)	2	2 Ready   0 Failed   2 Desired
metrics-server	kube-system	deployments	Created 15 hours ago	1	1 Ready   0 Failed   1 Desired

Have 1 HorizontalPodAutoscaler:

The screenshot shows the AWS Elastic Kubernetes Service (EKS) console. The left sidebar has a 'Clusters' section with a 'New' button. Below it are 'Related services' (Amazon ECR) and 'Documentation' with a 'Submit feedback' link. The main content area is titled 'Workloads: HorizontalPodAutoscalers (1)'. It shows a table with one entry:

Name	Age
cleanenergy	Created 16 hours ago

- AWS CodeCommit

AWS CodeCommit is managed source control service that hosts private Git repositories. It is secure, highly scalable and is easy for team to securely collaborate on code with contribution encrypted in transit. CodeCommit support standard functionality of Git and you can store anything from code to binaries.

For this project 1 repository was created:

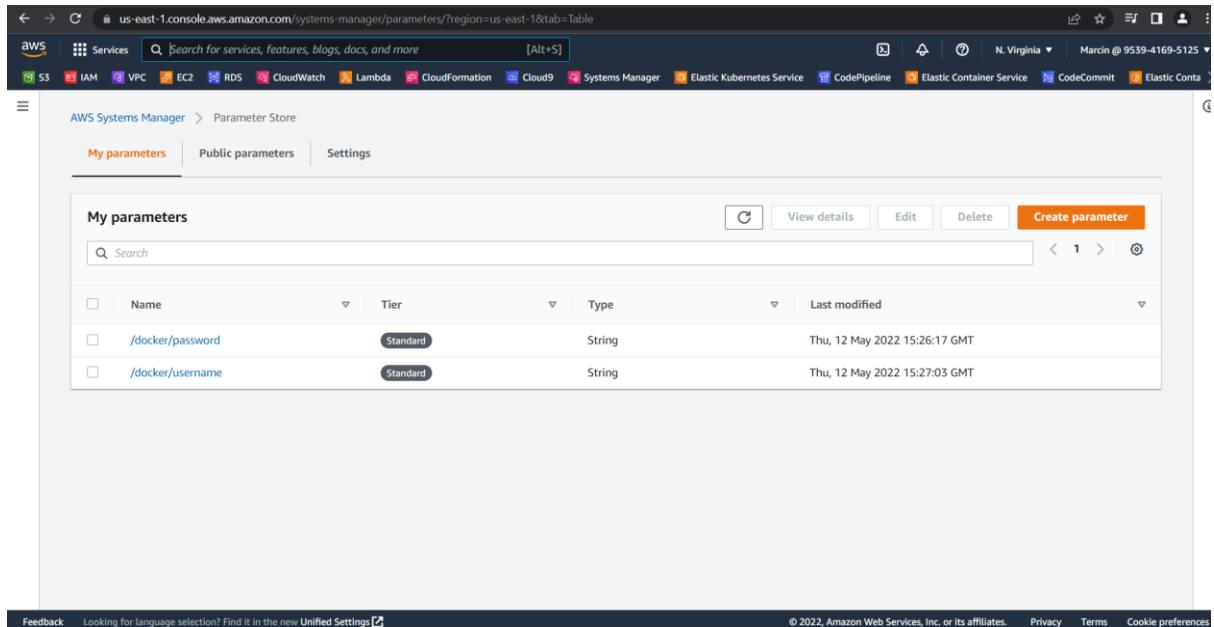
The screenshot shows the AWS CodeCommit console. The left sidebar has sections for 'Source' (CodeCommit), 'Artifacts' (CodeArtifact), 'Build' (CodeBuild), 'Deploy' (CodeDeploy), 'Pipeline' (CodePipeline), and 'Settings'. The main content area is titled 'Repositories' and shows a table with one entry:

Name	Description	Last modified	Clone URL
cleanenergyapp	-	19 hours ago	<a href="#">HTTPS</a> <a href="#">SSH</a> <a href="#">HTTPS (GRC)</a>

- AWS Parameter Store

AWS Parameter Store is part of the AWS System Manager for holding secrets, providing secure, hierarchical storage. You can store passwords, connection strings and other secrets as parameter values. All this information stored as plain text or encrypted, later can be used in your scripts, commands, documents, and configuration and automation workflows. Parameter store is integrated with other AWS services, and you can reference to Parameter store when needed.

For this project there was 2 values stored in Parameter store:



Name	Tier	Type	Last modified
/docker/password	Standard	String	Thu, 12 May 2022 15:26:17 GMT
/docker/username	Standard	String	Thu, 12 May 2022 15:27:03 GMT

- AWS CodePipeline

AWS CodePipeline is a fully managed continuous delivery service that helps to automate new applications or infrastructure updates. CodePipeline automate builds, test, and deploy phases once code is change. You can easily integrate third party services like Git, Jenkins, or other plugins. It is enabling features and updates for your application.

For this project was setup 1 CodePipeline project:

The screenshot shows the AWS CodePipeline console. The left sidebar is titled 'CodePipeline' and includes sections for Source, Artifacts, Build, Deploy, Pipeline, Getting started, Pipelines, and Settings. The main content area is titled 'Pipelines' and shows a table with one row. The table columns are Name, Most recent execution, Latest source revisions, and Last executed. The single row shows 'projectEKSpipeline' with a green 'Succeeded' status, 'Source - 531b5559: links' as the latest source revision, and '22 hours ago' as the last execution time. At the top of the main area, there are buttons for Create pipeline, Notify, View history, Release change, and Delete pipeline.

## - AWS CodeBuild

AWS CodeBuild is fully managed continuous integration service that allow to compile source code, run tests, and produce software packages that are ready to deploy. CodeBuild automatically patch and build servers, can build docker images and other software builds. It supports few programming languages and frameworks like Java, Bash, Go, Ruby, Python. It can concurrently run multiple builds; you pay for time spent to process tasks.

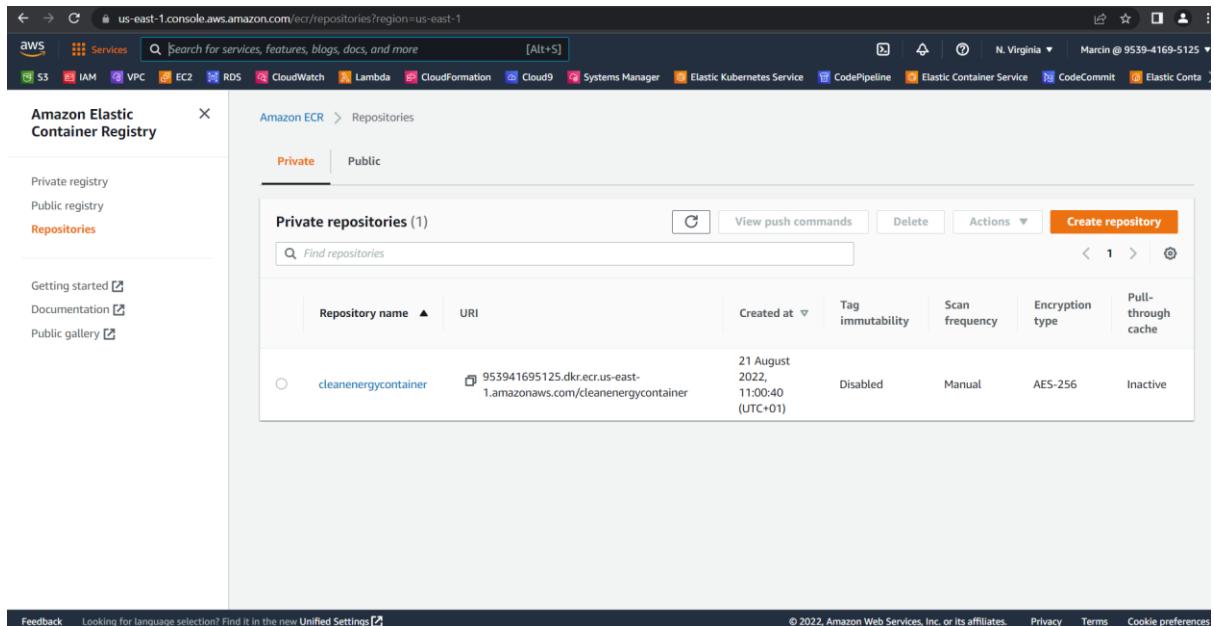
For this project 1 CodeBuild project was setup:

The screenshot shows the AWS CodeBuild console. The left sidebar is titled 'CodeBuild' and includes sections for Source, Artifacts, Build, Deploy, Pipeline, Getting started, Build projects, Build history, Report groups, Report history, Account metrics, and Settings. The main content area is titled 'Build projects' and shows a table with one row. The table columns are Name, Source provider, Repository, Latest build status, Description, and Last Modified. The single row shows 'projectEKSBuild' with 'AWS CodePipeline' as the source provider, '-' as the repository, a green 'Succeeded' status, '-' as the description, and '8 days ago' as the last modified time. At the top of the main area, there are buttons for Create build project, Notify, Start build, View details, Edit, Delete build project, and a search bar.

- AWS Elastic Container Repository (ECR)

Amazon Elastic Container Repository is managed container image registry service that is secure, scalable, and reliable. ECR is integrated with AWS Identity Access Management (IAM) for accessing images.

For this project 1 repository was created and 24 hours retention policy was added to reduce cost:



The screenshot shows the AWS ECR console with the URL [us-east-1.console.aws.amazon.com/ecr/repositories?region=us-east-1](https://us-east-1.console.aws.amazon.com/ecr/repositories?region=us-east-1). The page displays a single private repository named 'cleanenergycontainer'. The repository details are as follows:

Repository name	URI	Created at	Tag immutability	Scan frequency	Encryption type	Pull-through cache
cleanenergycontainer	953941695125.dkr.ecr.us-east-1.amazonaws.com/cleanenergycontainer	21 August 2022, 11:00:40 (UTC+01)	Disabled	Manual	AES-256	Inactive

- AWS CloudWatch

AWS CloudWatch is a monitoring and observability service build for DevOps, developers, site reliability engineers and other who use AWS services. CloudWatch collects data in form of logs, events, and metrics to provide complete visibility of AWS resources, applications and services running on AWS. You can use CloudWatch set alarms, visualize metrics and logs, take automated actions, or troubleshoot issues.

For this project, CloudWatch was used for EKS cluster, CodeBuild and for CodePipeline to trigger event when code is change on CodeCommit:

- AWS Identity Access Management (IAM)

With AWS Identity Access Management, you can specify who and what have access to services and AWS resources.

For this project a few IAM roles were created:

### CodePipeline

**cwe-role-us-east-1-projectEKSpipeline**

**Summary**

Creation date: August 27, 2022, 10:29 (UTC+01:00)

Last activity: 23 hours ago

ARN: arn:aws:iam::953941695125:role/service-role/cwe-role-us-east-1-projectEKSpipeline

Maximum session duration: 1 hour

**Permissions** | Trust relationships | Tags | Access Advisor | Revoke sessions

**Permissions policies (1)**

You can attach up to 10 managed policies.

Policy name	Type	Description
start-pipeline-execution-us-east-1-projectEKSpipeline	Customer managed	Allows Amazon CloudWatch Events to automatically...

## CodeBuild

**CodeBuildforEKSPProject**

Allows CodeBuild to call AWS services on your behalf.

**Summary**

Creation date: August 27, 2022, 09:50 (UTC+01:00)

Last activity: 23 hours ago

ARN: arn:aws:iam::953941695125:role/CodeBuildforEKSPProject

Maximum session duration: 1 hour

**Permissions** | Trust relationships | Tags | Access Advisor | Revoke sessions

**Permissions policies (8)**

You can attach up to 10 managed policies.

Policy name	Type	Description
CodeBuildBasePolicy-projectEKSPBuild-us-east-1	Customer managed	Policy used in trust relationship with CodeBuild

## Elastic Kubernetes Service

Identity and Access Management (IAM)

projectEKSclusterRole

Allows access to other AWS service resources that are required to operate clusters managed by EKS.

**Summary**

Creation date: August 21, 2022, 13:19 (UTC+01:00)

Last activity: 1 hour ago

ARN: arn:aws:iam::953941695125:role/projectEKSclusterRole

Maximum session duration: 1 hour

**Permissions**

Permissions policies (1)

AmazonEKSClusterPolicy

AWS managed

This policy provides Kubernetes the permissions it needs to manage your cluster.

## Elastic Kubernetes Service Nodes

Identity and Access Management (IAM)

projectEKSworkerNodeRole

Allows EC2 instances to call AWS services on your behalf.

**Summary**

Creation date: August 21, 2022, 14:06 (UTC+01:00)

Last activity: 23 minutes ago

ARN: arn:aws:iam::953941695125:role/projectEKSworkerNodeRole

Instance profile ARN: arn:aws:iam::953941695125:instance-profile/eks-b8c1728f-dd01-98d8-385b-57dacf2db43c

Maximum session duration: 1 hour

**Permissions**

Permissions policies (3)

AmazonEKSWorkerNodePolicy

AWS managed

This policy allows Amazon EKS worker nodes to call AWS services on behalf of the role.

## - AWS RDS Database

AWS RDS service provides selection of purpose-build databases.

For this project SQL server database was setup:

- AWS Security Groups

AWS Security Groups control the network traffic that is allowed to reach and leave resources that associates with it. For security reasons EKS nodes can allow inbound traffic just from EKS control plane, load balancer and database security groups. Database allow inbound traffic just from EKS nodes security group.

For this project 3 security groups were created:

### Elastic Kubernetes Service Nodes

## Load balancer

Details

Security group name sg-058eba78325dc7240	Security group ID sg-058eba78325dc7240	Description Security group for Kubernetes ELB a4a4412e02e54449bac3be0490108050 (default/cleanenergy)	VPC ID vpc-0f47e6a0ab92b6204
Owner 953941695125	Inbound rules count 2 Permission entries	Outbound rules count 1 Permission entry	

Inbound rules (2)

Name	Security group rule...	IP version	Type	Protocol	Port range
Allow traffic	sg-00da877c5eb5fc845	IPv4	Allow	tcp	80

## RDS SQL Server database

Details

Security group name sg-00da877c5eb5fc845	Security group ID sg-00da877c5eb5fc845	Description Allow traffic	VPC ID vpc-0f47e6a0ab92b6204
Owner 953941695125	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules (1/1)

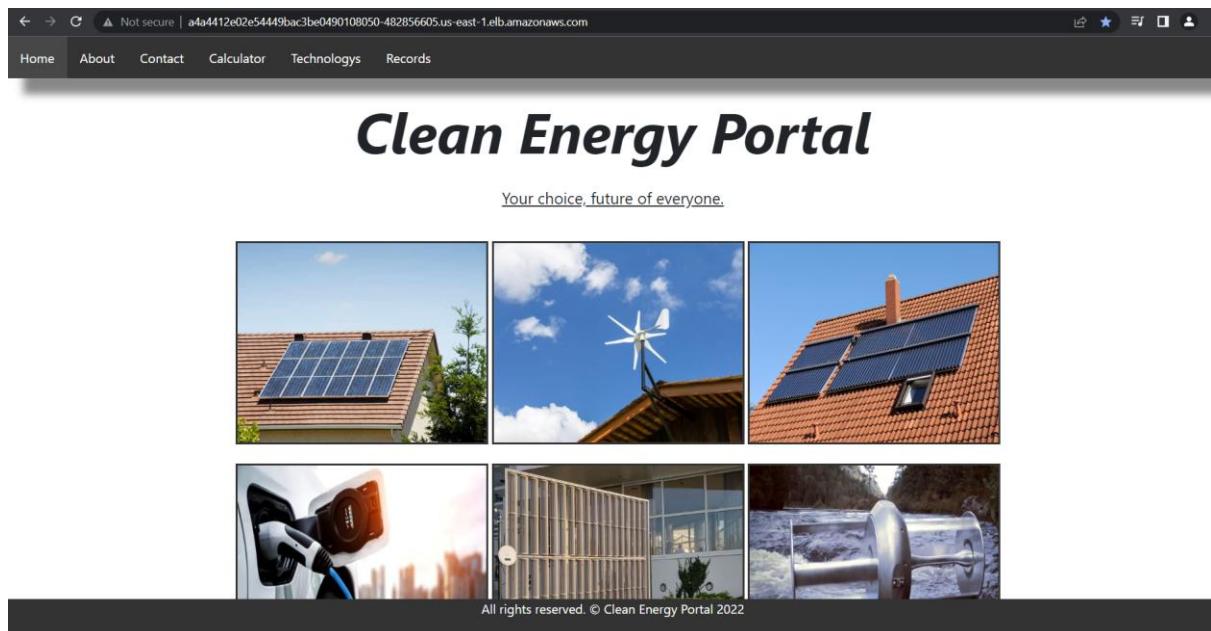
Name	Security group rule...	IP version	Type	Protocol	Port range
Allow traffic	sg-00da877c5eb5fc845	IPv4	Allow	tcp	80

## d. Application Design and Specification

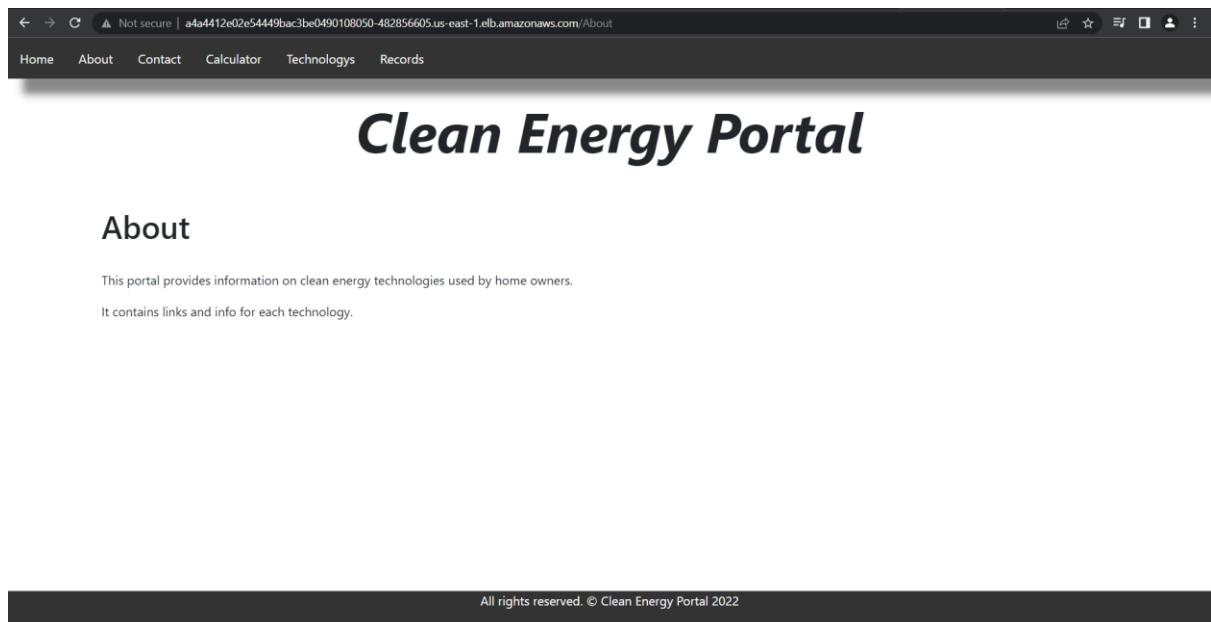
Application is written in C# ASP.NET core on Razor pages with CSS. Application contains 11 pages: Home, About, Contact, Records, Calculator, Solar Panels, Wind Turbines, Water Turbines, Solar Water Panels, Electric Cars, Fence Turbines. All pages have a navigation bar that allow to change pages and Home page has pictures that works as links to other pages. Each picture expands if hovering over it and displaying information about link, additional links on pages highlight.

Application is connected to AWS RDS database and stores information used by Calculator and results of calculation for future use. Records page display stored values in database but just after successful login on this page and is not visible for everyone that visit the site. Site is cross-web browser is working fine on Chrome, Internet Explorer, Firefox, Opera, Edge and probably others. It also works on any mobile devices.

- Home page



- About page



- Contact page

- Calculator page

- Records

← → ⌂ Not secure | a4a4412e02e54449bac3be0490108050-482856605.us-east-1.elb.amazonaws.com/Records

Home About Contact Calculator Technologies Records

## Records just for admin view.

### Records

Person ID	First Name	Last Name	Energy in KWh/year	Size of Solar Panels in m2
1	stave	hyes	2498	20
2	Madzia	Slonce	2622	21
3	Mike	Bis	3746	30
4	Daniel	Playo	3871	31
5	Nadine	Gfd	2498	20

Username

Password

Submit

All rights reserved. © Clean Energy Portal 2022

← → ⌂ Not secure | a4a4412e02e54449bac3be0490108050-482856605.us-east-1.elb.amazonaws.com/Records

Home About Contact Calculator Technologies Records

## Records just for admin view.

Username

Password

Submit

All rights reserved. © Clean Energy Portal 2022

- Solar Panels

# Clean Energy Portal

## Solar panels modules

Power your home with solar panels energy, solar panels use sunlight energy to generate direct current electricity through the photovoltaic effect.

A single solar module can produce only a limited amount of power that's why most of installations contain multiple modules to increase output of voltage.

A photovoltaic system typically includes an array of photovoltaic modules, an inverter, a battery pack for energy storage, charge controller, interconnection wiring, circuit breakers, fuses, disconnect switches, voltage meters, and optionally a solar tracking mechanism.

Power typically ranges from 100 to 365 Watts (W). The efficiency of a module determines the area of a module given the same rated output - an 8% efficient 230 W module will have twice the area of a 16% efficient 230 W module. Some commercially available solar modules exceed 24% efficiency.

Currently, the best achieved sunlight conversion rate is around 21.5% in new commercial products typically lower than the efficiencies of their cells in isolation. The most efficient mass-produced solar modules have power density values of up to 175 W/m<sup>2</sup> at the moment but may increase in future.

As new regulations when mounting solar panels power socket for electric cars charge will be mounted also.

Links to fitting companies in Ireland.

All rights reserved. © Clean Energy Portal 2022

- Electric Cars

# Clean Energy Portal

## Electric cars

An electric car, battery electric car, or all-electric car, is an automobile that is propelled by one or more electric motors, using only energy stored in batteries.

Compared to internal combustion engine vehicles, electric cars are quieter, have no exhaust emissions, and lower emissions overall.

Charging an electric car can be done at a variety of charging stations; these charging stations can be installed in both houses and public areas.

The most expensive part of an electric car is its battery. The price decreased from €605 per kWh in 2010, to €170 in 2017, to €100 in 2019.

Electricity almost always costs less than gasoline per kilometer travelled, but the price of electricity often varies depending on where and what time of day the car is charged. Cost savings are also affected by the price of gasoline which can vary by location.

Links to companies that offer electric cars in Ireland.

[Carzone](#)

[Hyundai](#)

All rights reserved. © Clean Energy Portal 2022

- Solar Water Panels

Not secure | a4a4412e02e54449bac3be0490108050-482856605.us-east-1.elb.amazonaws.com/Waterpanels

Home About Contact Calculator Technologies Records

# Clean Energy Portal

## Solar water panels

Solar water heating is heating water by sunlight using solar panels collector.

A sun-facing collector heats a working fluid that passes into a storage system for later use.

They operate independently or as hybrids with electric or gas heaters. In large-scale installations, mirrors may concentrate sunlight into a smaller collector.

Residential solar thermal installations fall into two groups: passive (sometimes called "compact") and active (sometimes called "pumped") systems.

Both typically include an auxiliary energy source (electric heating element or connection to a gas or fuel oil central heating system) that is activated when the water in the tank falls below a minimum temperature setting, ensuring that hot water is always available. The combination of solar water heating and back-up heat from a wood stove chimney can enable a hot water system to work all year round in cooler climates, without the supplemental heat requirement of a solar water heating system being met with fossil fuels or electricity.

Links to fitting companies in Ireland.

All rights reserved. © Clean Energy Portal 2022

- Fence turbines

Not secure | a4a4412e02e54449bac3be0490108050-482856605.us-east-1.elb.amazonaws.com/Fenceturbines

Home About Contact Calculator Technologies Records

# Clean Energy Portal

## Fence turbines

This technology is similar to wind turbines but fit to fences which generate electric power.

By author their can generate similar amount of energy as solar panels.

Panels are not yet in production but we are waiting for first products, at the moment engineers have working prototype and probably soon we can expect to show on market. Fences can produce energy with even small wind which is great for home owners.

Planning sales were on second part of 2021 but looks like is extended as engineers still testing the product.

Links for more inforamtion about product.

[Spider web info](#)

[Blog of Centrum nauki](#)

All rights reserved. © Clean Energy Portal 2022

- Wind Turbines

The screenshot shows a web browser window with the following details:

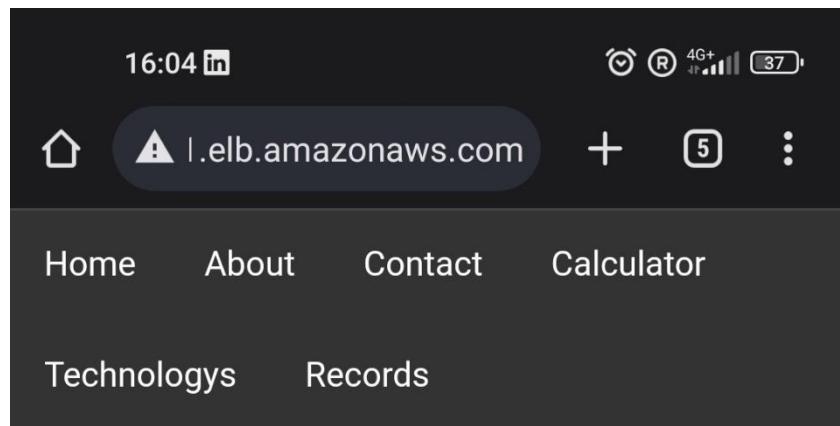
- Address Bar:** Not secure | a4a4412e02e54449bac3be0490108050-482856605.us-east-1.elb.amazonaws.com/Windturbines
- Page Title:** Clean Energy Portal
- Page Content:**
  - Section Header:** Wind Turbines
  - Text:** A wind turbine is a device that converts the kinetic energy of wind into electrical energy. Wind turbines can rotate about either a horizontal or a vertical axis, the former being both older and more common. They can also include blades or be bladeless. Household-size vertical designs produce less power and are less common. Small wind turbines may be used for a variety of applications including on- or off-grid residences, telecom towers, offshore platforms, rural schools and clinics, remote monitoring and other purposes that require energy where there is no electric grid, or where the grid is unstable. Wind turbines produce electricity at between two and six cents per kilowatt hour, which is one of the lowest-priced renewable energy sources. The energy harvested from the turbine will offset the installation cost, as well as provide virtually free energy for years.
- Footer:** Carbon free heat | All rights reserved. © Clean Energy Portal 2022

- River Turbines

The screenshot shows a web browser window with the following details:

- Address Bar:** Not secure | a4a4412e02e54449bac3be0490108050-482856605.us-east-1.elb.amazonaws.com/Riverturbines
- Page Title:** Clean Energy Portal
- Page Content:**
  - Section Header:** River Turbines
  - Text:** A water turbine is a rotary machine that converts kinetic energy and potential energy of water into mechanical work that is converted to electric power. Flowing water is directed on to the blades of a turbine runner, creating a force on the blades. Since the runner is spinning, the force acts through a distance (force acting through a distance is the definition of work). In this way, energy is transferred from the water flow to the turbine. Water turbines are divided into two groups: reaction turbines and impulse turbines. Reaction turbines are acted on by water, which changes pressure as it moves through the turbine and gives up its energy. They must be encased to contain the water pressure (or suction), or they must be fully submerged in the water flow. Impulse turbines change the velocity of a water jet. The jet pushes on the turbine's curved blades which changes the direction of the flow. The resulting change in momentum (impulse) causes a force on the turbine blades. Since the turbine is spinning, the force acts through a distance (work) and the diverted water flow is left with diminished energy. An impulse turbine is one in which the pressure of the fluid flowing over the rotor blades is constant and all the work output is due to the change in kinetic energy of the fluid.
- Footer:** Links to fitting companies in Ireland | All rights reserved. © Clean Energy Portal 2022

- Mobile page's view



Person ID	First Name	Last Name	Energy in KWh/year	Size of Solar Panels in m2
1	stave	hyes	2498	20
2	Madzia	Slonce	2622	21
3	Mike	Bis	3746	30
4	Daniel	Playo	3871	31
5	Nadine	Gfd	2498	20

Username

Password

Submit

All rights reserved. © Clean Energy Portal 2022

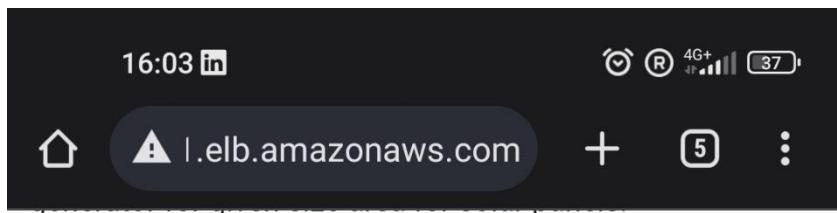
A screenshot of a smartphone displaying a mobile website. The top status bar shows the time as 16:03, signal strength, and battery level at 37%. The browser header bar includes a home icon, a search/address bar with the URL 'l.elb.amazonaws.com', a plus sign for new tabs, a tab indicator showing '5', and a more options menu. Below the header is a navigation menu with links: Home, About, Contact, Calculator, Technologys, and Records.

# *Clean Energy Portal*

Your choice, future of everyone.



All rights reserved. © Clean Energy Portal 2022



It is calculating with this formula:  $E = A * r * H * PR$ ,  $E$  = Energy (kWh),  $A$  = Total solar panel Area (m<sup>2</sup>),  $r$  = solar panel

yield or efficiency(%) (we take 15.7%)(the solar panel yield of a PV module of 250 Wp with an area of 1.6 m<sup>2</sup> is 15.6%),

$H$  = Annual average solar radiation on tilted panels (shadings not included)(for Ireland average is 970KWh/m<sup>2</sup>),  
 $PR$  = Performance ratio,

coefficient for losses (range between 0.5 and 0.9, default value we take = 0.82)

Bear in mind that is average and may vary, depend on type of solar panels and their efficiency.

## Calculation form

First Name

Last Name

Size of solar panels in m<sup>2</sup>

Submit

All rights reserved. © Clean Energy Portal 2022



- \_Layout.cshtml file to show CSS implementation for the website

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>@ ViewData["Title"] - CleanEnergy</title>
  <link rel="stylesheet" href="~/lib/bootstrap/dist/css/bootstrap.min.css" />
  <link rel="stylesheet" href="~/css/site.css" asp-append-version="true" />
  <link rel="stylesheet" href="~/CleanEnergy.styles.css" asp-append-
version="true" />
<style>
.mynav_nav {
  overflow: hidden;
  background-color: #333;
  box-shadow: 20px 20px 10px #888888;
  margin: 0;
  padding: 0;
}

.mynav_nav a {
  float: left;
  font-size: 16px;
  color: white;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}

.mynav_nav a:hover, .mynav_dropdown:hover .mynav_dropbtn {
  background-color: #4d4d4d;
}

.mynav_dropdown {
  float: left;
  overflow: hidden;
}

.mynav_dropdown .mynav_dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: white;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;
}

.mynav_dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;
  min-width: 160px;
}

```

```

        box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
        z-index: 1;
    }

    .mynav_dropdown-content a {
        float: none;
        color: black;
        padding: 12px 16px;
        text-decoration: none;
        display: block;
        text-align: left;
    }

    .mynav_dropdown-content a:hover {
        background-color: #ddd;
    }

    .mynav_dropdown:hover .mynav_dropdown-content {
        display: block;
    }

    .my_footer {
        list-style-type: none;
        background-color: #333;
        overflow: hidden;
        position: fixed;
        bottom: 0;
        color: white;
        text-align: center;
        height: 50px;
        width: 100%;

    }

    #pictures {
        display: block;
        width: 90%;
        height: 100%;
        object-fit: contain;
        margin-left: auto;
        margin-right: auto;
        padding-left: 110px;
    }

    #images {
        width: 30%;

        border: solid #333;
    }

    #images:hover {
        transform: scale(1.15);
    }

    h3 {
        text-align: center;
        font-size: 30px;
    }

    #links:hover {
        background-color: #33A8FF;
    }

```

```

</style>

</head>
<body>
    <header>

        <div class="mynav_nav">

            <a class="nav-link text-white" asp-area="" asp-page="/Index">Home</a>
            <a class="nav-link text-white" asp-area="" asp-page="/About">About</a>
            <a class="nav-link text-white" asp-area="" asp-page="/Contact">Contact</a>
            <a class="nav-link text-white" asp-area="" asp-page="/Calculator">Calculator</a>
            <div class="mynav_dropdown">

                <button class="mynav_dropbtn">Technologys</button>
                <div class="mynav_dropdown-content">
                    <a class="nav-link text-dark" asp-area="" asp-page="/Solarpanels">Solar Panels</a>
                    <a class="nav-link text-dark" asp-area="" asp-page="/Windturbines">Wind Turbines</a>
                    <a class="nav-link text-dark" asp-area="" asp-page="/Waterpanels">Water Panels</a>
                    <a class="nav-link text-dark" asp-area="" asp-page="/Electriccar">Electric Cars</a>
                    <a class="nav-link text-dark" asp-area="" asp-page="/Fenceturbines">Fence Turbines</a>
                    <a class="nav-link text-dark" asp-area="" asp-page="/Riverturbines">River Turbines</a>
                </div>
            </div>
            <a class="nav-link text-white" asp-area="" asp-page="/Records">Records</a>

        </div>
    </header>
    <div class="container">
        <main role="main" class="pb-3">
            @RenderBody()
        </main>
    </div>

    <footer>

        <div class="my_footer">
            <p>All rights reserved. &copy; Clean Energy Portal 2022</p>
        </div>

    </footer>

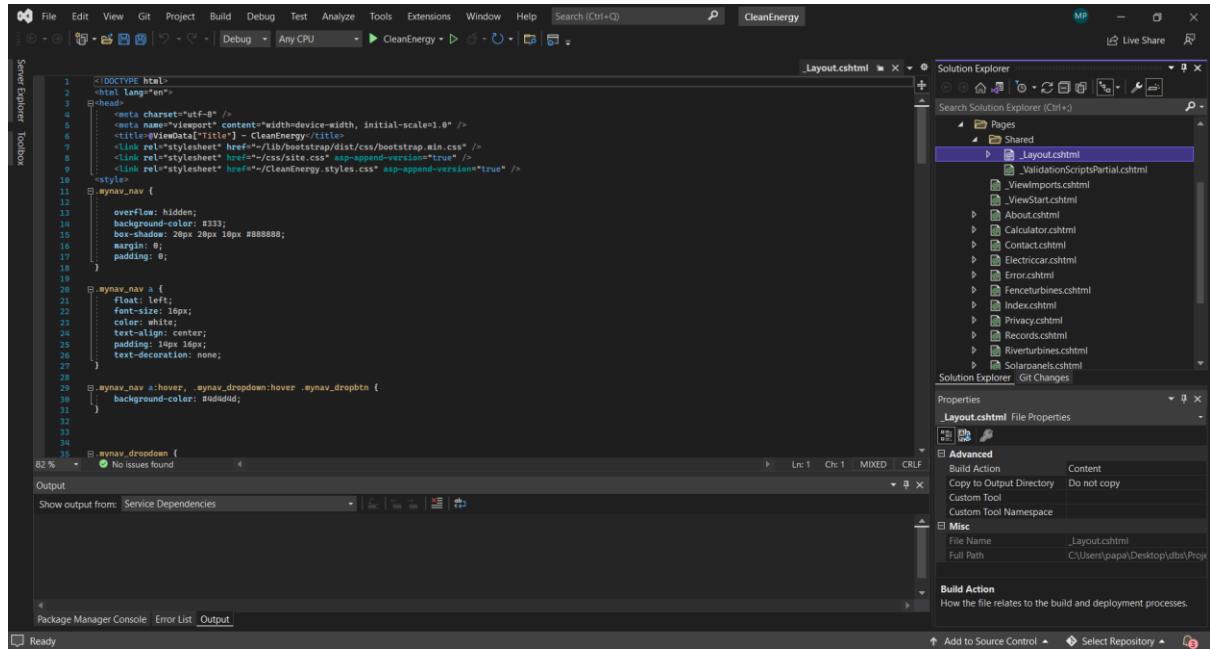
    <script src="~/lib/jquery/dist/jquery.min.js"></script>
    <script src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
    <script src="~/js/site.js" asp-append-version="true"></script>

    @await RenderSectionAsync("Scripts", required: false)
</body>
</html>

```

## 6. Application Development environment

For developing frontend of this application, Visual Studio 2022 was used and for backend SQL express with Microsoft SQL Server Management studio 18. To store code AWS CodeCommit was used. Visual Studio is widely used IDE for developing console and graphical interface applications, Windows Forms applications, websites, ASP.NET applications.



## 7. Infrastructure Development Environment

For this project Visual Studio Code was used, it is widely used code editor with support for development operations like debugging, task running, and version control. Consist of many various plugins and support many different programming languages.

```

 1  apiVersion: apps/v1
 2  kind: Deployment
 3  metadata:
 4    labels:
 5      app.kubernetes.io/name: cleanenergy
 6      app.kubernetes.io/instance: cleanenergy-instance
 7      app.kubernetes.io/version: "1.0.0"
 8      app.kubernetes.io/manage-by: kubectl
 9    name: cleanenergy
10  spec:
11    replicas: 2
12    selector:
13      matchLabels:
14        app: cleanenergy
15    template:
16      metadata:
17        labels:
18          app: cleanenergy
19      spec:
20        containers:
21          - image: 953941695125.dkr.ecr.us-east-1.amazonaws.com/cleanenergycontainer:latest
22          imagePullPolicy: Always
23          name: cleanenergy
24          resources:
25            requests:
26              cpu: "250m"
27            limits:
28              cpu: "500"
29        ports:
30          - containerPort: 80
31
32
33

```

## 8. Database Design

For this project AWS RDS SQL Server Database was used, it is relational database which contains highly structured tables, where each row reflects a data entity, and every column defines specific information field. Relational databases are build using the structured query language SQL to create, store, update or retrieve data.

Database records and table Person was created for propose of this project:

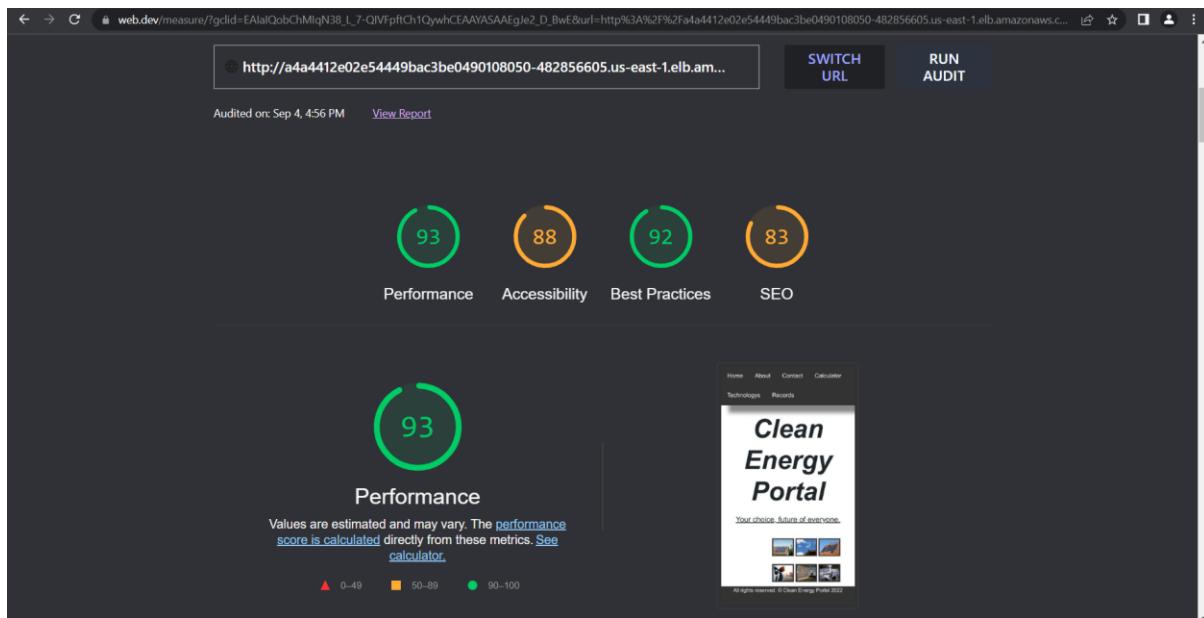
Column Name	Data Type	Allow Nulls
PersonID	Int	No
FirstName	Varchar(250)	Yes
LastName	Varchar(250)	Yes
Energy	numeric(30, 0)	Yes
Size	numeric(30, 0)	yes

## 9. Application Testing

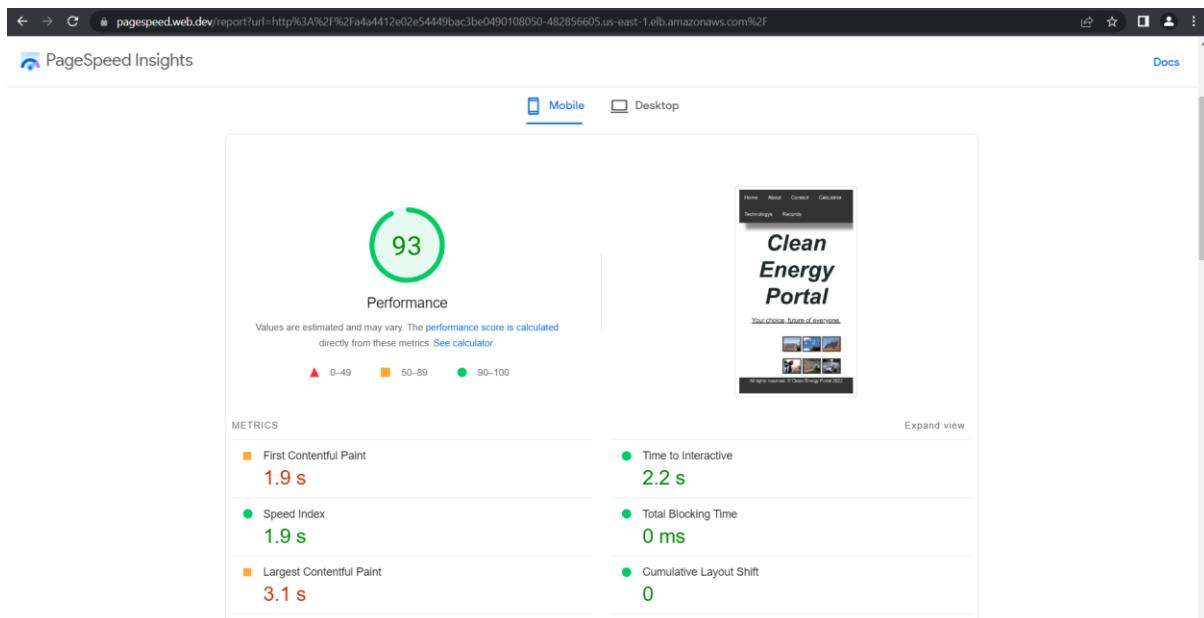
Purpose of testing is to give us confidence that such application is working correctly when interacting with users, has good performance, and all input forms work as intended.

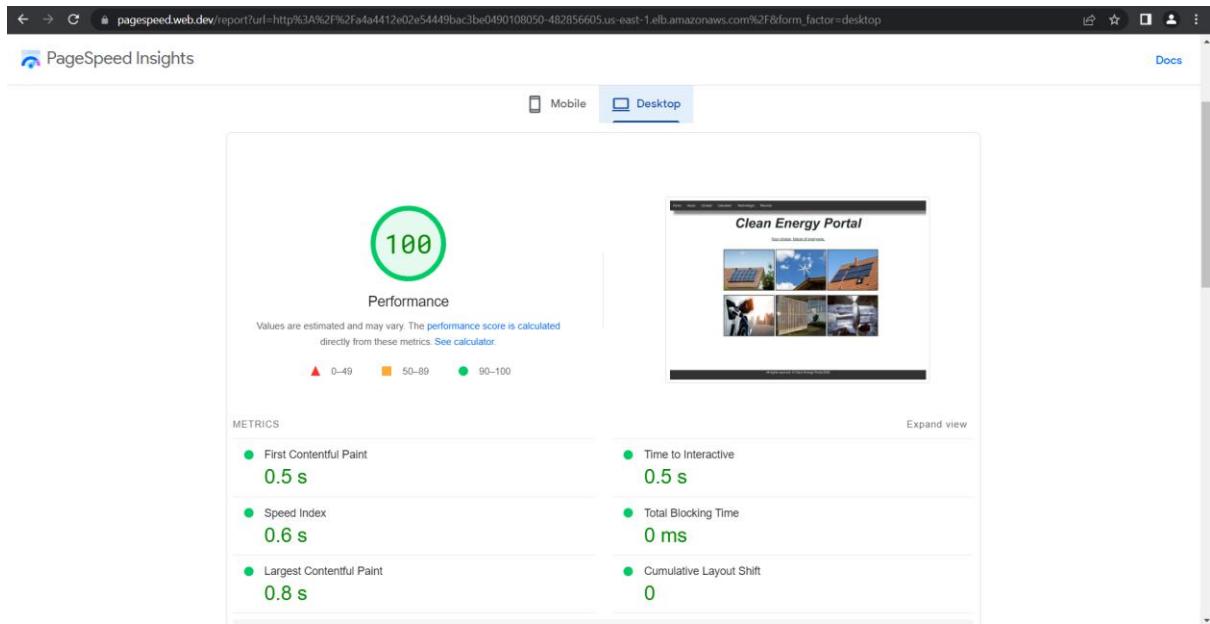
### a. Performance testing

Testing with Lighthouse from Google, it is open-source tool for running technical website audits. This tool was developed by Google, and it analyses: Performance, Progressive Web App, Accessibility, Best Practices and SEO (search website optimization).



Testing with Page Speed Insight, this is an online synthetic benchmark tool which helps in identifying performance best practices.





## b. Manual testing

In manual testing each link on website was tested and is working as supposed to:

Website link	Link	Working (yes or no)
Navigation bar home	<code>asp-page="/Index"</code>	yes
Navigation bar About	<code>asp-page="/About"</code>	Yes
Navigation bar Contact	<code>asp-page="/Contact"</code>	Yes
Navigation bar Records	<code>asp-page="/Records"</code>	Yes
Navigation bar Calculator	<code>asp-page="/Calculator"</code>	Yes
Navigation bar Solar Panels	<code>asp-page="/Solarpanels"</code>	Yes
Navigation bar Solar Water Panels	<code>asp-page="/Waterpanels"</code>	Yes
Navigation bar Water Turbines	<code>asp-page="/Waterpanels"</code>	Yes
Navigation bar Wind Turbines	<code>asp-page="/Windturbines"</code>	Yes

Navigation bar Electric Car	<code>asp-page="/Electriccar"</code>	Yes
Navigation bar Fence Turbines	<code>asp-page="/Fenceturbines"</code>	Yes
Navigation bar River Turbines	<code>asp-page="/Riverturbines"</code>	Yes
Picture Solar Panels on home site	<code>asp-page="/Solarpanels"</code>	Yes
Picture Electric Cars on home page	<code>asp-page="/Electriccar"</code>	Yes
Picture Water Turbines on home page	<code>asp-page="/Riverturbines"</code>	Yes
Picture Wind Turbines on home page	<code>asp-page="/Windturbines"</code>	Yes
Picture Fence Turbines on home page	<code>asp-page="/Fenceturbines"</code>	Yes
Picture Solar Water Panels on home page	<code>asp-page="/Waterpanels"</code>	Yes
Spider web info	<a href="https://spidersweb.pl/2022/08/plot-ktory-wytwarzaprad.html">https://spidersweb.pl/2022/08/plot-ktory-wytwarzaprad.html</a>	Yes
Blog of Centrum nauki	<a href="https://centrumnauki.eu/prad-z-plotu-chodnika-i-okiennejszyby/">https://centrumnauki.eu/prad-z-plotu-chodnika-i-okiennejszyby/</a>	Yes
BCD Energy	<a href="http://www.bcdenergy.ie/renewables-hydro-power.php">http://www.bcdenergy.ie/renewables-hydro-power.php</a>	Yes
Suneco	<a href="https://www.micro-hydro-power.com/">https://www.micro-hydro-power.com/</a>	Yes
Eco Evolution	<a href="http://www.ecoevolution.ie/small-scale-hydro.html">http://www.ecoevolution.ie/small-scale-hydro.html</a>	Yes
Caldosolar	<a href="https://caldorsolar.ie/">https://caldorsolar.ie/</a>	Yes
Wizer Energy	<a href="https://wizerenergy.ie/">https://wizerenergy.ie/</a>	Yes
AEI	<a href="https://www.aei.ie/">https://www.aei.ie/</a>	Yes
Solarstream	<a href="https://www.solarstream.ie/">https://www.solarstream.ie/</a>	Yes
AEI	<a href="https://www.aei.ie/">https://www.aei.ie/</a>	Yes
Pure Energy Technology	<a href="http://www.pet.ie/solar-heating.html">http://www.pet.ie/solar-heating.html</a>	Yes
Glenn Dimplex Ireland	<a href="https://www.glendimplexireland.com/brands/dimplex/domestic-heating-systems/solar-thermal-hot-water-systems">https://www.glendimplexireland.com/brands/dimplex/domestic-heating-systems/solar-thermal-hot-water-systems</a>	Yes

Carbon free heat	<a href="https://carbonfreeheat.ie/wind-turbines-for-sale-ireland">https://carbonfreeheat.ie/wind-turbines-for-sale-ireland</a>	Yes
Wind and Sun	<a href="https://www.windandsun.ie/product-category/wind-turbines/">https://www.windandsun.ie/product-category/wind-turbines/</a>	Yes
Sun stream energy	<a href="https://sunstreamenergy.ie/utility/wind/">https://sunstreamenergy.ie/utility/wind/</a>	Yes
Sustainable Energy Authority Ireland	<a href="https://www.seai.ie/grants/home-energy-grants/">https://www.seai.ie/grants/home-energy-grants/</a>	Yes

Testing Calculator form on Calculator page:

Test	Result	Pass
When no entries provided and click submit	First filed information that filed can't be empty	Yes
When no entries in last 2 fields	Second Filed information that field can't be empty	Yes
When no entries in last filed	Last filed information that can't be empty	Yes
When in last filed we try insert not number	Unable to place number	Yes
When in last field we try insert fractions	Information on filed "Please provide valid value. The two nearest numbers are x and x"	Yes
When all fields are correct entries	Information on web "Expected energy generation: x KWh per year."	Yes

Testing form on Records page:

Test	Result	Pass
Run page	Is not displaying records	Yes
Entering incorrect username or password	It is displaying on page "Wrong username or password"	Yes
Entering correct username and password	Displaying records from database on page	Yes

## 10. Infrastructure testing

Testing if it is possible to access database from internet by using Microsoft SQL Server Management Studio 18:

Error: "Cannot connect to databaseeksprojectv2.c0l2bn8ji4on.us-east-1.rds.amazonaws.com.

Additional information:

A network-related or instance-specific error occurred while establishing a connection to SQL Server. The Server was not found or was not accessible. Verify that the instance name is correct, and that SQL server is configured to allow remote connections. (Provider: Named Pipes Provider, error: 40 – Could not open a connection to SQL Server) (Microsoft SQL Server, Error: 53)

Network path was not found"

Testing if website is accessible: yes

Testing Continuous Integration and Continuous Deployment to see if is working as supposed to:

Doing small change to code and uploading to AWS CodeCommit and is working fine.

```
Administrator: Command Prompt - kubectl get pods -w
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Users\papa\Desktop\ dbs\Project\website\cleanenergyapp

C:\Users\papa\Desktop\ dbs\Project\website\cleanenergyapp>git add -A .

C:\Users\papa\Desktop\ dbs\Project\website\cleanenergyapp>git commit -m "small change"
[main 1d049fa] small change
 4 files changed, 8 insertions(+), 1 deletion(-)
 rewrite .vs/CleanEnergy/v17/.suo (73%)

C:\Users\papa\Desktop\ dbs\Project\website\cleanenergyapp>git push origin main
Enumerating objects: 23, done.
Counting objects: 100% (23/23), done.
Delta compression using up to 8 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (12/12), 2.38 KiB | 1.19 MiB/s, done.
Total 12 (delta 6), reused 0 (delta 0), pack-reused 0
To https://git-codecommit.us-east-1.amazonaws.com/v1/repos/cleanenergyapp
 531b555..1d049fa main -> main

C:\Users\papa\Desktop\ dbs\Project\website\cleanenergyapp>kubectl get pods -w
NAME           READY   STATUS    RESTARTS   AGE
cleanenergy-5c85599547-12q28  1/1     Running   0          24h
cleanenergy-5c85599547-18mv9  1/1     Running   0          24h
cleanenergy-6bb49bc69c-vsnmf  0/1     Pending   0          0s
cleanenergy-6bb49bc69c-vsnmf  0/1     Pending   0          0s
cleanenergy-6bb49bc69c-vsnmf  0/1     ContainerCreating   0          0s
cleanenergy-6bb49bc69c-vsnmf  1/1     Running   0          2s
cleanenergy-5c85599547-18mv9  1/1     Terminating   0          24h
cleanenergy-6bb49bc69c-shjt6  0/1     Pending   0          0s
cleanenergy-6bb49bc69c-shjt6  0/1     Pending   0          0s
cleanenergy-6bb49bc69c-shjt6  0/1     ContainerCreating   0          0s
cleanenergy-5c85599547-18mv9  0/1     Terminating   0          24h
cleanenergy-5c85599547-18mv9  0/1     Terminating   0          24h
cleanenergy-5c85599547-18mv9  0/1     Terminating   0          24h
cleanenergy-6bb49bc69c-shjt6  1/1     Running   0          2s
cleanenergy-5c85599547-12q28  1/1     Terminating   0          24h
cleanenergy-5c85599547-12q28  0/1     Terminating   0          24h
cleanenergy-5c85599547-12q28  0/1     Terminating   0          24h
cleanenergy-5c85599547-12q28  0/1     Terminating   0          24h
```

Testing if one of nodes fail another one will be deployed automatically, terminating one node and see if new is deployed to the EKS cluster, yes is working as intended, new node is deployed after 1-2 minutes.

## 11. Conclusions

Building system that is Highly Available, Secure, Redundant and Scalable, which consists automated procedures for continuous integration and continuous deployment was a big, exciting challenge. It was a fascinating adventure to bringing many different tools and building blocks to see like all these pieces work together as one system. During this project I have built first webapp in pure HTML, CSS and JavaScript realizing later that if application will be connected to database, every user can see code as JavaScript run on web browser, not on server and then I decided re-write application in ASP.NET core. Implementing Kubernetes orchestrator tool developed by Google (in AWS known as EKS) wasn't easy as it was new to me. Getting all features ready like Horizontal Pod Autoscaler which require additional pods logging to work, give me little bit of troubles and few moments of troubleshooting, finally finding that metrics-server need to be installed on Kubernetes cluster for autoscaling to work. It is nice to see how new technologies can reduce the time needed to deploy new updates or changes in your applications, by using containerization, continuous integration and continuous deployment time is reduced from days to few minutes. Developers can make changes in code many times a day and tested in minutes, if something is wrong with application previous working docker image can be deployed in minutes.

## 12. References / Bibliography

Industrial IT and Automation, 2021: <https://www.youtube.com/watch?v=iKmUMgZj-cE>

BoostMyTool, 2022: [https://www.youtube.com/watch?v=YUPg41kG\\_kw](https://www.youtube.com/watch?v=YUPg41kG_kw)

W3schools, 2022: <https://www.w3schools.com/asp/default.asp>

Sandip Das, 2019: <https://www.youtube.com/watch?v=LlisKI-gN5w>

Sandip Das, 2021: <https://www.youtube.com/watch?v=nEK7e0QUVio>

AWS, 2022: <https://docs.aws.amazon.com/eks/latest/userguide/getting-started.html>

AWS, 2022: <https://docs.aws.amazon.com/eks/latest/userguide/add-user-role.html>

AWS, 2022: <https://docs.aws.amazon.com/eks/latest/userguide/metrics-server.html>

Saur Energy International, 2022: <https://www.sairenergy.com/solar-energy-blog/here-is-how-you-can-calculate-the-annual-solar-energy-output-of-a-photovoltaic-system>

AWS, 2022: <https://docs.aws.amazon.com/whitepapers/latest/overview-deployment-options/amazon-elastic-kubernetes-service.html>

AWS, 2022: <https://docs.aws.amazon.com/eks/latest/userguide/metrics-server.html>

AWS, 2022: <https://aws.amazon.com/codecommit/>

AWS, 2022: <https://docs.aws.amazon.com/systems-manager/latest/userguide/systems-manager-parameter-store.html>

AWS, 2022: <https://aws.amazon.com/codepipeline/>

AWS, 2022: <https://aws.amazon.com/codebuild/>

AWS, 2022: <https://aws.amazon.com/ecr/>

AWS, 2022: <https://aws.amazon.com/ecr/>

AWS, 2022: <https://aws.amazon.com/cloudwatch/>

AWS, 2022: <https://aws.amazon.com/iam/>

Solargis, 2022: <https://solargis.com/maps-and-gis-data/download/ireland>