

Internship report

University diploma of technology

■ Computer Engineering

Creating a dynamic responsive website and host it on AWS (Amazon Web Services)

Realized by :

Fatima ezzahra el arrouchi

Supervised by :

Mr Laimina Chakir

Mr El ouahmani houssine

2024 - 2025

Gtatitude

As I reach the conclusion of my internship journey, I am filled with immense gratitude for the support, guidance, and opportunities that have enriched my experience. I would like to extend my heartfelt thanks to everyone who has contributed to making this internship a valuable and memorable chapter in my professional development.

First and foremost, I am deeply grateful to IP technology for providing me with this incredible opportunity. Ans its commitment to fostering a supportive and innovative environment has allowed me to learn and grow beyond my expectations.

I would like to express my sincere appreciation to my supervisor, El Ouahmani houssine. Your unwavering support, insightful feedback, and constructive guidance have been instrumental in shaping my understanding and skills. Your willingness to share your expertise and provide mentorship has been invaluable to my development.

I am also thankful to my colleagues and team members who have welcomed me warmly and made me feel like a part of the team from day one.

Additionally, I would like to extend my gratitude to my academic mentors and professors at the higher school of technology of casablanca for their continuous support and encouragement throughout my academic journey. Your teachings have laid a strong foundation that has prepared me well for this internship.

Last but not least, I want to thank my family and friends for their unwavering support and encouragement. Your belief in my abilities has been a constant source of motivation and strength.

In conclusion, this internship has been a transformative experience, and I am grateful to everyone who has played a part in making it possible. I look forward to applying the knowledge and skills I have gained to future endeavors and continuing to grow both personally and professionally.

Table of content

Chapter 1:

- Introduction
- Company presentation
- Added value of my project

Chapter 2:

- My tasks
- Fullstack
- Website deploying

Chapter 3:

- Tools and softwares used

Chapter 4:

- Conclusion

Chapter 1

→Introduction:

This report details the experience and outcomes of an internship focused on creating a dynamic and responsive website, which was then hosted on Amazon Web Services (AWS). The internship, carried out at IP technology, for a company (IngePRO maroc) which is an engineering firm specializing in infrastructure execution studies, provided the opportunity to engage with both front-end and back-end web development, utilizing technologies such as HTML, CSS, JavaScript, PHP, MySQL, and the Bootstrap framework. The objective was to enhance the company's online presence, improve client engagement, and streamline communication through a well-structured and accessible website. This report will discuss the various stages of the project, the tools and technologies employed, and the added value this project has brought to IngePro Maroc.

→Company presentation:

Since it was created in 2013, IngeProMaroc, an engineering firm specializing in infrastructure

execution studies, has been able to create a true durable relationship of trust with its partners, both contractors and customers.

Excellence is always the watchword of the IngePro Maroc teams.

The technical base of the design office means that technical analyses and advice are always the most effective.

Upon request from its clients and partners, IngePro Maroc has supplemented its range of services

to provide technical and financial oversight of large-scale projects.

IngePro Maroc contributed to the execution studies and complete oversight of major infrastructure

projects in Morocco as well as abroad such as in other countries, and so IngePro Maroc was

formed.

With its experience, and its partnerships with large contractor or engineering groups, IngePro

Maroc has persuaded customers and investors of the utility of setting up turnkey projects making it

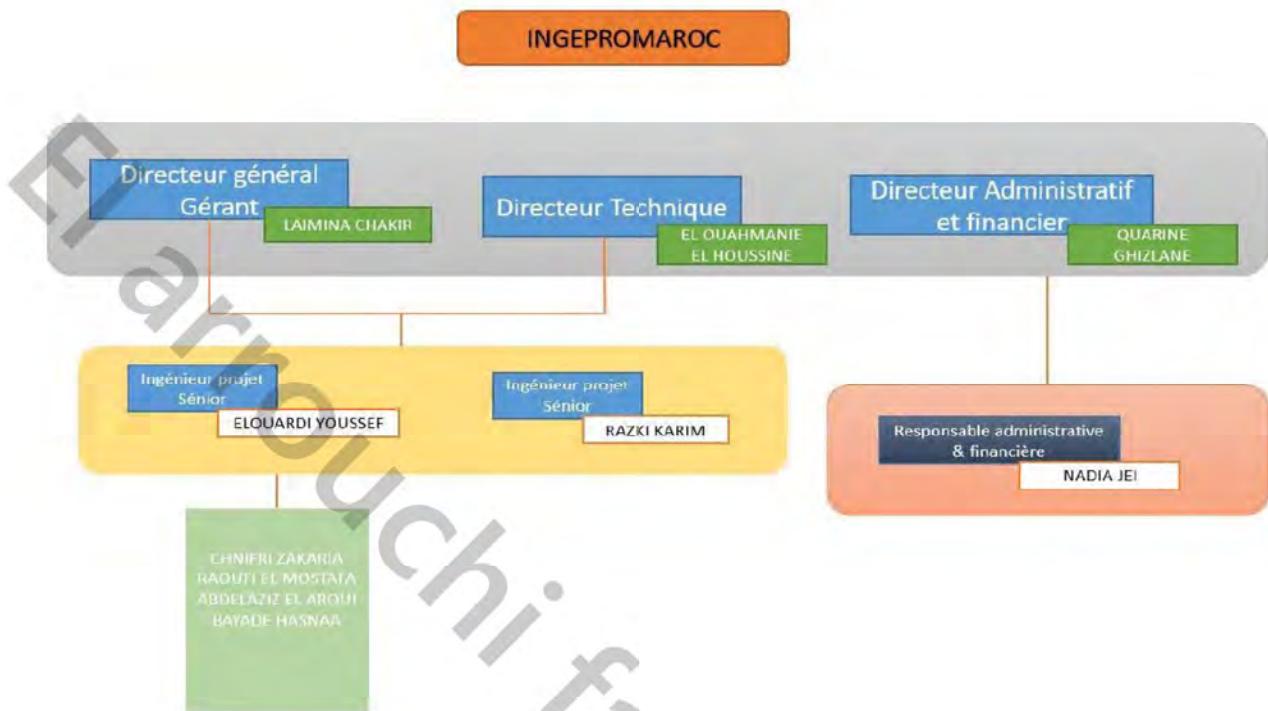
possible to identify:

- Infrastructure needs;
- Overall project design and their integration;
- Techniques to be put in place;
- Specific engineering partners to take on;
- Partner contractors that are able to carry out the work;
- Financial partners to meet budgets.

Bringing all these skills together lets tailored projects come to completion, technically and financially under control, and to fully satisfy the customer.

Setting up infrastructure projects concerns all fields such as ports, airports, roads, hydraulics, electricity...

➤ ORGANIZATION CHART INGEPROMAROC



→**Added value of my project:**

My project will elevate the company's success and increase market demand by showcasing IngeproMaroc's projects, services, and contact information. This will simplify the decision-making process for potential clients, making it easier for them to choose to work with them. Additionally, it will save time and effort for both the company's staff and clients. The company will benefit from an organized database containing client contact information, allowing for efficient communication of updates and new projects to those interested in our news. I decided to host our website on AWS using S3, EC2, and Route 53 because of its outstanding reliability, scalability, and security. AWS S3 gives us a dependable and expandable storage solution for all our project data, ensuring it's always available and secure. With EC2, we can easily handle different levels of traffic and workloads, thanks to its flexible and scalable compute capacity. Route 53 ensures that our website is accessible and performs well by efficiently routing internet traffic. Altogether, these AWS services create a powerful and efficient hosting environment that supports our growing business needs.

Chapter 2

→My tasks:

During my internship, I had the exciting task of creating a website from scratch. I used various languages like HTML, CSS, JavaScript, PHP, and MySQL, as well as the Bootstrap framework. These technologies were chosen because they offer a solid foundation for building responsive, user-friendly, and dynamic websites. HTML and CSS handle the structure and style, JavaScript brings interactivity, PHP and MySQL manage server-side operations and database interactions, and Bootstrap ensures a consistent and responsive design.

I also worked on finding the best front-end design that met my supervisor's expectations, essentially acting as a full-stack developer. Additionally, I delved into cloud computing to host the website, making sure it was always available, secure, easy to navigate and fast-loading. This experience gave me a well-rounded skill set in both development and deployment, ensuring the website provided a great user experience for our clients.

→ Full-Stack:

- **Front-End**

The image shows a screenshot of a website homepage. At the top, there is a brown header bar with the 'Pro Maroc' logo on the left and navigation links for 'Home', 'About US', 'Projects', 'Fields of activity', and 'Contact US' on the right. Below the header, the main content area has a brown background. On the left side, there is a large white text box containing the heading 'Crafting project specific solutions with expertise.' and a subtext: 'We're a creative company that focuses on establishing long-term relationships with customers.' At the bottom of this section are two white buttons: 'Explore Now' and 'Contact Us'. To the right of the text box is a large, rectangular image showing a busy port or industrial area with many shipping containers, several large yellow cranes, and a long pier extending into the water under a clear blue sky.

The navigation bar features IngePromaroc's logo in the top-left corner, with the navigation links aligned to the right. The links are written in Lato font.

For the background image, I chose an orangy-brown hue that complements the logo colors and the overall website theme.

To the right of the main text, there is a slideshow showcasing three images of the company's projects. The slideshow includes previous and next arrows, allowing clients to navigate between images. Additionally, a scroll bar is included to accommodate longer content and provide smooth navigation.

Under the main headlines, there are two primary buttons:

- **Explore Now:** This button likely leads to a page showcasing the company's projects.
- **Contact Us:** This button likely opens a contact form for visitors to reach out to the company.

The screenshot shows a website layout. At the top, there is a white header with the IngePromaroc logo on the left and navigation links for Home, About US, Projects, Fields of activity, and Contact us. Below the header is a large, semi-transparent brown banner with the text "IngePromaroc Maroc" and "Ingénierie et Construction". The main content area is white and features a heading "WHAT WE DO?" in orange, followed by a sub-heading "The service we offer is specifically designed to meet your needs." Below this, there are four service cards with titles and descriptions:

Project Design	Execution Studies and Drawings	Technical Works Assistance and Supervision	Special Formwork Studies
Prefeasibility, conceptual, feasibility, and detailed preliminary studies to ensure thorough project planning.	Designing 3D computer models, creating dimensioning calculation notes, and preparing execution drawings for precise project implementation.	Providing continuous technical support, ensuring quality execution, adherence to deadlines, financial control, and efficient jobsite organization.	Designing special metal formwork for reinforced concrete structures, optimizing production costs and schedules while maintaining quality.

The background is now white, which indicates that the navbar changes appearance (from brown to white) when scrolling down the page to maintain readability against various backgrounds, and the logo also turns to a dark logo since the navbar is white.

The text "WHAT WE DO?" is displayed in orange, which matches IngePromaroc's general theme. It is set in a smaller font above the main heading.

Below the heading, there are four cards, each describing a different service offered by the company.



The section has a brown background with subtle abstract patterns, maintaining the overall color scheme consistent with the previous sections of the website that highlights the company's key statistics and achievements.

Ingenieria
Maroc
Home
About US
Projects
Fields of activity
Contact US

PROJECTS

Check out some of our awesome projects
with creative ideas and great design.



New Shipyard at the Port of Casablanca Lot 2

Located in Casablanca, Morocco, this project for SDMAGEC involves the construction of a boat lift, a link dock, and a transfer and storage platform.



Construction of a Cement Plant

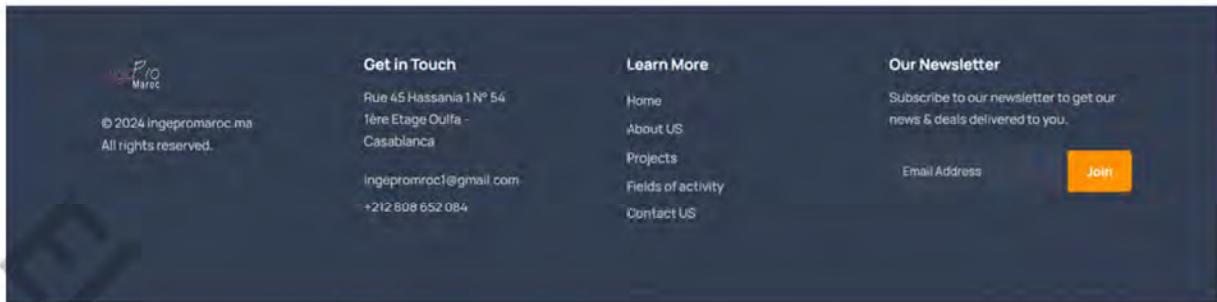
Situated in Akoga, Equatorial Guinea, and commissioned by SDMAGEC CE, this project entails the construction of a cement plant with a production capacity of 1,000,000 tons.



Warehouse and Shops for DMA Company (Michelin Importer)

In Casablanca, Morocco, for the client D.M.A., this project includes the construction of an 8,000 m² warehouse and plant for storage.

Three projects are displayed side by side in a grid format. Each project has an image, a title, and a brief description beneath it. When you hover over the project titles, they change color to orange, indicating interactivity. This is likely a visual hint to signal that the titles are clickable.



The footer contains the IngePro Maroc logo, with the copyright notice for the company displayed underneath it.

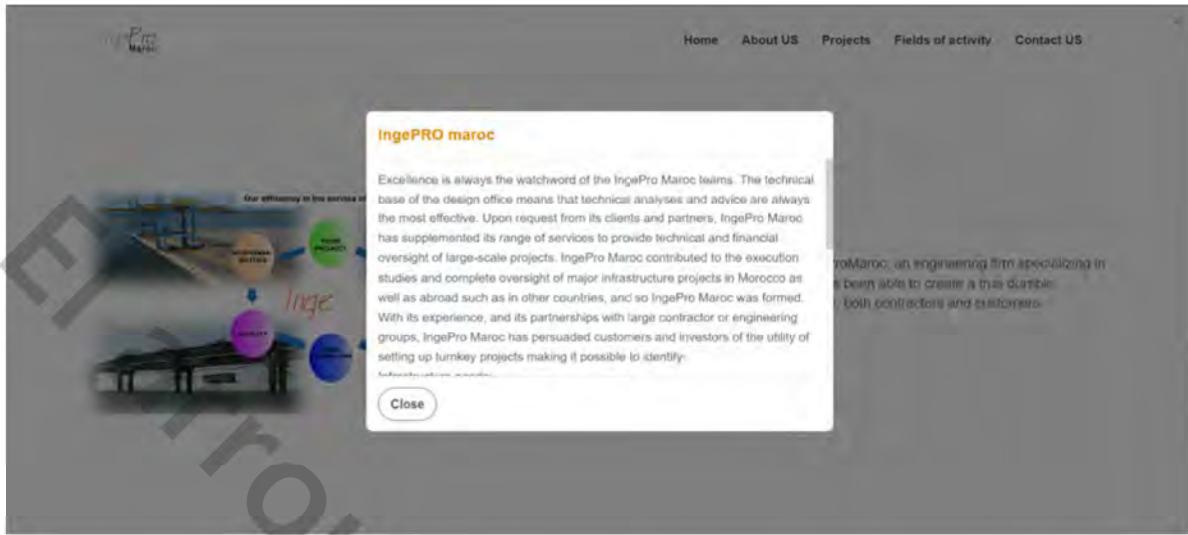
Next to the logo, there is contact information and the company's email address.

Beside this, there is a "Learn More" section that includes the navigation links.

On the right side of the footer, there is an input form for the newsletter, allowing visitors to stay updated with the company's latest news.

--! THE FOOTER IS PRESENT IN ALL THE PAGES !--

The page has a header with the IngePro Maroc logo and navigation links for Home, About US, Projects, Fields of activity, and Contact US. The main content is titled "ABOUT US". It features a graphic with four colored circles (green, red, purple, yellow) connected by arrows, labeled "YOUR PROJECT", "SKILLS", "QUALITY", and "INNOVATION". Above the graphic, a banner says "Our efficiency in the service of your projects". Below the graphic, a section titled "IngePRO maroc" describes the company's history and values, with a "Read More" button.



On the left side, there is an image that illustrates the company's focus areas.

A block of text provides an introduction to IngePro Maroc. Below the text, there is an orange "Read More" button.

When you hover over the images on the left, they enlarge slightly, drawing more attention to them.

Upon clicking the "Read More" button, the screen dims, and a pop-up window appears in the center of the screen, providing more detailed information about IngePro Maroc.

The modal includes a "Close" button at the bottom, which allows the user to dismiss the modal and return to the main page.

A screenshot of the 'Our Projects' section of the IngePro Maroc website. At the top, there is a navigation bar with links: Home, About US, Projects, Fields of activity, and Contact US. Below the navigation, the section is titled 'Our Projects'. There are three main project cards displayed, each with a thumbnail image, a title, a brief description, and a 'Read More' button. A red arrow points from the text 'on hover' to the second project card, indicating that hovering over it will cause it to expand. The first project card is titled 'New Shipyard at the Port of Casablanca Lot 2' and describes the construction of a boat lift, link dock, and transfer and storage platform. The second project card is titled 'Construction of a Cement Plant' and describes a cement plant with a production capacity of 1,000,000 tons. The third project card is titled 'Warehouse and Shops for DMA Company (Michelin Importer)' and describes the construction of an 8,000 m² warehouse and plant. At the bottom left, there is a link 'com/project5.php'.

The projects are presented in cards with a soft border radius. Each card slightly enlarges and gains a soft shadow beneath it when hovered over.

Elarrouchi flimza ezzahra

PROJECT DESCRIPTION

2 hangars for storage of phosphates with 30,000 T capacities. Structure in hybrid prefabricated elements.

Scope: 35 m.

Length: 150 m.

Width: 36 m

Bottom height: 30 m

Nature of execution studies:

- Hall installation, earthworks, foundation, footing, sill and galleries.
- Bov: Prefabricated beams.
- Hoppers, cover, framing, conveyors, towers.



DESCRIPTION OF THE MISSION

Execution studies for mixed reinforced concrete structure.

When clicking on a project, a slideshow appears in the center, featuring previous and next arrows that allow clients to navigate between images. On either side of the slideshow, there are detailed descriptions—one outlining the mission and the other describing the project.



[Home](#) [About US](#) [Projects](#) [Fields of activity](#) [Contact US](#)

OUR FIELDS OF ACTIVITY

IngePro Maroc. An experienced team that meets client expectations and requirements, and puts attentiveness, technical knowledge and respect for commitments at the heart of every collaboration.

- PROJECT DESIGN
- PREPARATION OF TENDER PACKAGES
- EXECUTION STUDIES AND DRAWINGS
- WORKING METHODS
- SPECIAL FORMWORK STUDIES
- TECHNICAL WORKS ASSISTANCE AND SUPERVISION
- WORKS MANAGEMENT
- ENVIRONMENTAL AND COASTAL ENGINEERING CONSULTING

[Get in Touch](#) [Learn More](#) [Our Newsletter](#)

The screenshot shows the 'Fields of activity' section of the IngePro Maroc website. At the top, there is a horizontal navigation bar with links: Home, About US, Projects, Fields of activity, and Contact US. Below this, there are several expandable panels represented by grey boxes with a '+' sign in the top right corner. The first panel is expanded, showing the following content:

PREPARATION OF TENDER PACKAGES

Our engineering and design team carry out all types of studies for concrete and metal structures for new or rehabilitated structures.

This type of mission consists of:

- Designing computer models on three-dimensional finite element calculation software
- Editing all dimensioning calculation notes in accordance with the regulations in force
- Preparing all types of execution drawings

WORKING METHODS

With there being a great number of construction methods, IngePro Maroc, working closely with contractors, studies the appropriate construction process so contractors can plan the most efficient jobsite organization considering the following elements:

- The options required by the client in the contract documents
- Jobsite access and installation possibilities
- The company's "capacité" that materialize in two ways:
 - The skills and experience of the company's men
 - The equipment available or usually used in the company

SPECIAL FORMWORK STUDIES

TECHNICAL WORKS ASSISTANCE AND SUPERVISION

WORKS MANAGEMENT

ENVIRONMENTAL AND COASTAL ENGINEERING CONSULTANT

In the fields of activity section, collapsible panels expand to reveal detailed descriptions when clicked. When hovered over, the panels change to a darker shade.

The screenshot shows the contact form section of the IngePro Maroc website. At the top, there is a horizontal navigation bar with links: Home, About US, Projects, Fields of activity, and Contact US. Below this, the title 'CONTACT US FOR A FUTURE COLLABORATION' is centered. The form consists of several input fields:

Name: [Input field]

Company: [Input field]

Role: [Input field]

Country: [Input field] **Number:** [Input field]

E-mail: [Input field] **Choose File:** [Choose File button] **No file chosen**

Documentation needs: [Input field]

Groupe presentation:

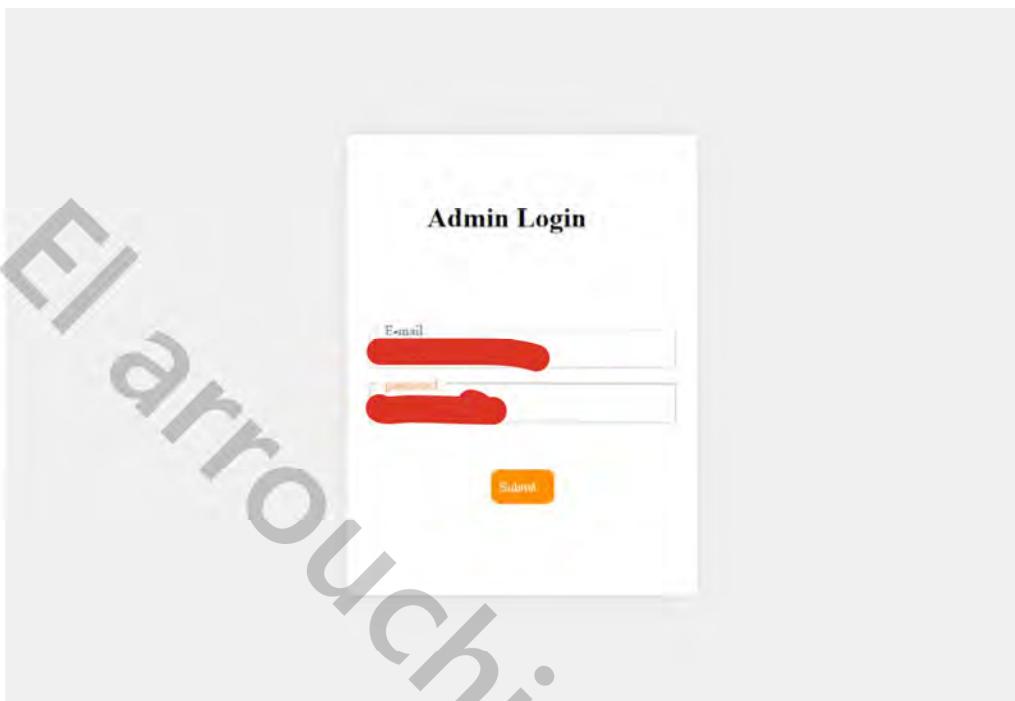
Please describe your needs: [Text area]

INSA M: [Input field]

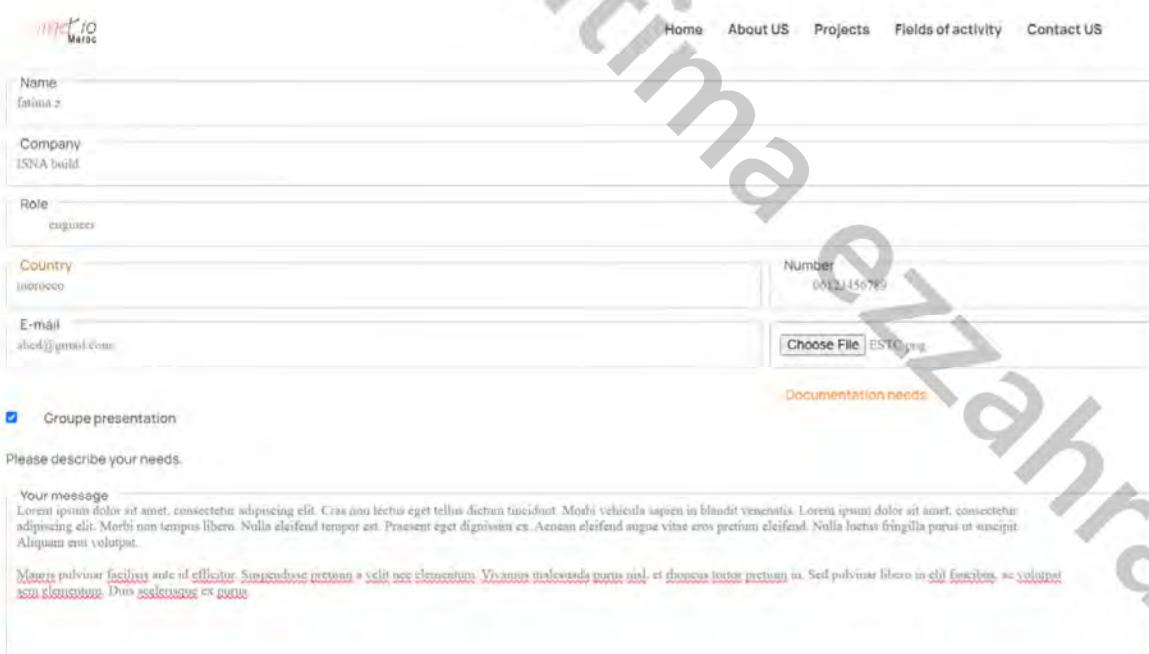
When a user clicks on an input field in the contact form, the label associated with that field floats above the input box and changes its color to orange.

The admin has his own session inside the website where he can access information from the database via this link

<https://www.ingepromaroc1.com/login.php>



After entering the credentials the admin is authentified and can access his database infos.



A screenshot of a user profile form. At the top left is a logo for 'M&P/IQ Maroc'. At the top right are navigation links: Home, About US, Projects, Fields of activity, and Contact US. The form consists of several input fields:

- Name: fatima z.
- Company: ISNA build
- Role: engineer
- Country: morocco
- E-mail: abed@gmail.com
- Number: 06121450789

Below the form are two buttons: 'Choose File' (with 'ESTC.jpg' selected) and 'Documentation needs'.

Groupe presentation

Please describe your needs.

Your message

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras non lectus eget tellus dictum tincidunt. Morbi vehicula sapien in blandit venenatis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi non tempus libero. Nulla eleifend tempor est. Praesent eget dignissim ex. Aenean eleifend augue vitae eros pretium eleifend. Nulla locutus fringilla porus ut suscipit. Aliquam erat volutpat.

Mauris pulvinar facilisis ante id efficitur. Suspendisse pretium a velit nec elementum. Vivamus malesuada turpis nisl, et rhoncus tortor pretium in. Sed pulvinar libero in efficitur, ac volutpat nisi elementum. Duis vel erat euismod.

Client Information

ID	Name	Company	Role	Country	Number	Email	Documentation	Group Presentation	Client Needs	Submission Time	Action
1	fatima	ISNA	build	06123456780	morocco	06123456789	abcd@gmail.com	ESTC.png	<p>Loreum ipsum dolor sit amet, consectetur adipiscing elit. Cras non lectus eget tellus dictum tincidunt. Morbi vehicula sapien in blandit venenatis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi non tempus libero. Nulla eleifend tempor est. Praesent eget dignissim ex. Aenean eleifend augue vitae eros prelum eleifend. Nulla luctus fringilla purus ut suscipit. Aliquam erat volutpat. Mauris pulvinar facilisis ante id efficitur. Suspendisse pretium a velit nec elementum. Vivamus malesuada purus nisl, et rhoncus tortor prelum in. Sed pulvinar libero in eii faubibus, ac volutpat sem elementum. Duis scelerisque ex purus.</p>	2024-08-09 13:51:43	<button>Done</button>

After the form is submitted, all the submitted information is displayed in the admin's session. Each entry includes a "Done" button, which allows the staff to differentiate between clients who have already been contacted and those who have not yet been reached.

The admin section features a navigation bar with the company's logo in the top-left corner. On the right side, there are two links: one for the Newsletter page and another for Client Information. Additionally, there is a Logout button that ends the admin session.

ID	Name	Company	Role	Country	Number	Email	Documentation	Group Presentation	Client Needs	Submission Time	Action
1	fatima	ISNA	build	06123456780	morocco	06123456789	abcd@gmail.com	ESTC.png	<p>Loreum ipsum dolor sit amet, consectetur adipiscing elit. Cras non lectus eget tellus dictum tincidunt. Morbi vehicula sapien in blandit venenatis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi non tempus libero. Nulla eleifend tempor est. Praesent eget dignissim ex. Aenean eleifend augue vitae eros prelum eleifend. Nulla luctus fringilla purus ut suscipit. Aliquam erat volutpat. Mauris pulvinar facilisis ante id efficitur. Suspendisse pretium a velit nec elementum. Vivamus malesuada purus nisl, et rhoncus tortor prelum in. Sed pulvinar libero in eii faubibus, ac volutpat sem elementum. Duis scelerisque ex purus.</p>	2024-08-09 13:51:43	<button>Done</button>

After clicking the "Done" button, the entire row is crossed out, and the button disappears.



The screenshot shows the "Newsletter Subscribers" section in the admin interface. It has a header with "Newsletter Subscribers" and navigation links for newsletter, client info, and logout. Below is a table with one row of data:

ID	Email	Action
1	mimouuu@gmail.com	<button>Done!</button>

After submitting an email in the footer, the email is recorded in the database and then displayed in the admin section.

As shown, the email is recorded, and when the "Done" button is clicked, the entire row is crossed out, and the button disappears. The staff clicks the button after saving the email in their contact list.

The screenshot shows the "Newsletter Subscribers" section in the admin interface. It has a header with "Newsletter Subscribers" and navigation links for newsletter, client info, and logout. Below is a table with one row of data, which is visually crossed out (strikethrough) compared to the previous screenshot:

ID	Email	Action
1	mimouuu@gmail.com	<button>Done!</button>

On smaller screens the website is responsive and **on mobile** the website looks like this :

IngePro Maroc

Crafting project specific solutions with expertise.

We're a creative company that focuses on establishing long-term relationships with customers.

[Explore Now](#) [Contact Us](#)



ingepronaroc.com/contact

WHAT WE DO?

The service we offer is specifically designed to meet your needs.

Project Design
Prefeasibility, conceptual, feasibility, and detailed preliminary studies to ensure thorough project planning.

Execution Studies and Drawings
Designing 3D computer models, creating dimensioning calculation notes, and preparing execution drawings for precise project implementation.

Technical Works Assistance and Supervision
Providing continuous technical support, ensuring quality execution, adherence to deadlines, financial control, and efficient jobsite organization.

IngePro Maroc

Special Formwork Studies
Designing special metal formwork for reinforced concrete structures; optimizing production costs and schedules while maintaining quality.

515 Completed Projects	513 Happy Customers
7 Expert Engineers	11 Years Of Experience

PROJECTS

Check out some of our awesome projects with creative ideas and great design.



New Shipyard at the Port of Casablanca Lot 2

Located in Casablanca, Morocco, this project for SOMAGEC involves the construction of a boat lift, a link dock, and a transfer and storage platform.



Construction of a Cement Plant

Situated in Akoga, Equatorial Guinea, and commissioned by SOMAGEC GE, this project entails the construction of a cement plant with a production capacity of 1,000,000 tons.



Warehouse and Shops for DMA Company (Michelin Importer)

In Casablanca, Morocco, for the client D.M.A, this project includes the construction of an 8,000 m² warehouse and plant for storage.

ingePro Maroc

© 2024 ingepromaroc.ma
All rights reserved.

Get in Touch
Rue 45 Hassania 1 N° 54 1ère Etage Oulfa - Casablanca
ingepromroc1@gmail.com
+212 808 652 084

Learn More
[Home](#)
[About US](#)
[Projects](#)
[Fields of activity](#)
[Contact US](#)

Our Newsletter
Subscribe to our newsletter to get our news & deals delivered to you.

Email Address: Join

ingePro Maroc

Home **About US** **Projects** **Fields of activity** **Contact US**

Contact Us



ingepromroc1@gmail.com
+212 808 652 084

FOOTER

Its included in all
the sections

NAVBAR

ABOUT US



IngePRO maroc

Since it was created in 2013, IngeProMaroc, an engineering firm specializing in infrastructure execution studies, has been able to create a true durable relationship of trust with its partners, both contractors and customers.

[Read More](#)

IngePRO maroc

Excellence is always the watchword of the IngePro Maroc teams. The technical base of the design office means that technical analyses and advice are always the most effective. Upon request from its clients and partners, IngePro Maroc has supplemented its range of services to provide technical and financial oversight of large-scale projects. IngePro Maroc contributed to the execution studies and complete oversight of major infrastructure projects in Morocco as well as abroad such as in other countries, and so IngePro Maroc was formed. With its experience, and its partnerships with large contractor or engineering groups, IngePro Maroc has persuaded customers and investors of the utility of setting up turnkey projects making it possible to identify:

Infrastructure needs;

Overall project design and their integration;

Techniques to be put in place;

Specific engineering partners to take on;

Partner contractors that are able to carry out the work.

[Close](#)

Our Projects



New Shipyard at the Port of Casablanca Lot

2

Construction of boat lift, link dock, and transfer and storage platform

Construction of a Cement Plant

Cement plant with a production capacity of 1,000,000 tons



Warehouse and Shops for DMA Company (Michelin Importer)

Construction of an 8,000 m² warehouse and plant



www.ingepromaroc1.com/project5.php

OUR FIELDS OF ACTIVITY

IngePro Maroc, An experienced team that meets client expectations and requirements, and puts attentiveness, technical knowledge and respect for commitments at the heart of every collaboration.

PROJECT DESIGN

PREPARATION OF TENDER PACKAGES

EXECUTION STUDIES AND DRAWINGS

WORKING METHODS

SPECIAL FORMWORK STUDIES

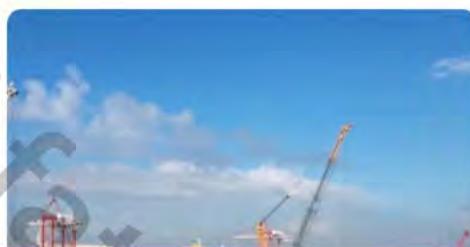
TECHNICAL WORKS ASSISTANCE AND SUPERVISION

WORKS MANAGEMENT

ENVIRONMENTAL AND COASTAL ENGINEERING CONSULTING

DESCRIPTION OF THE MISSION

Studies and execution drawings, mold/formwork drawings, lifting beam drawings and work methodology. Technical assistance and site preparation. Assistance with creation of the site facility drawing. Establishment of needs in prefabrication / storage / stacking areas and location on the drawings of site installation areas. Prefabrication methodology. Metal formwork fabrication drawings. Lifting beam fabrication drawings. Methodology of installation of prefabricated components.



PROJECT DESCRIPTION

2 docks on piles for the lifting platform
Length: 200 ml / platform.
Platform width: 25.50 m and 17.50 m + 6.50 m connection.
Dimension of upper course: +6.00 m/zh. Link dock and slab bridge:
Length: 80 ml.
Height: 16.70 / 17.70 m under distribution slab.
Seating dimension slab bridge section: -11.00 / -12.00 m/zh.
Seating dimension current section: -8.70 / -9.70 m/zh.
Dimension of upper course: +6.00 m/zh
Ferry and Ship storage area:

PREPARATION OF TENDER PACKAGES

EXECUTION STUDIES AND DRAWINGS

Our engineering and design team carry out all types of studies for concrete and metal structures for new or rehabilitated structures.
This type of mission consists of:

- Designing computer models on three-dimensional finite element calculation software.
- Editing all dimensioning calculation notes in accordance with the regulations in force
- Preparing all types of execution drawings

WORKING METHODS

SPECIAL FORMWORK STUDIES

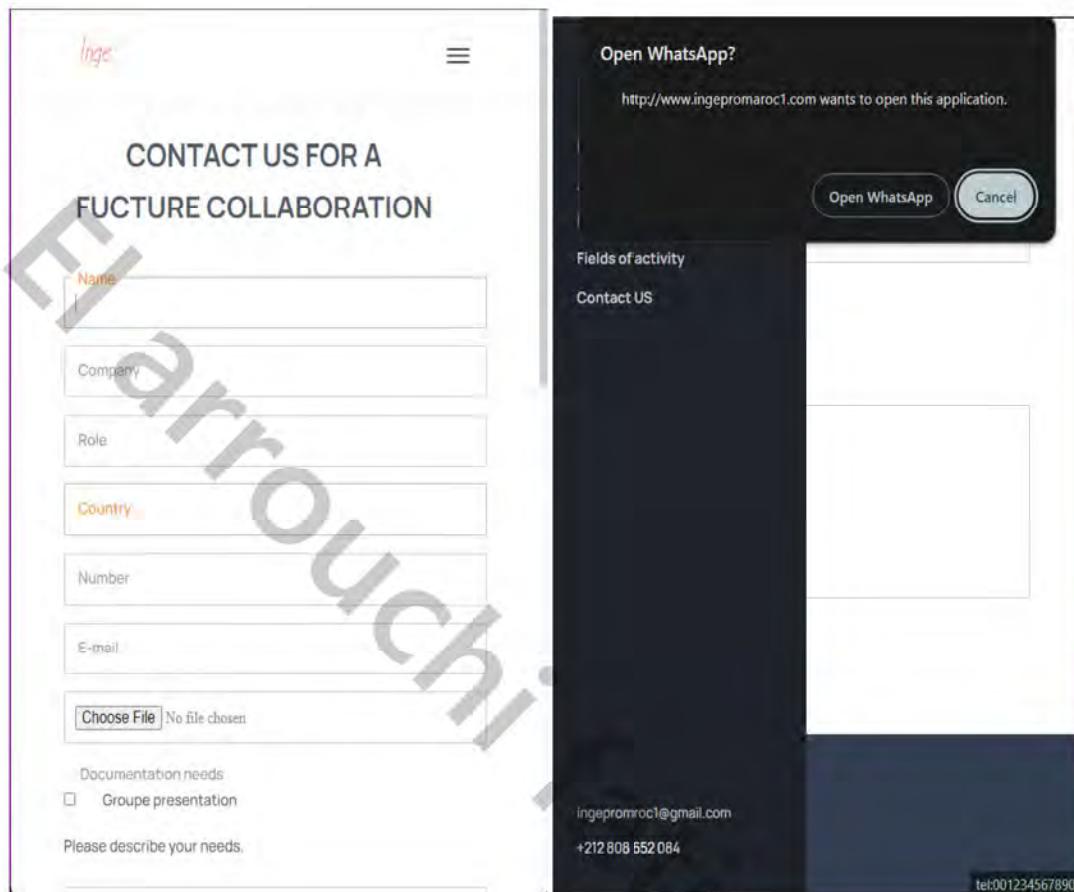
TECHNICAL WORKS ASSISTANCE AND SUPERVISION

Through a comprehensive service offering including execution studies and drawings, IngePro Maroc GROUP deploys continuous technical assistance during the work, whether assistance to project management, to the prime contractor or to subcontractors.

Our teams let contracting authorities benefit from acquired "field" experience and a real commitment.

WORKS MANAGEMENT

ENVIRONMENTAL AND COASTAL ENGINEERING CONSULTING



When we click on the phone number it takes us to whatssap

And when we click on the email it takes us to the mail box directly.

• Back-end

Table	Action	Lignes	Type	Interclassement	Taille	Perte
clients_info	Parcourir Structure Rechercher Insérer Vider Supprimer	1	InnoDB	utf8mb4_general_ci	16,0 kio	-
newsletter	Parcourir Structure Rechercher Insérer Vider Supprimer	1	InnoDB	utf8mb4_general_ci	16,0 kio	-
		2	InnoDB	utf8mb4_general_ci	32 kio	0 o

The image displays two screenshots of the phpMyAdmin web application interface, both titled "localhost / 127.0.0.1 / ingepro_maroc".

Screenshot 1 (Top): Shows the "clients_info" table in the "ingepro_maroc" database. The table has columns: ch_id, name, company, role, country, number, E-mail, documentation, grp_presentation, clt_needs, submission_time, and done. One row is visible:

ch_id	name	company	role	country	number	E-mail	documentation	grp_presentation	clt_needs	submission_time	done
1	fatma	ISNA build	06123456789	morocco	06123456789	abcd@gmail.com	ESTG.png	Lorem ipsum dolor sit amet, consectetur adipiscing elit.		2024-08-09 13:51:43	

Screenshot 2 (Bottom): Shows the "newsletter" table in the same database. The table has columns: id, email, and done. One row is visible:

	id	email	done
	1	mimouuu@gmail.com	1

During my internship, I focused on backend development using **PHP**, **MySQL**, and **PHPMyAdmin**. These tools were fundamental in creating a dynamic and responsive web application that efficiently handles data storage and retrieval.

To maintain a clean and organized codebase, I utilized PHP's include function, specifically `include('navbar.php')` & `include('navbar_admin.php')`, & `include('footer.php')`, across all web pages. This approach allowed me to reuse the same navigation bar throughout the site, ensuring consistency in the user interface and reducing code duplication. By including the navbar.php file, any updates to the navigation could be made in a single place, automatically reflecting across all pages.

The database structure was designed using **MySQL**, and **PHPMyAdmin** was employed to manage and manipulate the database efficiently. The database included tables like clients_info and newsletter, which was critical for storing and managing client data.

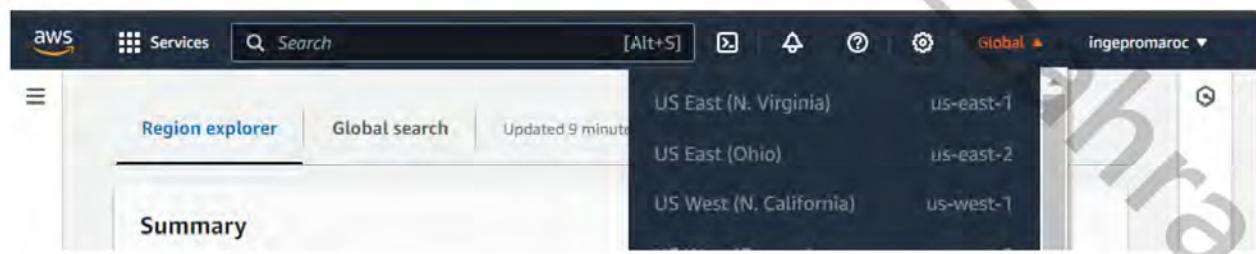
Database Connection: I established a secure connection to the MySQL database using a separate db_conn.php file. This script was included in all necessary files to ensure a centralized and consistent database connection method.

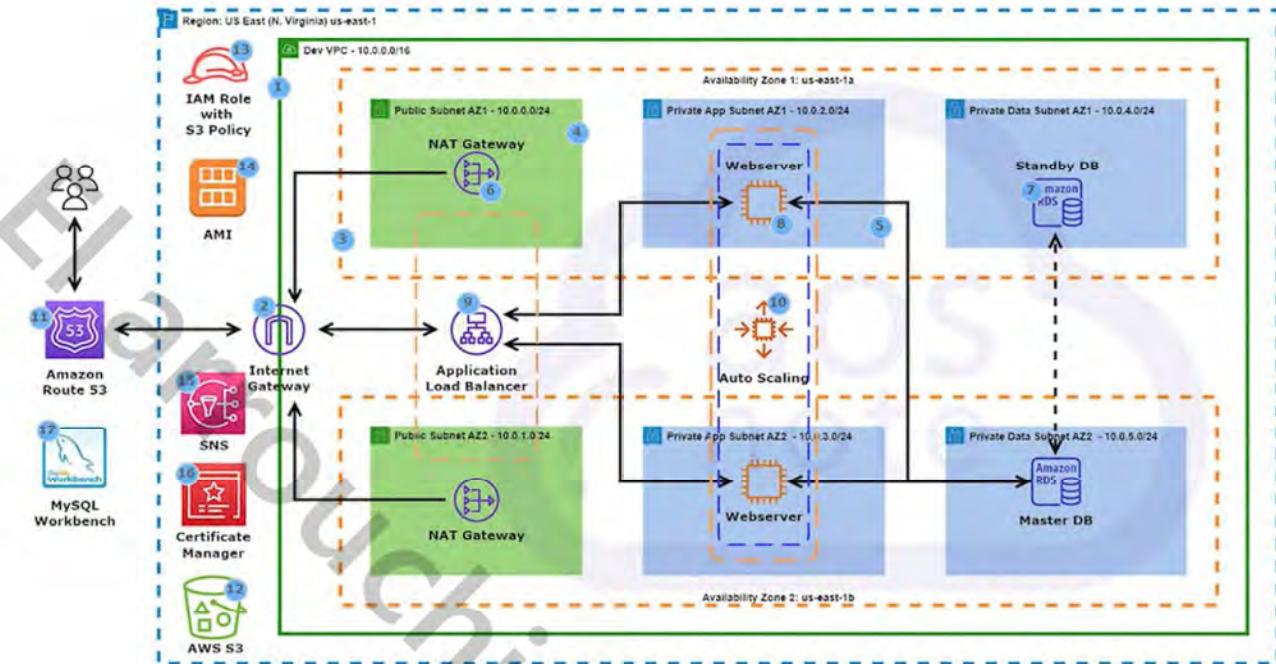
Data Handling: The client_info.php script was responsible for interacting with the clients_info table. It retrieved and displayed client information dynamically on the frontend, allowing for real-time data presentation without manual HTML updates. And the same process was done to newsletter.php which is responsible of the newsletter table.

- **Website deploying**

In order to deploy our dynamic website that interacts with users by generating content in real-time, often based on user input or data from a database, we will use AWS which is a comprehensive cloud computing platform provided by Amazon that offers a wide range of services, including computing power, storage, and networking.

First signUP into AWS and fill out the informations, then for the region I choose us-east-1(north virginia) from the drop-down.





In the following deploying steps we will be using this architecture.

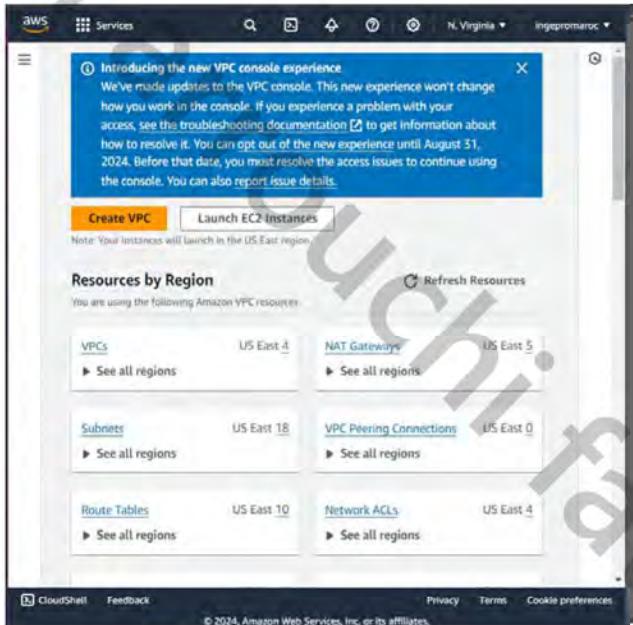
Steps:

1. Build a 3 Tier AWS Network VPC
2. Create a AWS Nat Gateway
3. Create Security Groups
4. Launch a MySQL RDS Instance
5. Create S3 Bucket and Upload File
6. Create an IAM Role with S3 Policy
7. Create a Key Pair in AWS
8. Launch a Setup EC2 Instance
9. Use MySQL Workbench to Import Data Into an RDS Database
10. Install a Dynamic Website on an EC2 Instance (Lamp Stack)
11. Create an AMI
12. Create an Application Load Balancer
13. Register a New Domain Name in Route 53
14. Create a Record Set in Route 53
15. Register an SSL Certificate in AWS Certificate Manager
16. Secure a Website or Web Application with ACM SSL Certificate
17. SSH Into an EC2 Instance in the Private Subnet
18. Create an AMI
19. Create Auto Scaling Group

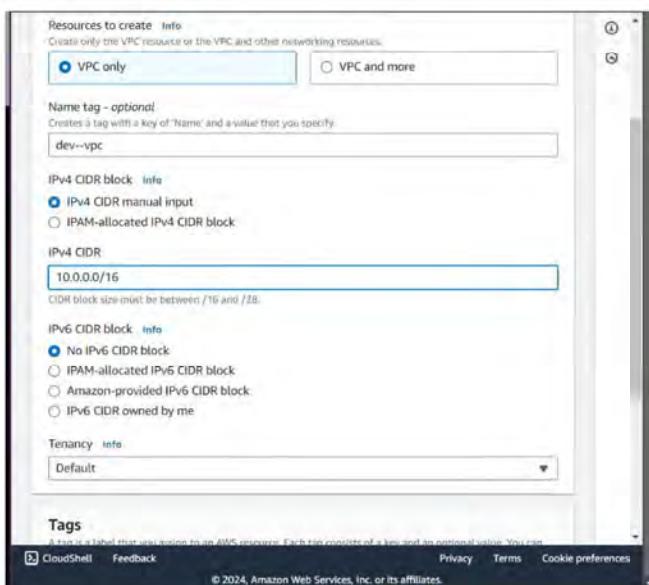
I.

→Create a VPC:

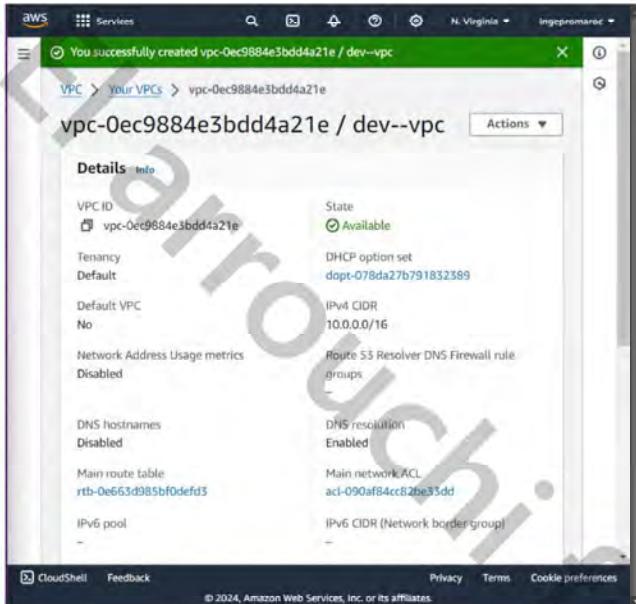
VPC (virtual private cloud) is a virtual network dedicated to your AWS account where you can launch AWS resources.



Type VPC in the search bar and choose VPC from services, after that click on "create VPC"

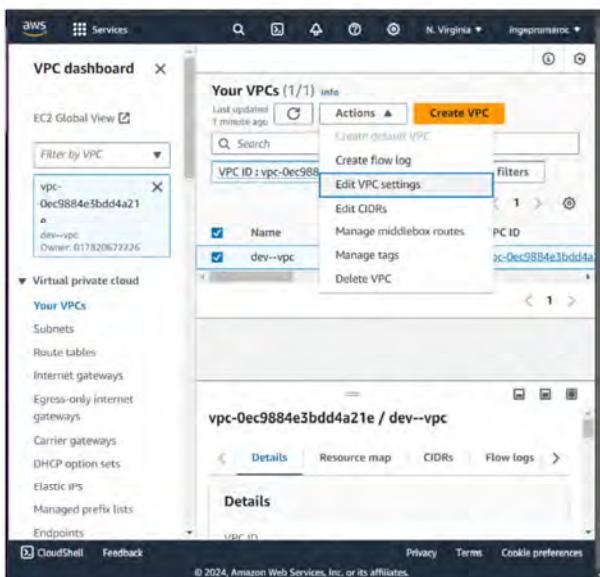


Check VPC only, and give your tag a name after that choose the ipv4 CIDR manual input to since we want to control over the IP address range and then enter the VPC's CIDR block as shown in the architecture (10.0.0.0/16), then choose the No IPV6 CIDR block, leave all other options as default and then click create VPC.

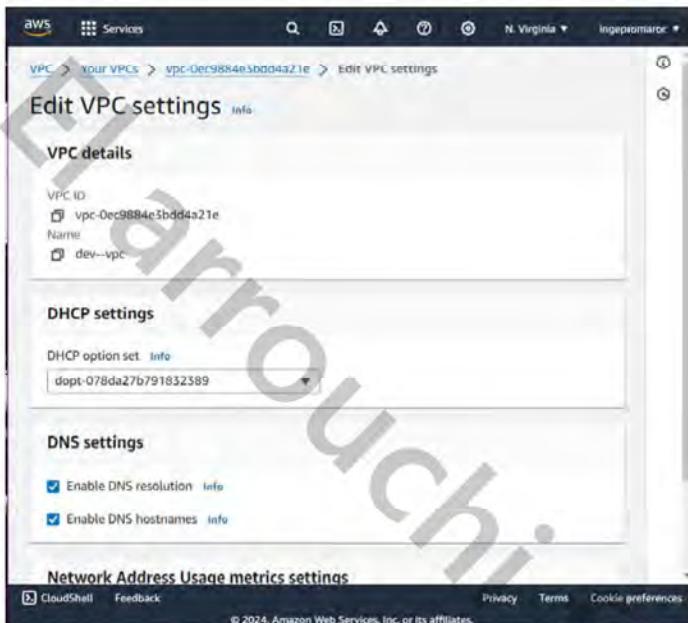


→Enable DNS hostnames for VPC:

In order to resolve domain names to IP addresses for instances within the VPC. This is essential for using features like Route 53 and accessing instances by domain names instead of IP addresses.



select the DEV VPC then select from the actions dropdown “Edit VPC settings”, under DNS settings enable DNS hostnames

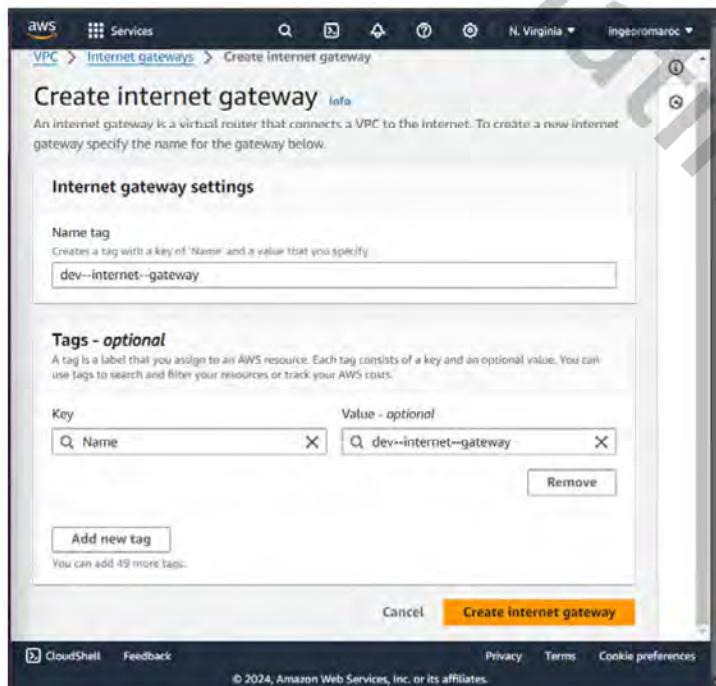
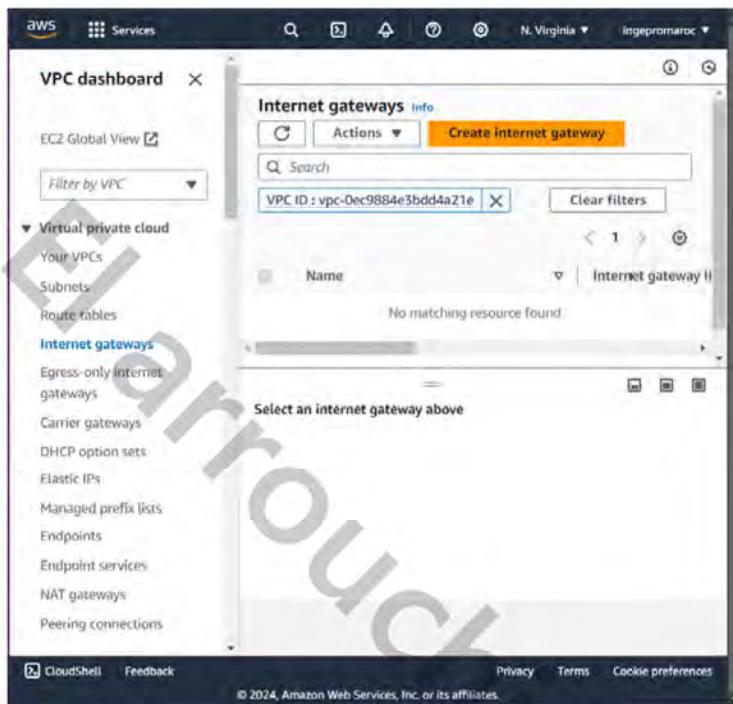


Click on save changes.

→Create an Internet Gateway and attach it to DEV VPC:

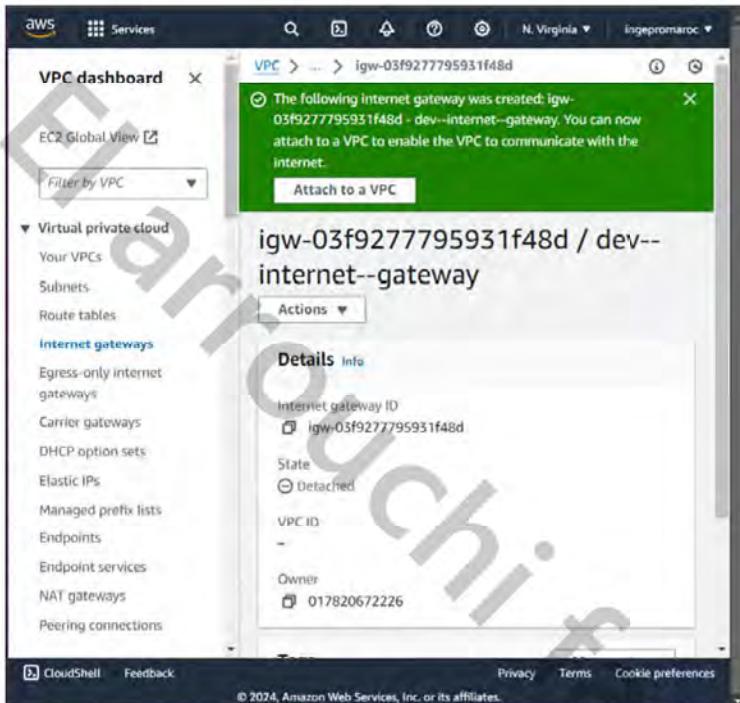
It enables instances within the VPC to communicate with the internet. This allows for outbound (from the instance to the internet) internet access and receiving inbound (from the internet to the instance) traffic, which is crucial for web applications and updates.

In VPC dashboard click on Internet gateways, then click on “create internet gateway”.

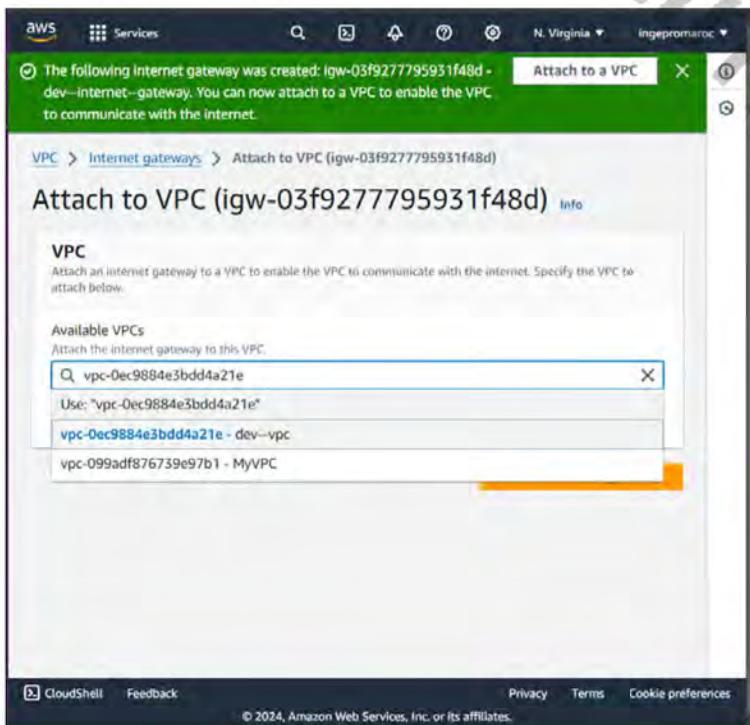


Give your internet gateway a name (dev—internet—gateway in my case), for tags give your tag a name and a description then click on “Create internet gateway”.

We have successfully created our Internet gateway.



Click on "Attach to VPC".

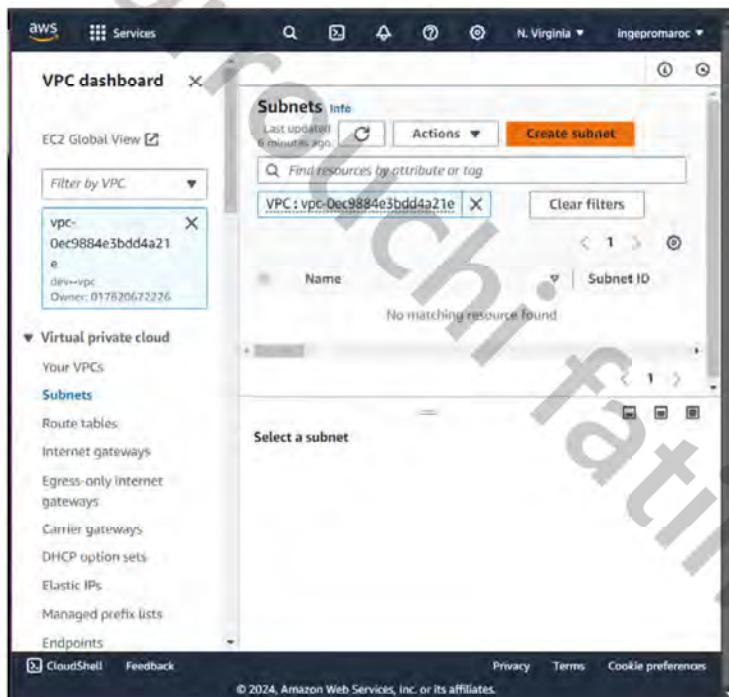


From the available VPCs choose DEV VPC and click on save changes.

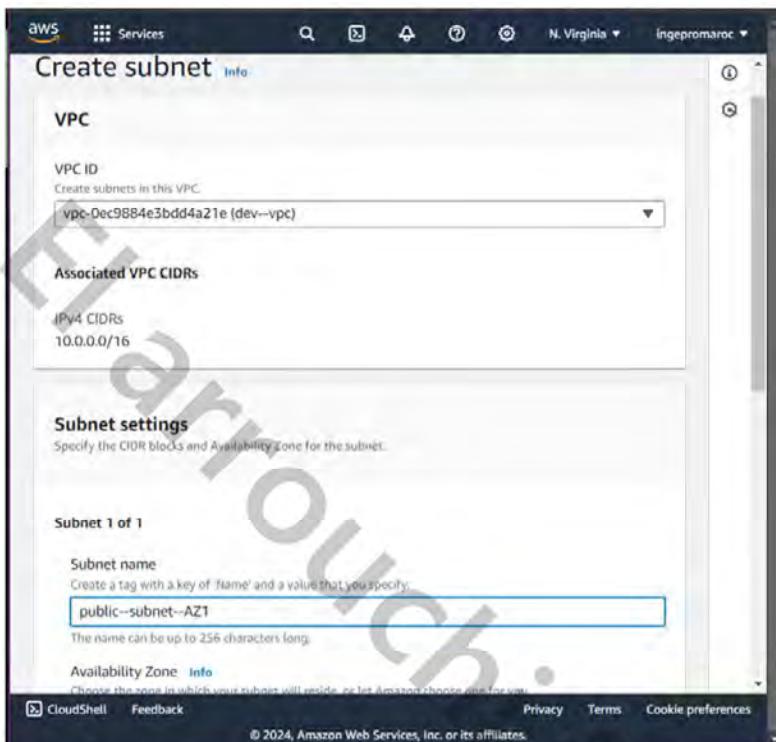
→ Create public subnets:

allow instances within them to have direct access to the internet. This is done by associating the subnets with a route table that routes traffic to an Internet Gateway, enabling both inbound and outbound internet traffic

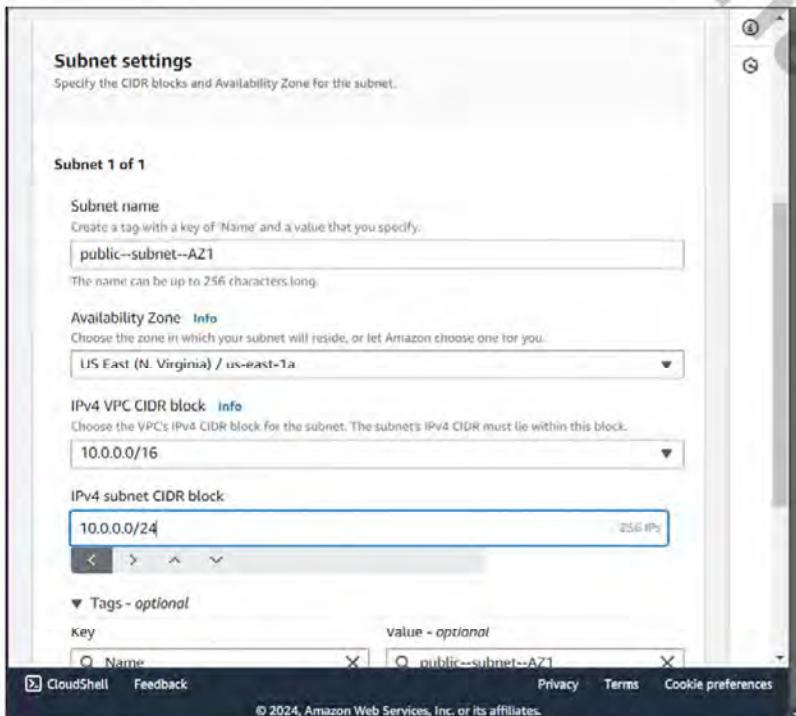
from your VPC dashboard choose subnets and click on “Create subnet”



Select your VPC from the dropdown, give your subnet a name (public--subnet--AZ1 in my case)



For availability zone choose us-east-1a, the IPV4 subnet CIDR block type 10.0.0.0/24 as shown in the reference architecture



In the same way we create the second subnet AZ2.

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs
[Change](#) [Preview](#)

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="public--subnet--AZ2"/>

[CloudShell](#) [Feedback](#) [Privacy](#) [Terms](#) [Cookie preferences](#)

© 2024, Amazon Web Services, Inc. or its affiliates.

We have successfully created the 2 public subnets AZ1 & AZ2.

VPC dashboard

EC2 Global View

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

Peering connections

Subnets (2) [Info](#)

Last updated less than a minute ago

[Actions](#) [Create subnet](#)

[X](#) [Clear filters](#)

Name	Subnet ID
public--subnet--AZ1	subnet-03a274c32b28
public--subnet--AZ2	subnet-026d16a462ae8ddfa

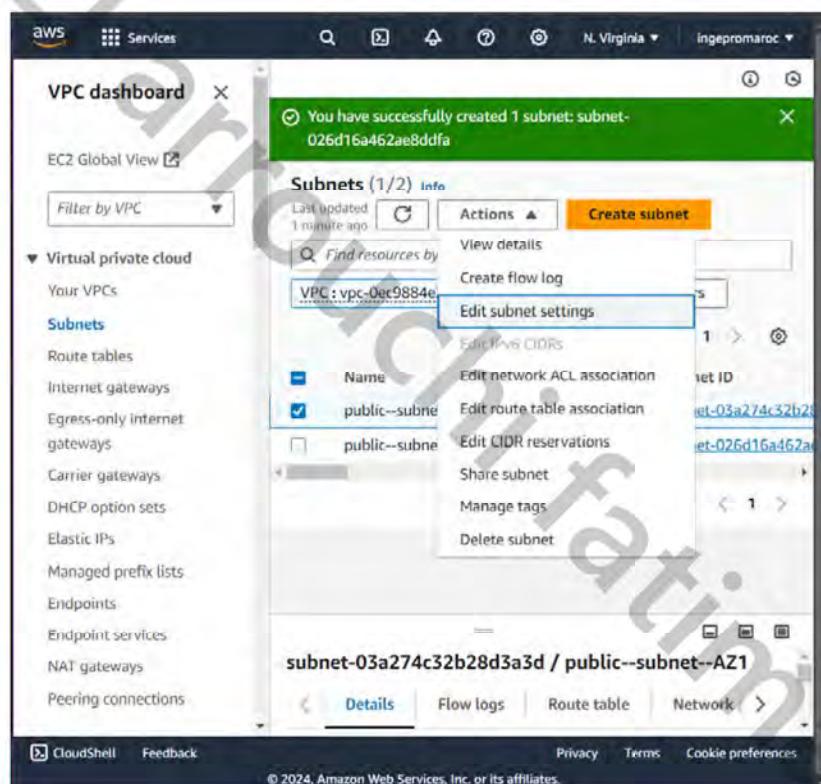
Select a subnet

[CloudShell](#) [Feedback](#) [Privacy](#) [Terms](#) [Cookie preferences](#)

© 2024, Amazon Web Services, Inc. or its affiliates.

Now we have to ensure that instances can directly access the internet without manual IP assignment, by enabling auto assign IP address.

→enable auto assign IP address:



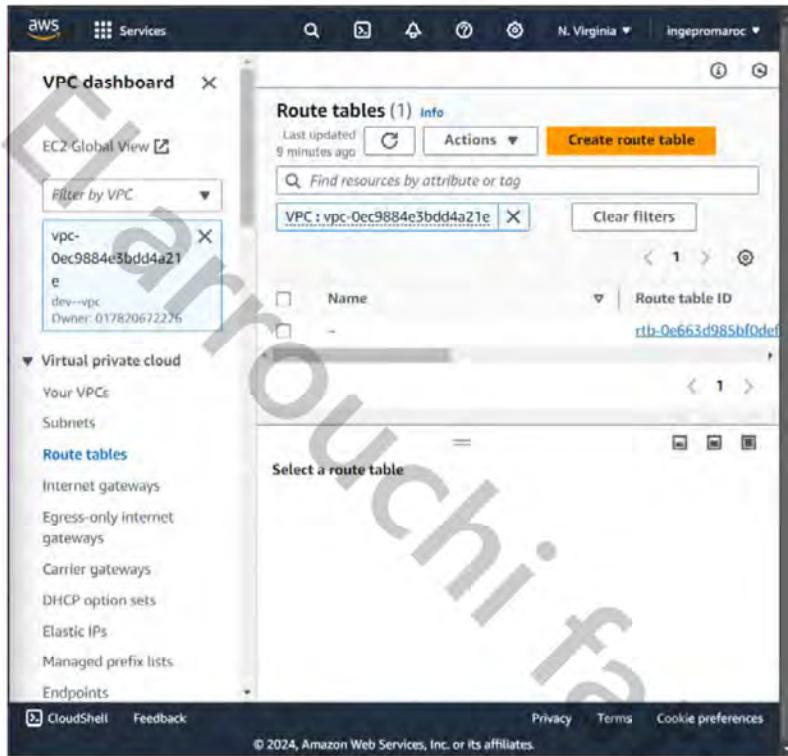
First select the public subnet AZ1, then click on actions and choose edit subnet settings from the dropdown.

Check the auto assign box, and repeat the same steps for the public subnet AZ2.

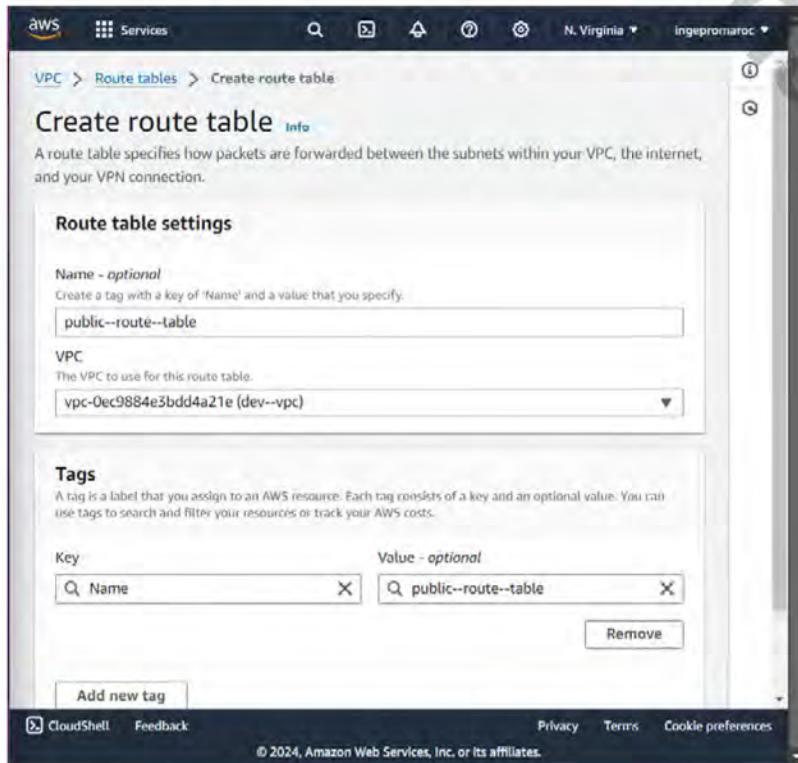
The screenshot shows the 'Edit subnet settings' page for a subnet with ID subnet-03a274c32b28d3a3d. The subnet is named 'public--subnet--AZ1'. Under 'Auto-assign IP settings', the 'Enable auto-assign public IPv4 address' checkbox is checked. The 'Enable auto-assign customer-owned IPv4 address' checkbox is unchecked and has a note: 'Option disabled because no customer owned pools found'. Other sections like 'Resource-based name (RBN) settings' and 'Hostname type' are also visible.

The screenshot shows the 'Edit subnet settings' page for the same subnet. A green success message at the top states: 'You have successfully changed subnet settings: Enable auto-assign public IPv4 address'. The rest of the interface is identical to the first screenshot, showing the subnet details and configuration options.

→ Create a route table:



From the VPC dashboard select route tables, then click on “create a route table” .



Give your route table a name and select your vpc from the VPC dropdown, give the same name to the vpc tags and click on “create”

→ Add public route to the route table:

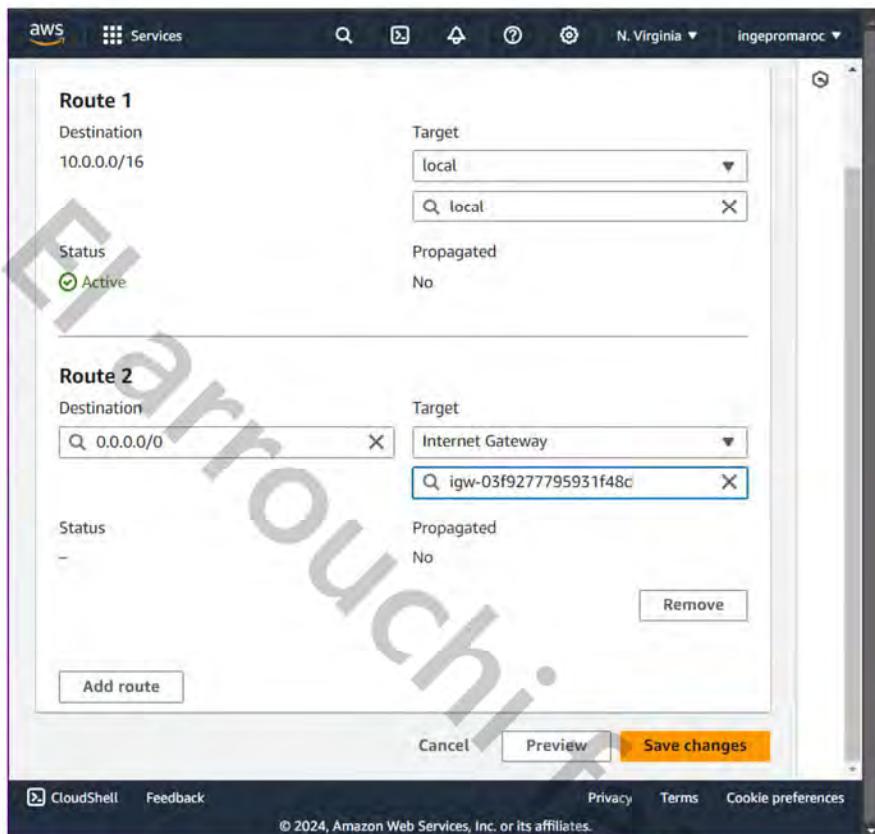
Adding a public route to the route table allows traffic from the subnet to flow to the Internet Gateway, enabling internet access.

The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with options like EC2 Global View, Filter by VPC, Virtual private cloud (Your VPCs, Subnets), Route tables (selected), Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections. The main panel shows a route table named 'vpc-0ec9884e3bdd4a21e | dev--vpc'. It has sections for Main (No), Owner ID (017820672226), Explicit subnet associations (empty), and Edge associations (empty). Below this, there are tabs for Routes, Subnet associations, and Edge associations. The Routes tab is selected, showing a table with one row:

Destina...	Target	Status	Propag...
10.0.0.0/16	local	Active	No

At the bottom of the main panel, there are links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences, along with a copyright notice: © 2024, Amazon Web Services, Inc. or its affiliates.

Here is our route table , first scroll down and under routes select “edit routes”.



Click on add route, in the destination type 0.0.0.0/0 (The internet) and for the target select your Internet Gateway and then save changes

→ associate route table with the public subnets:

Associating a route table with a public subnet directs traffic to the Internet Gateway, enabling internet access for resources in that subnet.

The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with options like EC2 Global View, Filter by VPC, Virtual private cloud, Your VPCs, Subnets, Route tables (which is selected), Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections. The main panel shows a route table named 'rtb-00011af2ef3661167' with the identifier 'vpc-0ec9884e3bdd4a21e | dev--vpc'. It displays 'Main' and 'No' under 'Owner ID' (017820672226). Under 'Explicit subnet associations', there are two entries: '0.0.0.0/0' pointing to 'igw-03f92...' and '10.0.0.0/16' pointing to 'local'. The 'Routes' tab is selected.

Select subnet associations .

The screenshot shows the 'Edit subnet associations' dialog box for route table 'rtb-00011af2ef3661167'. The title is 'Edit subnet associations' and it says 'Change which subnets are associated with this route table.' The 'Available subnets (2/2)' section lists two subnets: 'public--subnet--AZ1' (subnet-03a274c32b28d3a3d, 10.0.0.0/24) and 'public--subnet--AZ2' (subnet-026d16a462ae8ddfa, 10.0.1.0/24). Both are selected with checked checkboxes. In the 'Selected subnets' section, the same two subnets are listed with an 'X' icon next to each. At the bottom are 'Cancel' and 'Save associations' buttons.

After selecting the 2 public subnets AZ1 & AZ2 click “Save associations”.

→ Create private subnets:

In order to isolate resources from the internet, enhancing security by restricting direct external access.

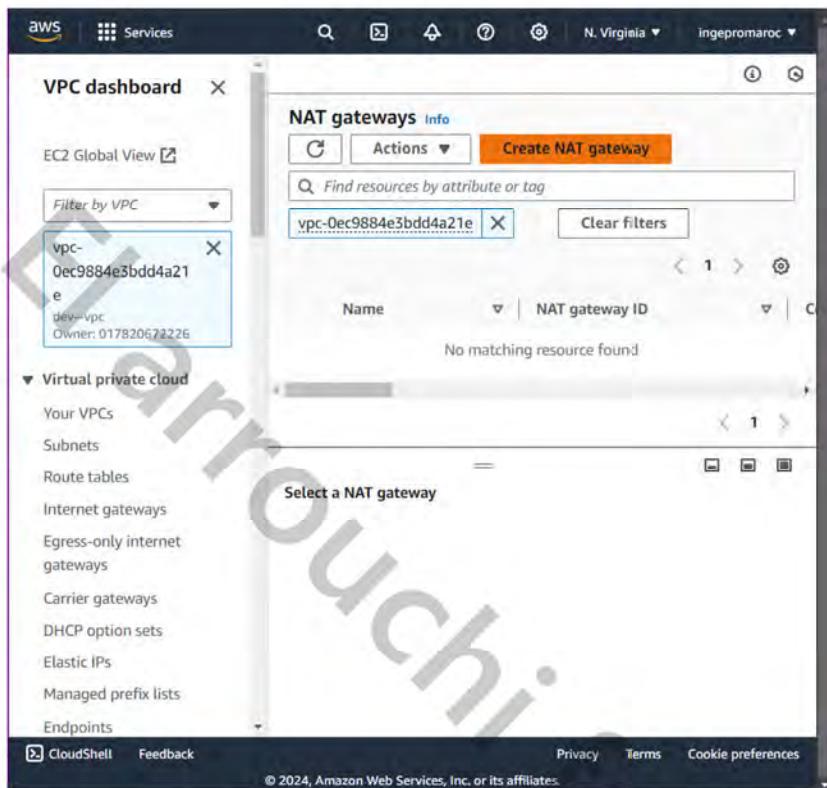
Name	Subnet ID
public--subnet--AZ1	subnet-03a274c321
public--subnet--AZ2	subnet-026d16a46
private--app--subnet--AZ1	subnet-0deb1dfbbf
private--app--subnet--AZ2	subnet-0a775129b
private--data--subnet-AZ1	subnet-0e9385b60
private--data--subnet-AZ2	subnet-0e843876d

Under subnets click on create subnet, and create the private subnets using the informations from the reference architecture following the steps mentioned in ‘Create public subnets’

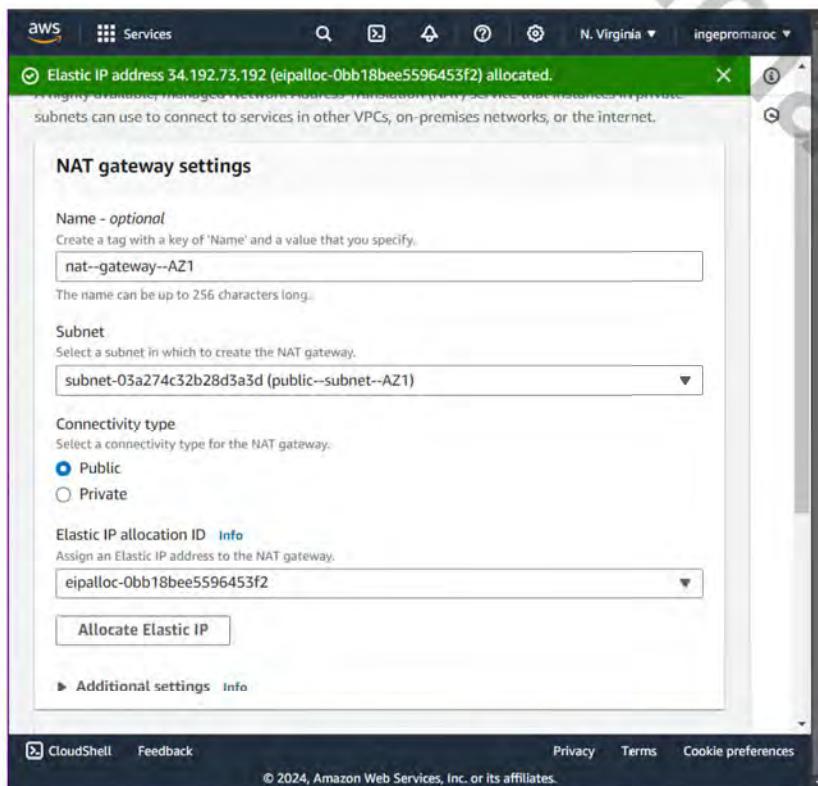
II.

→ Create NAT gateway AZ1:

To allow instances in private subnets to access the internet for updates and patches without exposing them directly to the internet.

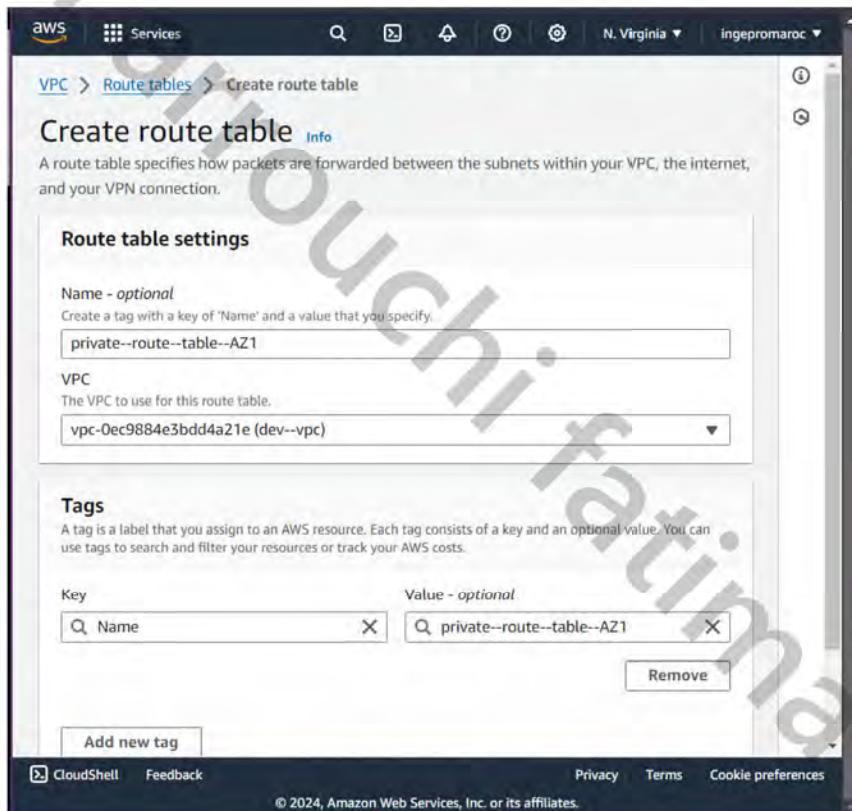


In your VPC dashboard select NAT Gateways, click on “Create NAT gateway”

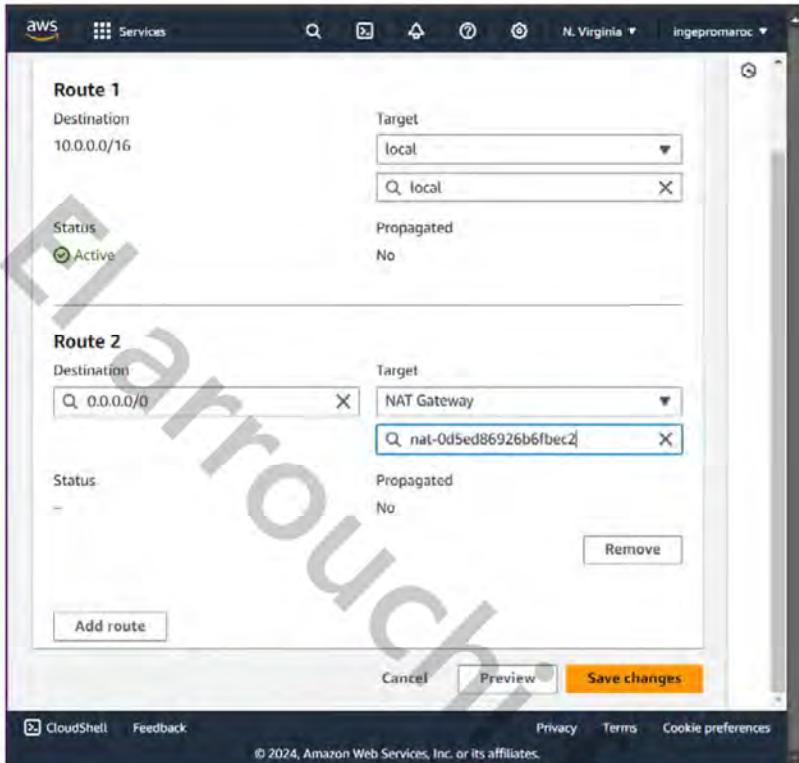


Give your NAT gateway a name nat—gateway—AZ1 in my case, then select your public subnet AZ1 from the dropdown, leave connectivity type as default and click “allocate IP address ” then save changes.

→ Create route table:

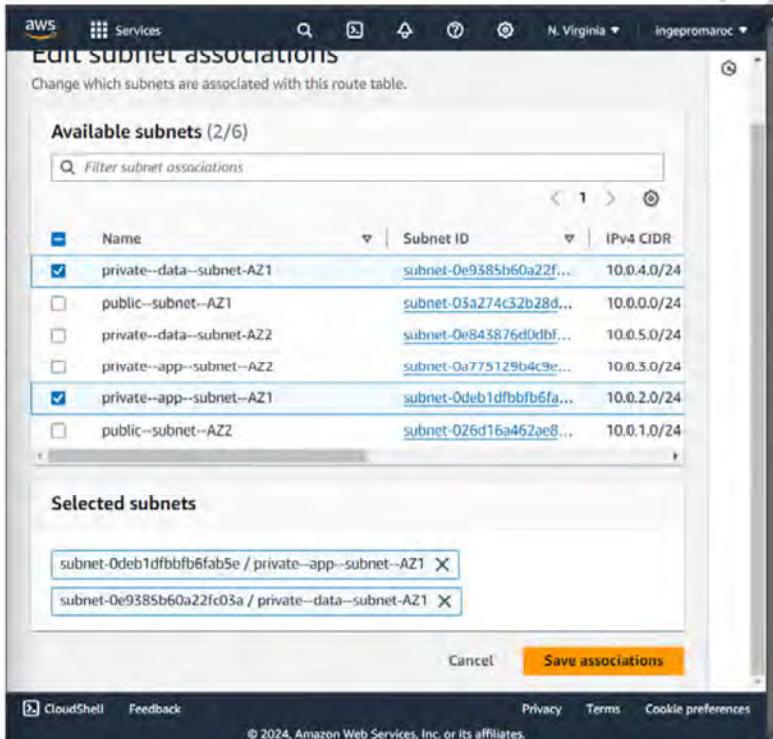


In VPC dashboard choose route tables, click “create route table ”, give your route table a name, choose your VPC and type the same name for the tags value then save the changes.



Under edit routes enter 0.0.0.0/0 in the destination and NAT Gateway in the target, finally save changes.

Go to subnet associations, associate private app AZ1 with the private data AZ1 and save association.

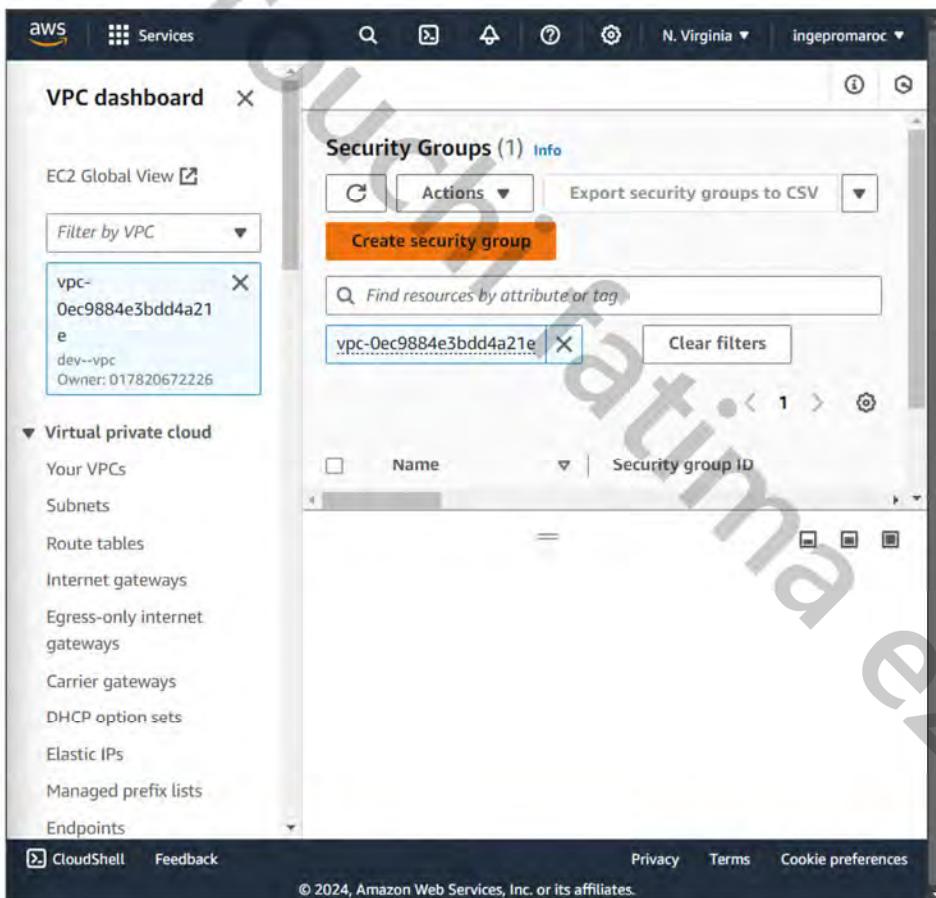


Repeat the same process with the subnet AZ2.

III.

→ Create security group:

In order to control inbound and outbound traffic to your AWS resources, ensuring only authorized access.



In your VPC dashboard select security groups and click on "create security group".

Security group name [Info](#)
alb--security--group
Name cannot be edited after creation.

Description [Info](#)
alb--security--group

VPC [Info](#)
vpc-0ec9884e3bdd4a21e (dev-vpc)

Inbound rules [Info](#)

This security group has no inbound rules.

Give your sec group a name and description, select your VPC.

Under Inbound rules we will set two rules one for HTTP port 80 and the other for HTTPS port 443.

Inbound rule 1

Type [Info](#): HTTP Protocol [Info](#): TCP Port range [Info](#): 80
Source type [Info](#): Anywhere-IPv4 Description - optional [Info](#):
0.0.0.0/0 X

Inbound rule 2

Type [Info](#): HTTPS Protocol [Info](#): TCP Port range [Info](#): 443
Source type [Info](#): Anywhere-IPv4 Description - optional [Info](#):
0.0.0.0/0 X

Add rule

The source will be 0.0.0.0/0.

Follow the same steps and create another security group (SSH—security--group) but in the Inbound rules we set it to SSH port 22.

The screenshot shows the AWS Security Groups Inbound rules configuration. It displays one rule named "Inbound rule 1". The rule details are as follows:

- Type:** SSH
- Protocol:** TCP
- Port range:** 22
- Source type:** My IP
- Source:** 102.96.11.211/32
- Description:** optional

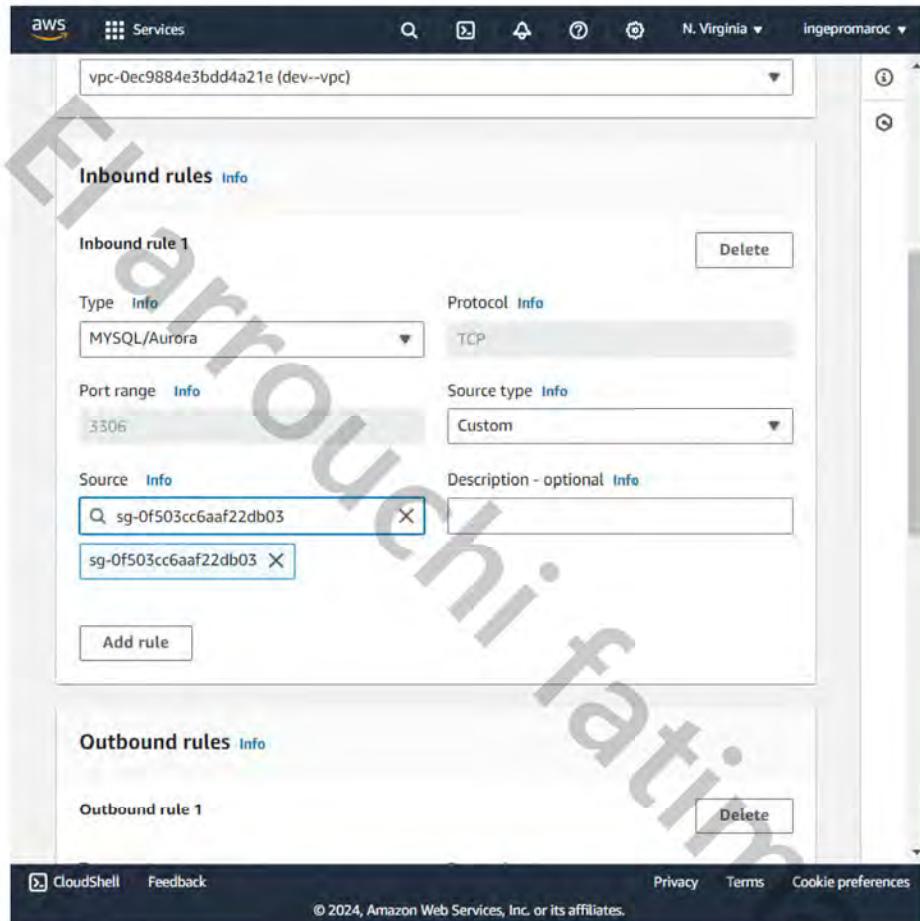
Below this, there is a section for Outbound rules, which currently contains no entries.

Again create another security group (webserver—security--group) and in the Inbound rules we set it to HTTP, HTTPS and SSH.

The screenshot shows the AWS Security Groups Inbound rules configuration. It displays three rules:

- Inbound rule 1:** Type: HTTP, Protocol: TCP, Port range: 80, Source type: Custom, Source: sg-04e54a55f3686cd3f, Description: optional
- Inbound rule 2:** Type: HTTPS, Protocol: TCP, Port range: 443, Source type: Custom, Source: sg-04e54a55f3686cd3f, Description: optional
- Inbound rule 3:** Type: SSH, Protocol: TCP, Port range: 22, Source type: Custom, Source: sg-029fe16ad07925dd6, Description: optional

Finally create the last sec group (database—security--group), in the Inbound rules select MYSQL/AURORA port 3306.



IV.

→ Launch a MYSQL RDS instance:

So as to provide a managed, scalable, and secure database service for your applications.

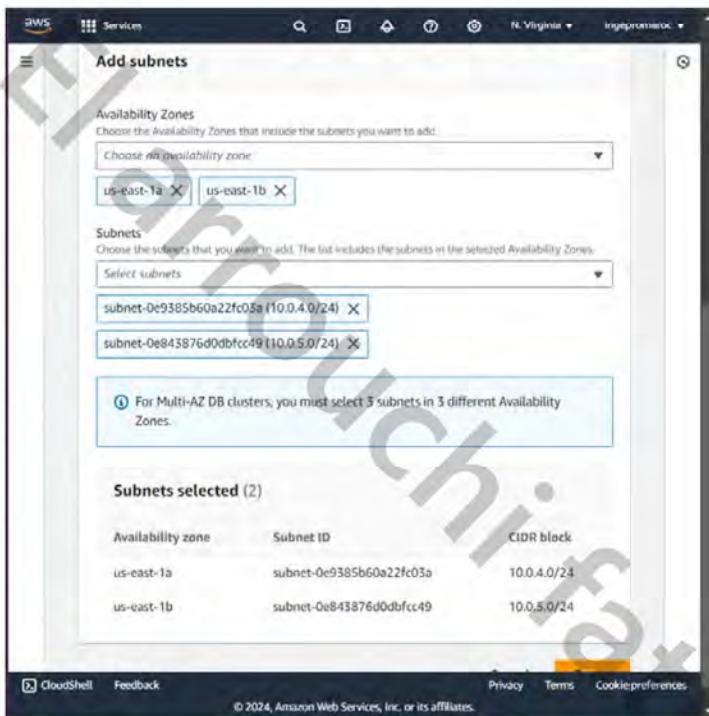
The screenshot shows the AWS RDS Subnet groups page. The left sidebar has links for Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups (selected), Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, and Event subscriptions. The main content area shows 'Subnet groups (2)'. A 'Create DB subnet group' button is at the top. Below it is a search bar and a table with columns: Name, Description, and Status. Two entries are listed: 'database subnets' and 'database-subnets', both with status 'C'.

In the search bar type RDS and select subnet groups from its dashboard, then create a DB subnet group.

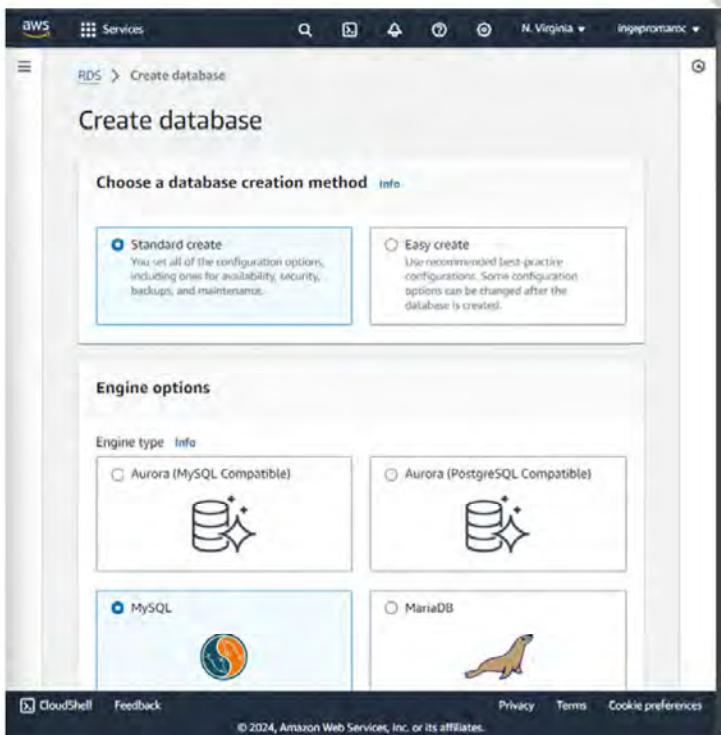
The screenshot shows the 'Create DB subnet group' wizard. Step 1: Subnet group details. It asks for a name ('database--subnets'), a description ('database--subnets'), and a VPC ('dev--vpc (vpc-0ec9884e5bdd4a21e)'). Step 2: Add subnets. It asks to choose Availability Zones. The bottom of the screen shows CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Give it a name and description after that select your VPC.

Select your availability zones and the corresponding subnets as well.



After that go back to the dashboard under databases click on create a database.

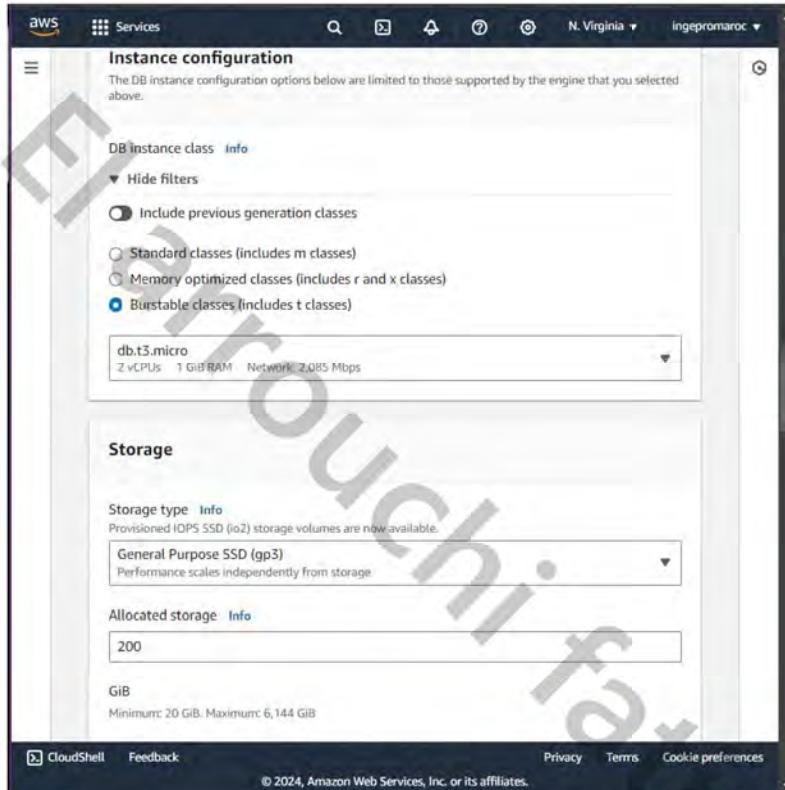


Using a standard creation select MYSQL as an engine option and MYSQL 5.7 for engine version.

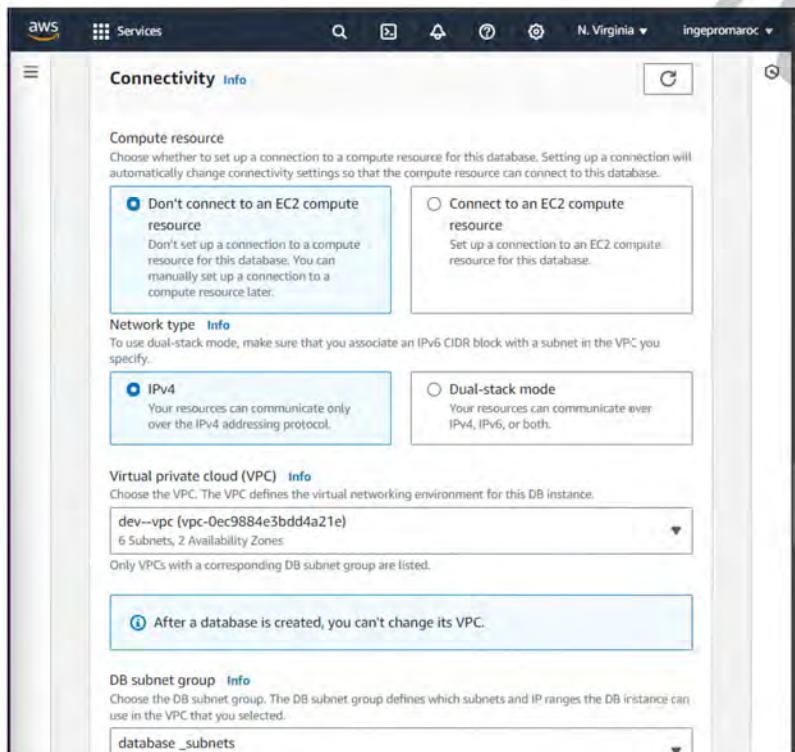
The screenshot shows the AWS RDS engine selection interface. At the top, there are two options: Microsoft SQL Server and IBM Db2. Below them, the MySQL Community edition is selected. Under 'Engine version', 'MySQL 5.7.44' is chosen. A note indicates that MySQL engine versions earlier than 8.0.17 do not support the newest m6g or r6g generation instance classes. At the bottom, there are links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

The screenshot shows the AWS RDS instance configuration page. Under 'Availability and durability', the 'Single DB instance' option is selected. In the 'Settings' section, the 'DB instance identifier' is set to 'dev--rds--db'. In the 'Credentials Settings' section, the 'Master username' is set to 'ingepromaroc'. At the bottom, there are links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Choose a single DB, give your DB instance identifier a name and fill in the credentials settings.



Choose db.t3.micro since it is the free one .



AWS Services N. Virginia ingepromaroc

dev--vpc (vpc-0ec9884e3bdd4a21e)
6 Subnets, 2 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

ⓘ After a database is created, you can't change its VPC.

DB subnet group [Info](#)
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

database_subnets
2 Subnets, 2 Availability Zones

Public access [Info](#)
 Yes
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.
 No
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall) [Info](#)
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

Choose existing
Choose existing VPC security groups

Create new
Create new VPC security group

Existing VPC security groups
Choose one or more options

database--security--group X

Availability Zone [Info](#)
us-east-1b

CloudShell Feedback Privacy Terms Cookie preferences © 2024, Amazon Web Services, Inc. or its affiliates.

AWS Services N. Virginia ingepromaroc

Database authentication options [Info](#)
 Password authentication
Authenticates using database passwords.
 Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.
 Password and Kerberos authentication
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

Monitoring

Enable Enhanced Monitoring
Enabling Enhanced Monitoring metrics are useful when you want to see how different processes or threads use the CPU.

Granularity
60 seconds

Monitoring Role
default

Clicking "Create database" will authorize RDS to create the IAM role rds-monitoring-role

Additional configuration
Database options, encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned off.

Database options

Initial database name [Info](#)
ingepromaroc--db

If you do not specify a database name, Amazon RDS does not create a database.

CloudShell Feedback Privacy Terms Cookie preferences © 2024, Amazon Web Services, Inc. or its affiliates.

V.

→ Create S3 bucket :

Since it provides scalable, secure storage for files, data backups, and hosting static content.

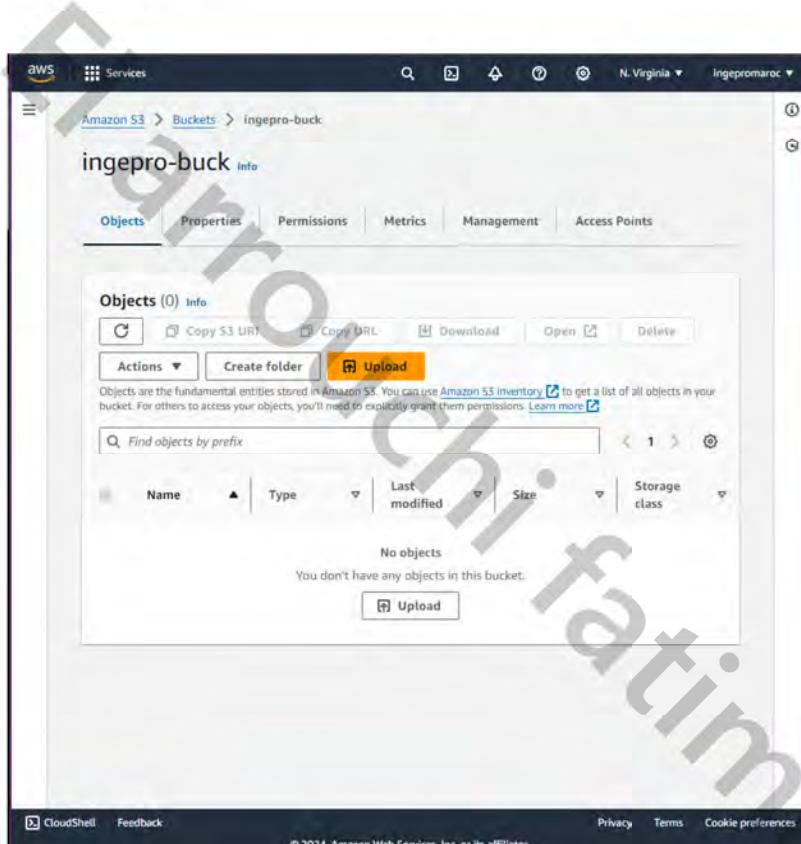
The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with links like 'Buckets', 'Access Grants', 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Bucket Operations', 'IAM Access Analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens' (with 'Dashboards' and 'Storage lens groups'), 'AWS Organizations settings', 'Feature spotlight', and 'AWS Marketplace for S3'. The main area is titled 'Amazon S3' and shows an 'Account snapshot - updated every 24 hours'. It has tabs for 'General purpose buckets' (selected) and 'Directory buckets'. A 'Create bucket' button is prominently displayed. Below it, a table lists existing buckets: 'inge-promaroc-web-files' (Name), 'us-east-1' (AWS region), 'ingepromaroc' (IAM Access Analyzer), and 'August 7, 2024, 16:42:35 (UTC+01)' (Creation date). At the bottom, there are links for 'Privacy', 'Terms', and 'Cookie preferences', along with a copyright notice: '© 2024, Amazon Web Services, Inc. or its affiliates.'

Type S3 in the search box, click on “Create bucket”

This screenshot shows the 'Create bucket' configuration page. The top section is 'General configuration'. It includes fields for 'AWS Region' (set to 'US East (N. Virginia) us-east-1'), 'Bucket type' (radio button selected for 'General purpose'), and 'Bucket name' (input field containing 'inge-pro-buck'). There's also a note about naming rules and a 'Choose bucket' button for copying settings from another bucket. The 'Object Ownership' section follows, with 'Object Ownership info' and two options: 'ACLs disabled (recommended)' (selected) and 'ACLs enabled'. Both options have detailed descriptions below them. At the bottom, there are links for 'Privacy', 'Terms', and 'Cookie preferences', along with a copyright notice: '© 2024, Amazon Web Services, Inc. or its affiliates.'

Give it a name.

→ Upload files:

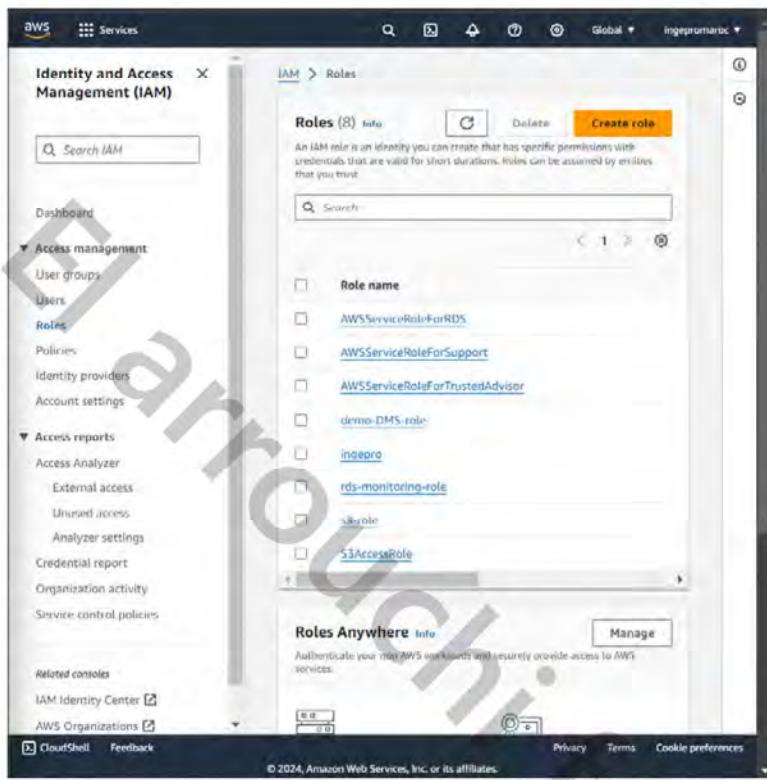


Click on upload and select a zipped version of your website

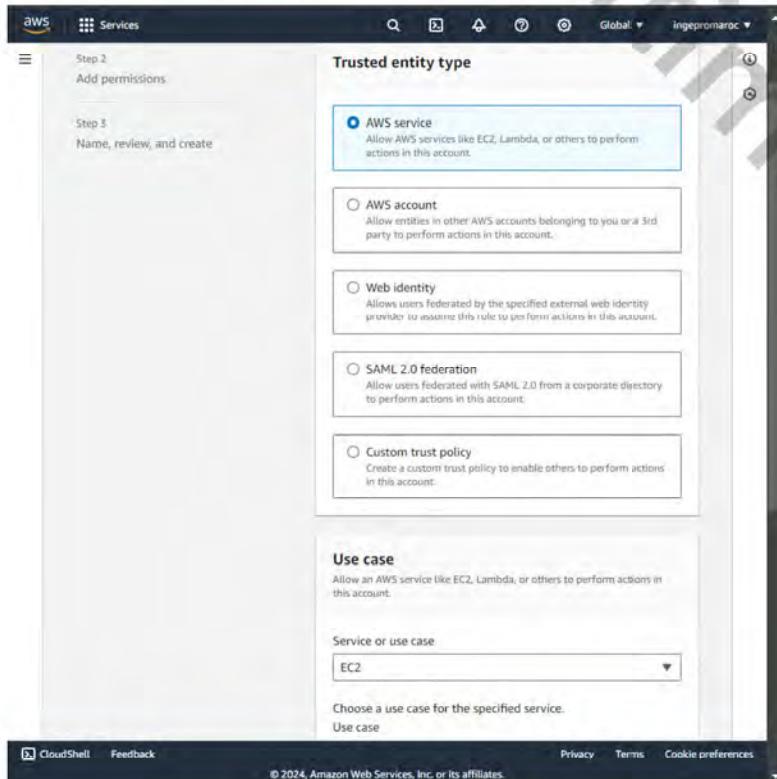
VI.

→ Create an IAM role with S3 policy:

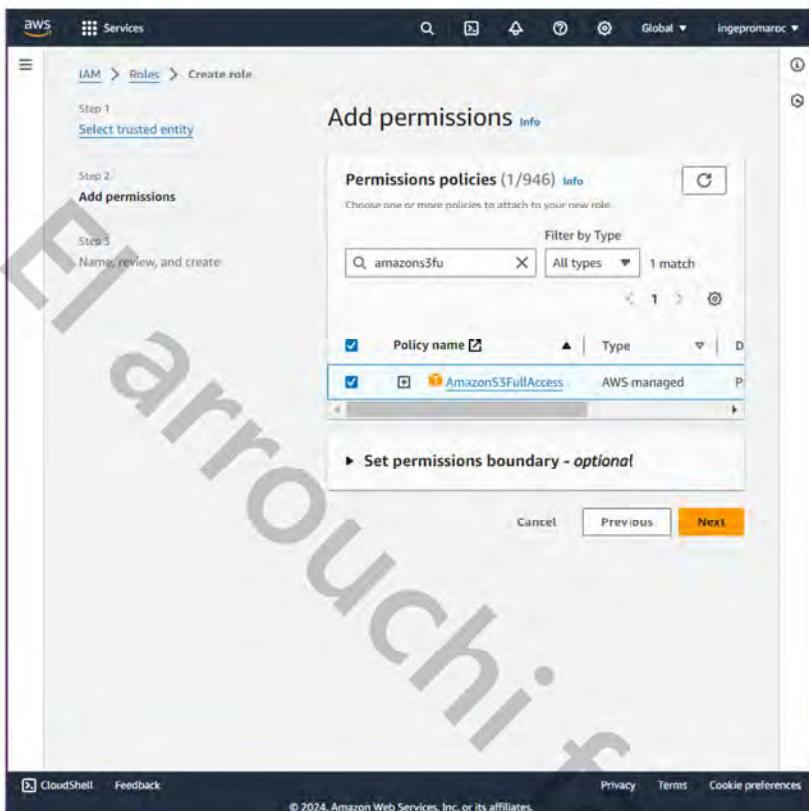
To grant controlled access to S3 buckets, allowing services or applications to securely interact with S3 resources.



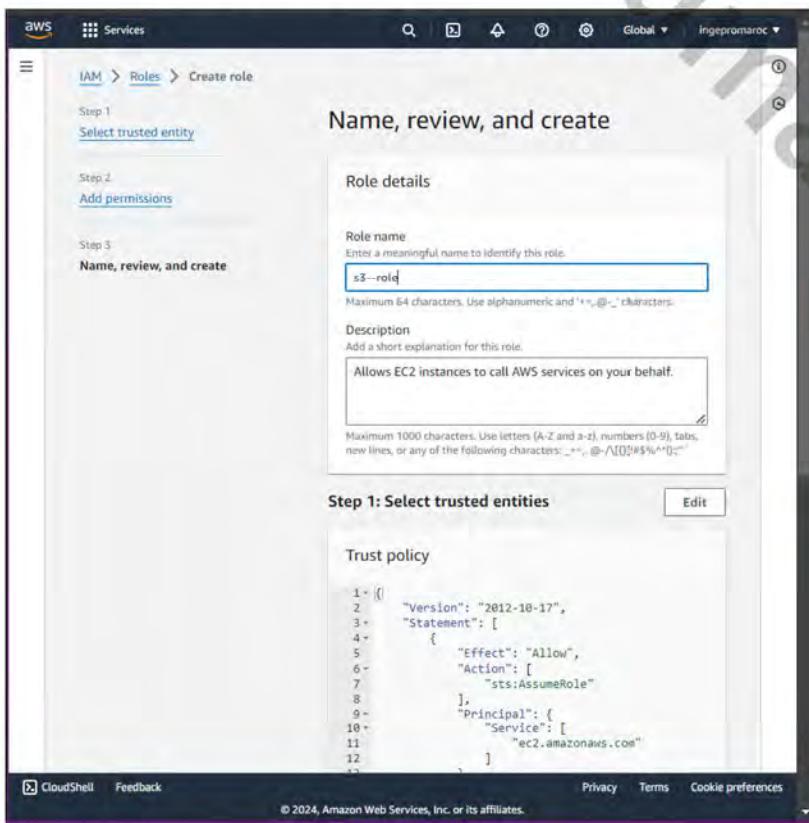
In your search box type IAM, select Roles from the dashboard then create a role



In the trusted entities select AWS service, for the use case select EC2.



In the permissions policies search box type AmazonS3FullAccess and select it.

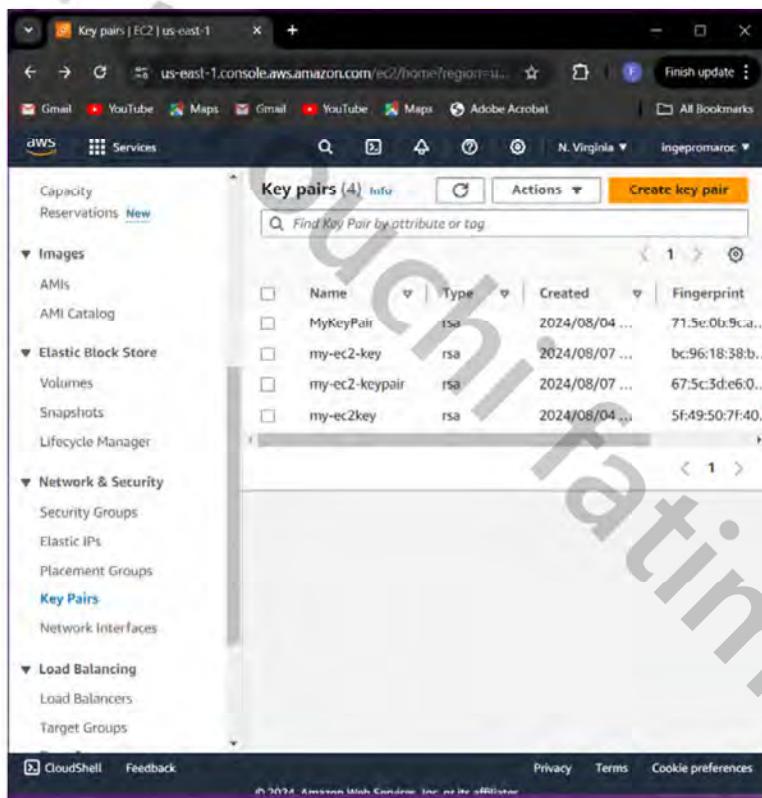


Give your role a name and description then save changes.

VII.

→ Create a keypair :

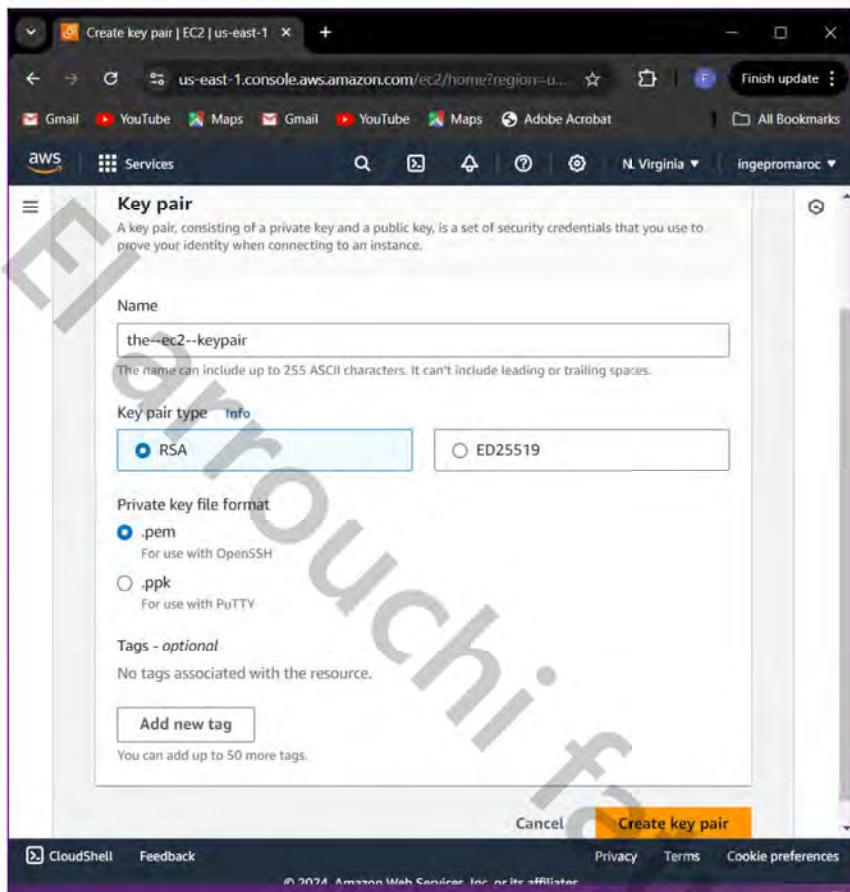
To allow secure SSH access to your EC2 instances, ensuring only authorized users can connect.



The screenshot shows the AWS EC2 Key Pairs page. On the left, there's a sidebar with various services like Capacity Reservations, Images, AMIs, AMI Catalog, and more. The 'Key Pairs' section is highlighted. The main area displays a table of key pairs:

Name	Type	Created	Fingerprint
MyKeyPair	rsa	2024/08/04 ...	71:5c:0b:9c:a...
my-ec2-key	rsa	2024/09/07 ...	bc:96:18:38:b...
my-ec2-keypair	rsa	2024/08/07 ...	67:5c:3d:e6:0...
my-ec2key	rsa	2024/08/04 ...	5f:49:50:7f:40...

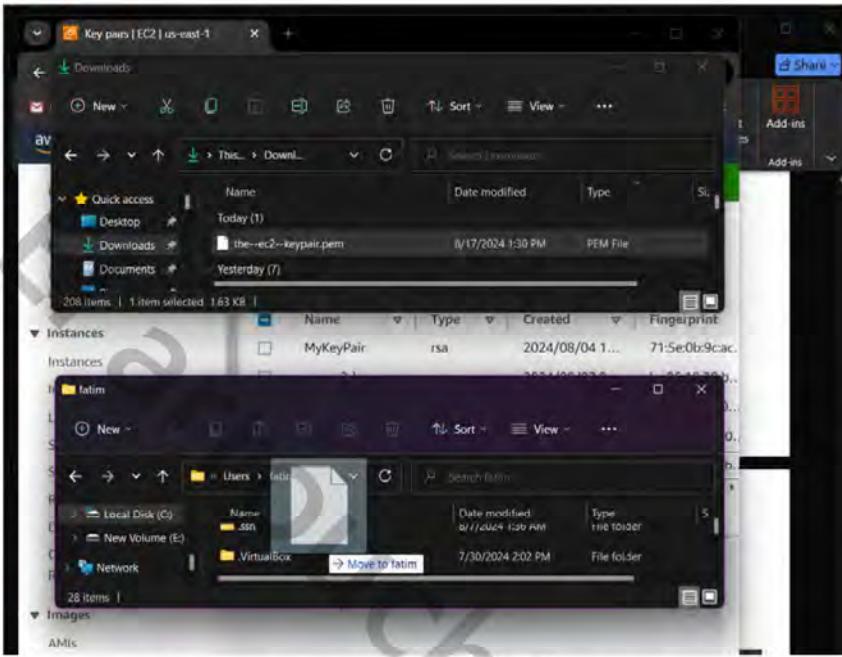
In your ec2 dashboard, select key pairs and create one.



Give it a name, and choose RSA as a type because it provides a robust and widely-supported encryption standard, and .pem as format since it ensures easy compatibility with OpenSSH and other SSH clients commonly used for managing EC2 instances.

After saving changes a private keypair will be downloaded to your downloads directory, you have to move that private key to the home directory that powershell opens to in order to ensure easy access and correct permissions, since we will have to connect (SSH connection) to our EC2 instance using powershell.

A screenshot of a Windows PowerShell window. The title bar says 'Windows PowerShell'. The content shows the standard PowerShell welcome message: 'Windows PowerShell', 'Copyright (C) Microsoft Corporation. All rights reserved.', 'Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows', and the command prompt 'PS C:\Users\fatim>'. The background is dark blue.



Open your powershell and see the present working directory copy its path, then go to your file explorer open the copied path, cut the private key and paste it there.

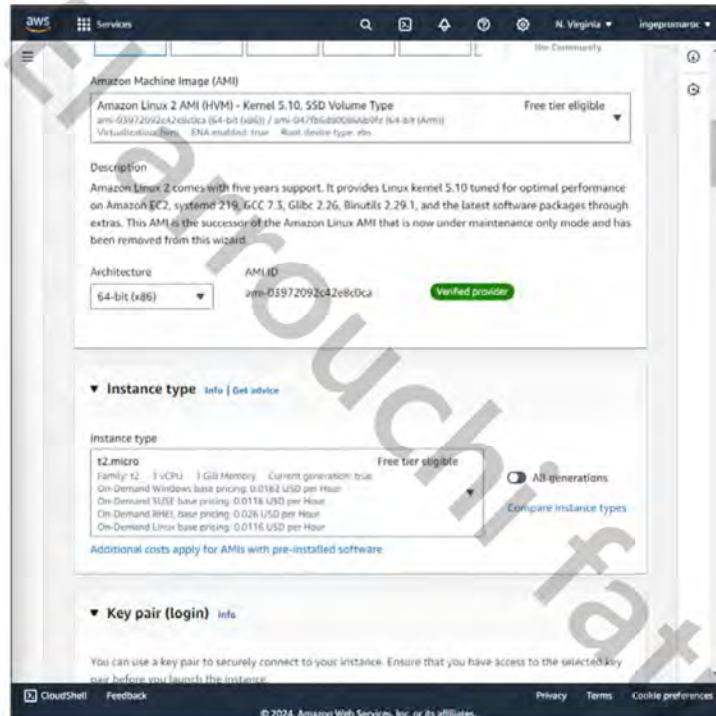
VIII.

→ Launch a setup EC2 instance:

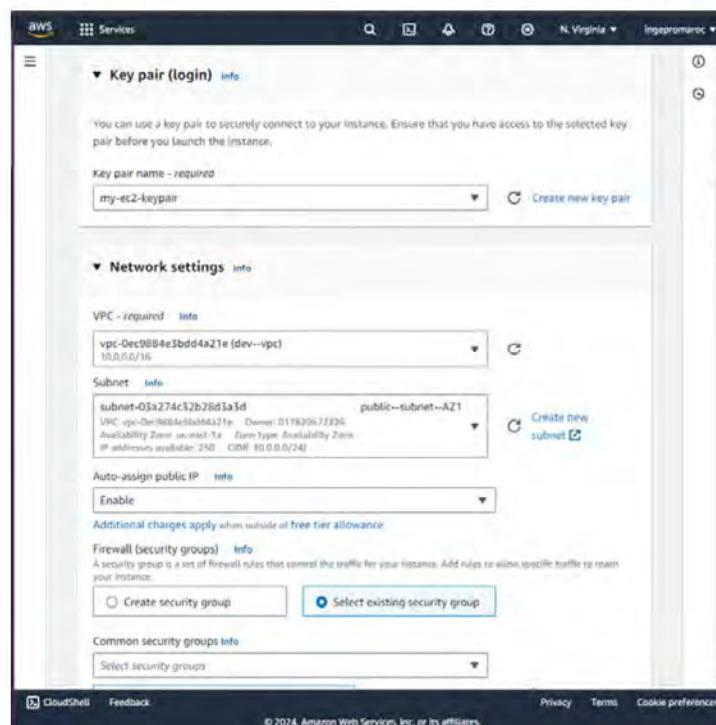
A screenshot of the AWS CloudFormation console. The top navigation bar shows 'Services', 'N. Virginia', and a user profile. The main page title is 'Launch an instance'. Below it, a sub-section title 'Launch an instance' is shown with a sub-instruction: 'Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.' There are two main sections: 'Name and tags' and 'Application and OS Images (Amazon Machine Image)'. In the 'Name and tags' section, the 'Name' field is filled with 'setup-server'. In the 'Application and OS Images (Amazon Machine Image)' section, there's a search bar and a list of recent and quick-start AMIs: Amazon Linux, macOS, Ubuntu, Windows, and Red Hat. A 'Browse more AMIs' link is also present.

Go to instances in your EC2 dashboard and click “launch instance”.

Give your instance a name, under AMI choose Amazon linux because it's optimized for AWS, offering better performance, security, and seamless integration with AWS services.



T2 micro as a type since it's the lowest-cost option in AWS.



Choose the already created key pair and VPC, select public subnet AZ1, choose SSH, ALB and webserver security groups, select IAM role then save changes.

IX.

→ Use MYSQL workbench to import data into an RDS db:

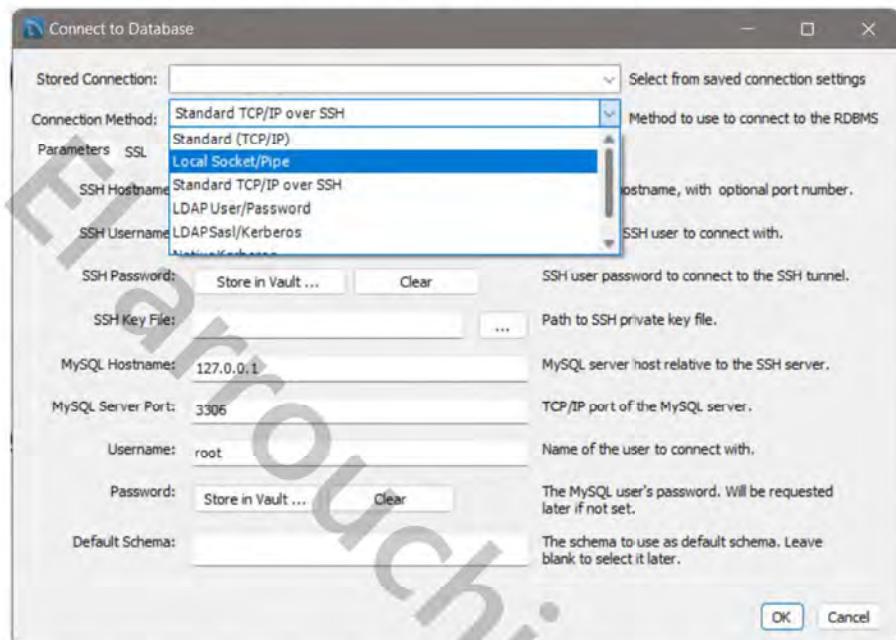
is beneficial because it provides a user-friendly interface to manage and transfer data.

The screenshot shows the AWS EC2 Instances page. At the top, there are buttons for 'Connect', 'Instance state', and 'Actions'. A search bar says 'Find instance by attribute or tag (case-sensitive)' and dropdowns show 'Stopped' and '1'. Below is a table with columns 'Name', 'Instance ID', 'Instance state', and 'Instance type'. One row is selected: 'setup server' (Instance ID i-0edf37909417688de), which is 'Running' on an 't2.micro' instance. At the bottom, there's a details panel for the selected instance, showing its summary with fields like Instance ID, Public IPv4 address (3.89.202.39), Private IPv4 addresses (10.0.0.13), and Instance state (Running).

In your EC2 dashboard select instances, copy the public IPV4 DNS.

The screenshot shows the MySQL Workbench application window. The title bar says 'MySQL Workbench'. The main menu includes File, Edit, View, Database, Tools, Scripting, Help, and a 'Connect to Database' dropdown. Below the menu is a toolbar with icons for connecting, disconnecting, and other database operations. The central area has tabs for 'MySQL Connections' and 'Script Editor'. A message at the bottom states: 'MySQL Workbench could not detect any MySQL server running. This means that MySQL is not installed or is not running.' There are links to 'Browse Documentation', 'Read the Blog', and 'Discuss on the Forum'.

Open MySQL workbench, go to database and select connect to database.



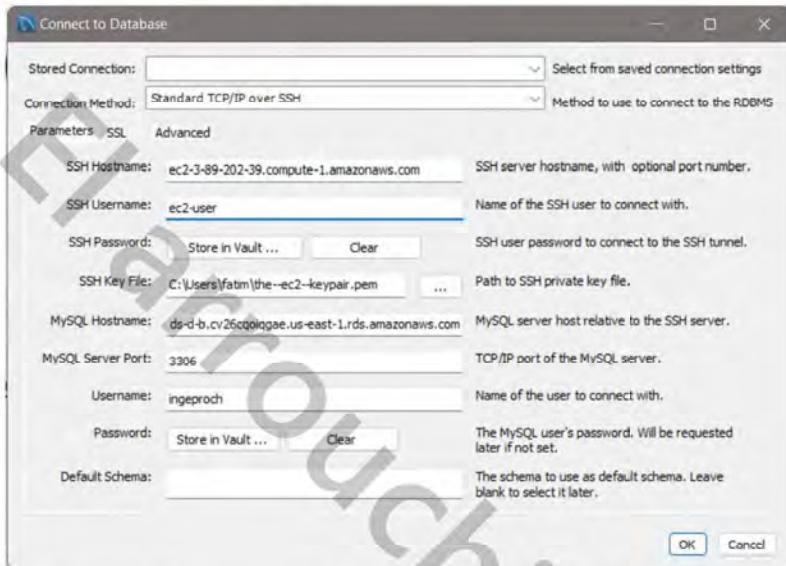
Under connection method choose standard TCP/IP over SSH.

The screenshot shows the AWS RDS console. The URL is 'us-east-1.console.aws.amazon.com/rds/home?region...'. The navigation path is 'RDS > Databases > dev-rds-d-b'. The main view is titled 'dev-rds-d-b' with 'Actions' buttons for 'Edit', 'Modify', and 'Actions'. Below is a 'Summary' table:

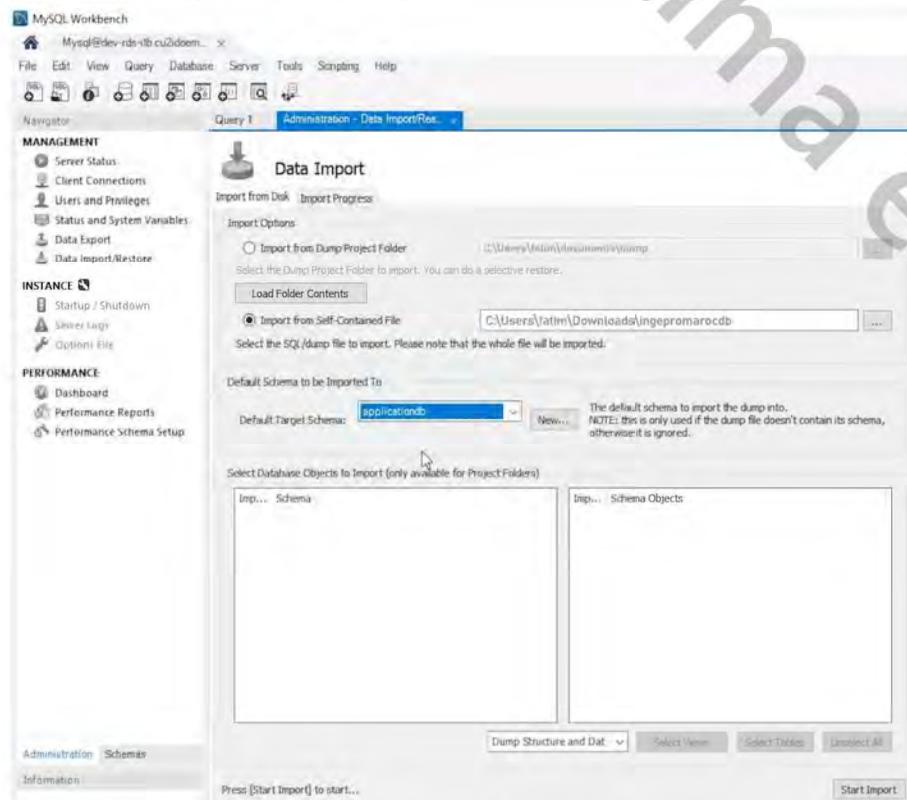
DB identifier	Status	Role	Engine	Recommendations
dev-rds-d-b	Available	Instance	MySQL	Community
CPU	Class	Current activity	Community	Region & AZ
2.95%	db.t3.micro	0	MySQL 8.0	1 High and 1 other
		Connections	us-east-1b	

Below the summary is a 'Connectivity & security' tab, which is currently active. At the bottom, there are links for 'CloudShell', 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences'.

Go to your RDS db, under connectivity & security copy the endpoint, and go to configuration and copy the name of db master.



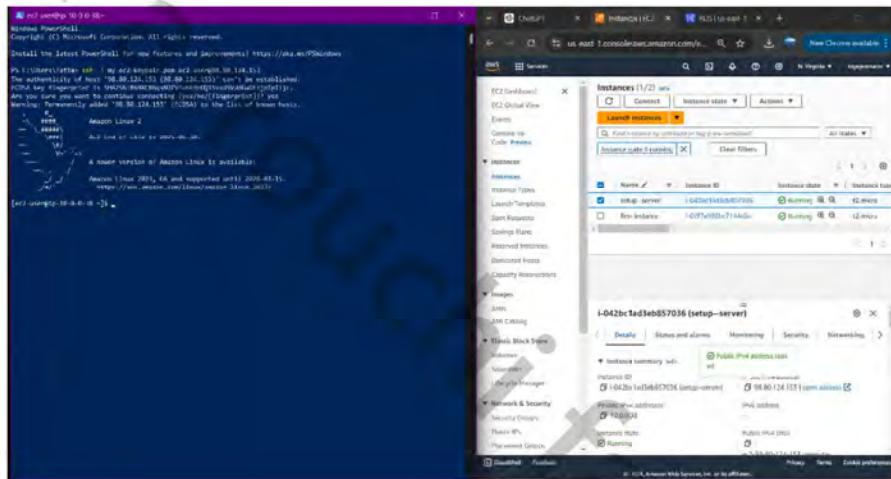
In the SSH hostname paste public IPV4 DNS, the SSH username is always ec2-user, in the SSH key file choose the key pair placed in the directory that powershell opens to, for the MYSQL hostname paste the endpoint for your RDS, leave the port as it is, and for the username you need to paste the db master from your RDS db and finally enter the db password then hit ok.



After setting a connection with your database, under import from self-contained file select your database file then start the import.

X.

→ Install our dynamic website on an EC2 instance:



First open your instance and copy the public IPV4 address, after that open your powershell and type this command to SSH into your EC2 instance.

```
ssh -i your_keypair_name ec2-user@public_IPV4_address
```

after that paste theses commands in your powershell:

#1. Update EC2 instance

```
sudo yum update -y
```

#2. Install Apache

```
sudo yum install -y httpd
```

```
sudo systemctl enable httpd
```

```
sudo systemctl start httpd
```

#3. Install PHP 8.2

```
sudo amazon-linux-extras enable php8.2  
sudo yum clean metadata  
sudo yum install php php-common php-pear -y  
sudo yum install php-{cgi,curl(mbstring),gd,mysqlnd,gettext,json,xml,fpm,intl,zip} -y
```

#4. Install MySQL 8.0

```
sudo rpm -Uvh https://dev.mysql.com/get/mysql80-community-release-el7-5.noarch.rpm  
sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022  
sudo yum install mysql-community-server -y  
sudo systemctl enable mysqld  
sudo systemctl start mysqld
```

#5. Download the ingepromaroc zip from S3 to the html directory on the EC2 instance

```
sudo aws s3 sync s3://ingepro-buck /var/www/html
```

#6. Unzip the ingepromaroc zip folder

```
cd /var/www/html  
sudo unzip ingepromaroc.zip
```

#7. Move all the files and folders from the ingepromaroc directory to the html directory

```
sudo mv ingepromaroc/* /var/www/html
```

#8. Move all the hidden files from the ingepromaroc directory to the html directory

```
sudo mv ingepromaroc/.well-known /var/www/html  
sudo mv ingepromaroc/.env /var/www/html  
sudo mv ingepromaroc/.htaccess /var/www/html
```

#9. Delete the ingepromaroc and ingepromaroc.zip folder

```
sudo rm -rf ingepromaroc ingepromaroc.zip
```

#10. Enable mod_rewrite on EC2 Linux

```
sudo sed -i '/<Directory "/var/www/html"/,</Directory>/ s/AllowOverride None/AllowOverride All/' /etc/httpd/conf/httpd.conf
```

#11. Set permissions

```
sudo chmod -R 777 /var/www/html
```

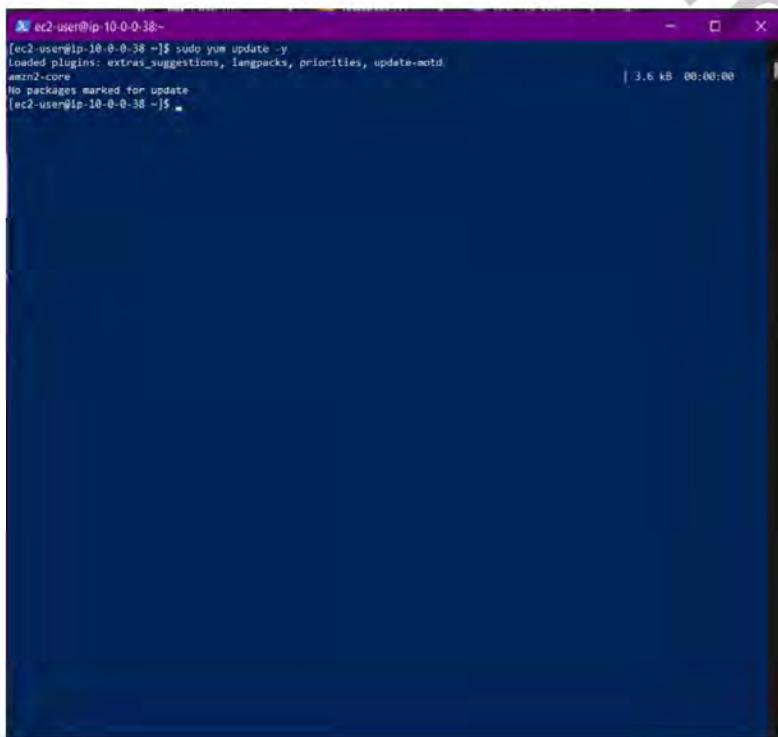
```
sudo chmod -R 777 storage/
```

#12. Add database credentials

```
sudo vi .env
```

#13. Restart server

```
sudo service httpd restart
```



```
ec2-user@ip-10-0-0-38:~$ sudo yum install -y httpd httpd-tools mod_ssl sudo systemctl enable httpd sudo systemctl start httpd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package sudo-1.8.23-10.amzn2.3.7.x86_64 already installed and latest version
No package systemctl available.
No package enable available.
Package httpd-1.8.23-10.amzn2.3.7.x86_64 already installed and latest version
No package systemctl available.
No package start available.
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.61-1.amzn2.0.1 will be installed
--> Processing Dependency: httpd-filesystem = 2.4.61-1.amzn2.0.1 for package: httpd-2.4.61-1.amzn2.0.1.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.61-1.amzn2.0.1.x86_64
--> Processing Dependency: mod-filesystem for package: httpd-2.4.61-1.amzn2.0.1.x86_64
--> Processing Dependency: system-logos-httd for package: httpd-2.4.61-1.amzn2.0.1.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.61-1.amzn2.0.1.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.61-1.amzn2.0.1.x86_64
--> Package httpd-tools.x86_64 0:2.4.61-1.amzn2.0.1 will be installed
--> Package mod_ssl.x86_64 1:2.4.61-1.amzn2.0.1 will be installed
--> Processing Dependency: sscc >= 2.2.0 for package: 1:mod_ssl-2.4.61-1.amzn2.0.1.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.7.2-2.amzn2 will be installed
--> Package apr-util.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Processing Dependency: apr-util-libs(x86-64) = 1.6.3-1.amzn2.0.1 for package: apr-util-1.6.3-1.amzn2.0.1.x86_64
--> Package generic-logos-httd.noarch 0:18.0.0-4.amzn2 will be installed
--> Package httpd-filesystem.noarch 0:2.4.61-1.amzn2.0.1 will be installed
--> Package mod_mllcap.noarch 0:2.1.41-2.amzn2 will be installed
--> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.2 will be installed
--> Package sscc.x86_64 0:2.3.3-2.amzn2.0.1 will be installed
--> Processing Dependency: libtalloc.so.2(TALLOC_2.0.2)(64bit) for package: sscc-2.3.3-2.amzn2.0.1.x86_64
--> Processing Dependency: libtalloc.so.2()(64bit) for package: sscc-2.3.3-2.amzn2.0.1.x86_64
--> Running transaction check
--> Package apr-util-bdb.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Package libtalloc.x86_64 0:2.1.16-1.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

-----  
Package          Arch      Version       Repository      Size  
-----  
Installing:  
httpd           x86_64   2.4.61-1.amzn2.0.1   amzn2-core      1.4 M  
httpd-tools     x86_64   2.4.61-1.amzn2.0.1   amzn2-core      89 k  
mod_ssl         x86_64   1:2.4.61-1.amzn2.0.1   amzn2-core      118 k  
Installing for dependencies:  
apr              x86_64   1.7.2-2.amzn2          amzn2-core      130 k  
apr-util        x86_64   1.6.3-1.amzn2.0.1   amzn2-core      101 k
```

```
ec2-user@ip-10-0-0-38:~$ sudo amazon-linux-extras enable php8.1
[ec2-user@ip-10-0-0-38 ~]$ sudo yum clean metadata
Topic php8.1 has end-of-support date of 2024-11-25
  2 httpd_modules           available [ ~1.0 ~stable ]
  3 memcached1.5            available \
    [ ~1.5.1 ~1.5.16 ~1.5.17 ]
  9 R3.4                    available [ ~3.4.3 ~stable ]
10 rust1                  available \
    [ ~1.22.1 ~1.26.0 ~1.26.1 ~1.27.2 ~1.31.0 ~1.38.0
      ~stable ]
18 libreoffice             available \
    [ ~5.0.6.2_15 ~5.3.6.1 ~stable ]
19 gimp                   available [ ~2.8.22 ]
20 *dockershim-extras      enabled \
    [ ~17.12.1 ~18.03.1 ~18.06.1 ~18.09.9 ~stable ]
21 mate-desktop1.x         available \
    [ ~1.19.0 ~1.20.0 ~stable ]
22 GraphicsMagick1.3       available \
    [ ~1.3.29 ~1.3.32 ~1.3.34 ~stable ]
24 epel                   available [ ~7.11 ~stable ]
25 testing                 available [ ~1.0 ~stable ]
26 ecs                    available [ ~stable ]
27 *corretto8              available \
    [ ~1.8.0_192 ~1.8.0_202 ~1.8.0_212 ~1.8.0_222 ~1.8.0_232
      ~1.8.0_242 ~stable ]
32 lustre2.10              available \
    [ ~2.10.5 ~2.10.8 ~stable ]
33 *java-openjdk11          available [ ~11 ~stable ]
34 lynis                  available [ ~stable ]
36 RCC                    available [ ~0.x ~rcahla ]
37 mono                   available [ ~5.x ~stable ]
38 nginx1                available [ ~stable ]
40 mock                   available [ ~stable ]
43 livepatch              available [ ~stable ]
44 *python3.8               available [ ~stable ]
45 haproxy2                available [ ~stable ]
46 collectd                available [ ~stable ]
47 aws-nitro-enclaves-cli available [ ~stable ]
48 R4                     available [ ~stable ]
kernel-5.4                available [ ~stable ]
50 selinux-ng              available [ ~stable ]
52 tomcat9                available [ ~stable ]
53 unbound1.13             available [ ~stable ]
54 mariadb10.5             available [ ~stable ]
55 *kern-5.10-distress     enabled [ ~stable ]
56 redis6                 available [ ~stable ]
58 *postgresql12            available [ ~stable ]
59 *postgresql13            available [ ~stable ]
60 mock2                  available [ ~stable ]
61 dnsmasq2.85             available [ ~stable ]
62 kernel-5.15              available [ ~stable ]
```

```

[x] ec2-user@ip-10-0-0-38:~$ sudo yum clean metadata
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-kernel-5.10 amzn2extra-php8.1
17 metadata files removed
6 sqlite files removed
0 metadata files removed
[x] ec2-user@ip-10-0-0-38:~$ sudo yum install php php-common php-pear -y
yum install php-{cgi,curl,mbstring,gd,mysqlnd,gettext,json,xml,fpm,intl,zip} -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-docker
amzn2extra-kernel-5.10
amzn2extra-php8.1
(1/9): amzn2-core/2/x86_64/group_gz | 3.6 kB 00:00:00
(2/9): amzn2-core/2/x86_64/updateinfo | 2.9 kB 00:00:00
(3/9): amzn2extra-docker/2/x86_64/updateinfo | 3.0 kB 00:00:00
(4/9): amzn2extra-php8.1/2/x86_64/updateinfo | 2.9 kB 00:00:00
(5/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo | 2.7 kB 00:00:00
(6/9): amzn2extra-docker/2/x86_64/primary_db | 951 kB 00:00:00
(7/9): amzn2extra-php8.1/2/x86_64/primary_db | 16 kB 00:00:00
(8/9): amzn2extra-kernel-5.10/2/x86_64/primary_db | 5.7 kB 00:00:00
(8/9): amzn2extra-kernel-5.10/2/x86_64/primary_db | 74 kB 00:00:00
(9/9): amzn2-core/2/x86_64/primary_db | 185 kB 00:00:00
(9/9): amzn2-core/2/x86_64/primary_db | 125 kB 00:00:00
(9/9): amzn2-core/2/x86_64/primary_db | 28 MB 00:00:00
(9/9): amzn2-core/2/x86_64/primary_db | 69 MB 00:00:00
Resolving Dependencies
--> Running transaction check
--> Package php.x86_64 0:8.1.29-1.amzn2 will be installed
--> Processing Dependency: php-cli(x86-64) = 8.1.29-1.amzn2 for package: php-8.1.29-1.amzn2.x86_64
--> Package php-common.x86_64 0:8.1.29-1.amzn2 will be installed
--> Processing Dependency: libzip.so.5()(64bit) for package: php-common-8.1.29-1.amzn2.x86_64
--> Package php-pear.noarch 1:1.10.12-9.amzn2 will be installed
--> Processing Dependency: php-posix for package: 1:php-pear-1.10.12-9.amzn2.noarch
--> Processing Dependency: php-xml for package: 1:php-pear-1.10.12-9.amzn2.noarch
--> Running transaction check
--> Package libzip.x86_64 0:1.3.2-1.amzn2.0.1 will be installed
--> Package php-cli.x86_64 0:8.1.29-1.amzn2 will be installed
--> Package php-process.x86_64 0:8.1.29-1.amzn2 will be installed
--> Package php-xml.x86_64 0:8.1.29-1.amzn2 will be installed
--> Processing Dependency: libxml.so.1(LIBXML2_1.0.11)(64bit) for package: php-xml-8.1.29-1.amzn2.x86_64
--> Processing Dependency: libxmltr-enc-1(LIBXML2_1.0.13)(64bit) for package: php-xml-8.1.29-1.amzn2.x86_64
--> Processing Dependency: libxml.so.1(LIBXML2_1.0.18)(64bit) for package: php-xml-8.1.29-1.amzn2.x86_64
--> Processing Dependency: libxml.so.1(LIBXML2_1.0.22)(64bit) for package: php-xml-8.1.29-1.amzn2.x86_64
--> Processing Dependency: libxml.so.1(LIBXML2_1.0.24)(64bit) for package: php-xml-8.1.29-1.amzn2.x86_64
--> Processing Dependency: libxml.so.0()(64bit) for package: php-xml-8.1.29-1.amzn2.x86_64
--> Running transaction check
--> Package libxml.x86_64 0:1.12.8-6.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

```

```

[x] ec2-user@ip-10-0-0-38:~$ sudo yum install php-{cgi,curl,mbstring,gd,mysqlnd,gettext,json,xml,fpm,intl,zip} -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Existing lock /var/run/yum.pid: another copy is running as pid 3553.
Another app is currently holding the yum lock; waiting for it to exit...
The other application is: yum
  Memory : 208 M RSS (428 MB VSZ)
  Started: Wed Aug  7 16:44:00 2024 - 00:04 ago
  State   : Running, pid: 3553
Another app is currently holding the yum lock; waiting for it to exit...
The other application is: yum
  Memory : 211 M RSS (433 MB VSZ)
  Started: Wed Aug  7 16:44:00 2024 - 00:06 ago
  State   : Running, pid: 3553
Package php-cli-8.1.29-1.amzn2.x86_64 already installed and latest version
Package php-common-8.1.29-1.amzn2.x86_64 already installed and latest version
Package php-common-8.1.29-1.amzn2.x86_64 already installed and latest version
Package php-common-8.1.29-1.amzn2.x86_64 already installed and latest version
Package php-xml-8.1.29-1.amzn2.x86_64 already installed and latest version
Package php-common-8.1.29-1.amzn2.x86_64 already installed and latest version
Resolving Dependencies
--> Running transaction check
--> Package php-fpm.x86_64 0:8.1.29-1.amzn2 will be installed
--> Package php-gd.x86_64 0:8.1.29-1.amzn2 will be installed
--> Package php-intl.x86_64 0:8.1.29-1.amzn2 will be installed
--> Package php-mbstring.x86_64 0:8.1.29-1.amzn2 will be installed
--> Processing Dependency: libonig.so.2()(64bit) for package: php-mbstring-8.1.29-1.amzn2.x86_64
--> Package php-mysqlnd.x86_64 0:8.1.29-1.amzn2 will be installed
--> Processing Dependency: php-pdo(x86-64) = 8.1.29-1.amzn2 for package: php-mysqlnd-8.1.29-1.amzn2.x86_64
--> Running transaction check
--> Package oniguruma.x86_64 0:5.9.6-1.amzn2.0.7 will be installed
--> Package php-pdo.x86_64 0:8.1.29-1.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

Transaction Summary

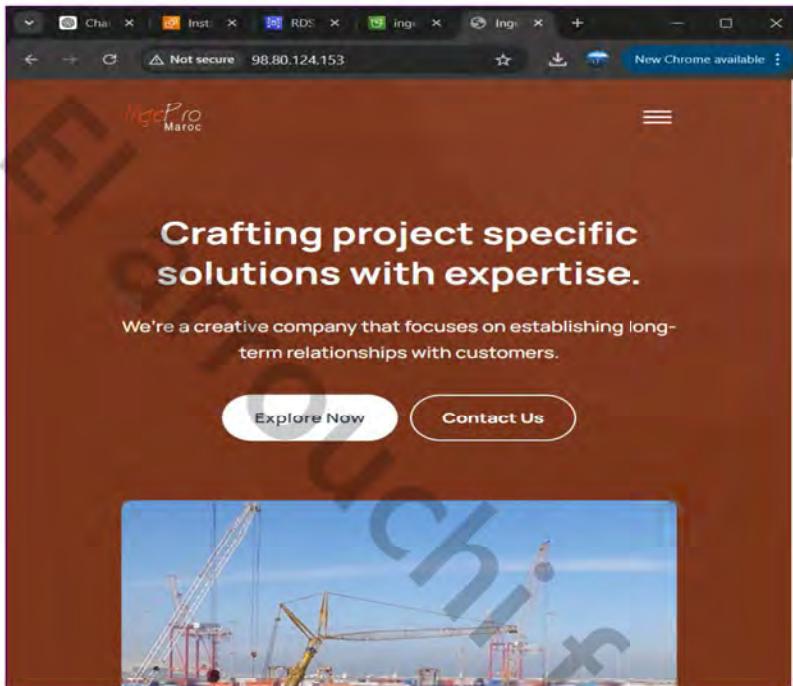
```

Package	Arch	Version	Repository	Size
Installing:				
php-fpm	x86_64	8.1.29-1.amzn2	amzn2extra-php8.1	1.9 M
php-gd	x86_64	8.1.29-1.amzn2	amzn2extra-php8.1	170 k
php-intl	x86_64	8.1.29-1.amzn2	amzn2extra-php8.1	182 k
php-mbstring	x86_64	8.1.29-1.amzn2	amzn2extra-php8.1	475 k
php-mysqlnd	x86_64	8.1.29-1.amzn2	amzn2extra-php8.1	184 k
Installing for dependencies:				
oniguruma	x86_64	5.9.6-1.amzn2.0.7	amzn2-core	127 k
php-pdo	x86_64	8.1.29-1.amzn2	amzn2extra-php8.1	125 k

```
[ec2-user@ip-10-0-0-38 ~]$ sudo systemctl enable mysqld
[ec2-user@ip-10-0-0-38 ~]$ sudo systemctl start mysqld
[ec2-user@ip-10-0-0-38 ~]$ 
[ec2-user@ip-10-0-0-38 ~]$ sudo aws s3 sync s3://inge-pro-maroc-web-files /var/www/html
download: s3://inge-pro-maroc-web-files/ingepromaroc-v1.2.zip to ../../var/www/html/ingepromaroc-v1.2.zip
[ec2-user@ip-10-0-0-38 ~]$
```

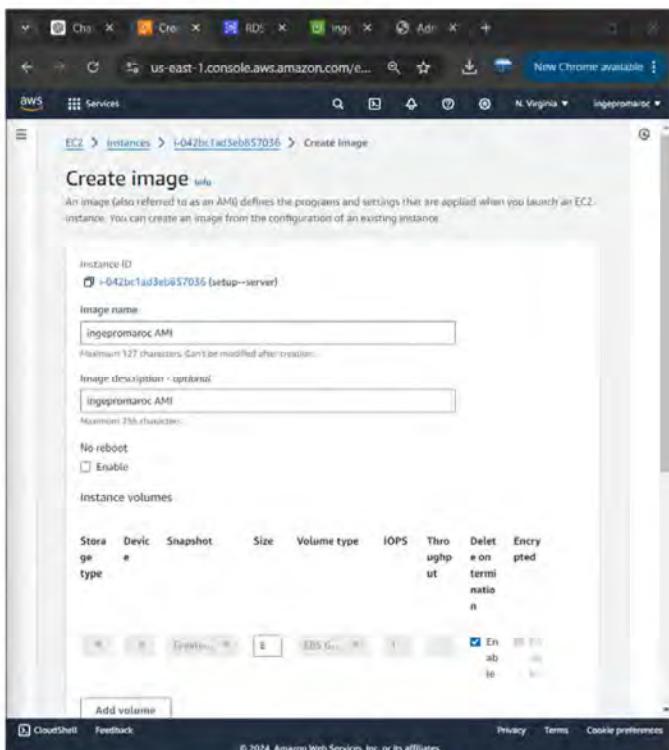
```
[ec2-user@ip-10-0-0-38/var/www/html]
[ec2-user@ip-10-0-0-38 ~]$ sudo aws s3 sync s3://inge-pro-maroc-web-files /var/www/html
download: s3://inge-pro-maroc-web-files/ingepromaroc-v1.2.zip to ../../var/www/html/ingepromaroc-v1.2.zip
[ec2-user@ip-10-0-0-38 ~]$ clear
[ec2-user@ip-10-0-0-38 ~]$ [ec2-user@ip-10-0-0-38 ~]$ cd /var/www/html
[ec2-user@ip-10-0-0-38 html]$ sudo unzip ingepromaroc-v1.2.zip
Archive: ingepromaroc-v1.2.zip
  creating: ingepromaroc-v0.8/
  inflating: ingepromaroc-v0.8/about.php
  creating: ingepromaroc-v0.8/assets/
  creating: ingepromaroc-v0.8/assets/css/
  inflating: ingepromaroc-v0.8/assets/css/style.css
  inflating: ingepromaroc-v0.8/assets/css/style1.css
  extracting: ingepromaroc-v0.8/assets/css/style2.css
  inflating: ingepromaroc-v0.8/assets/css/style3.css
  inflating: ingepromaroc-v0.8/assets/css/style4.css
  inflating: ingepromaroc-v0.8/assets/css/style5.css
  inflating: ingepromaroc-v0.8/assets/css/style6.css
  inflating: ingepromaroc-v0.8/assets/css/style7.css
  inflating: ingepromaroc-v0.8/assets/css/style8.css
  inflating: ingepromaroc-v0.8/assets/css/style9.css
  inflating: ingepromaroc-v0.8/assets/css/style10.css
  inflating: ingepromaroc-v0.8/assets/css/style11.css
  inflating: ingepromaroc-v0.8/assets/css/style12.css
  inflating: ingepromaroc-v0.8/assets/css/style13.css
  inflating: ingepromaroc-v0.8/assets/css/style14.css
  inflating: ingepromaroc-v0.8/assets/css/style15.css
  inflating: ingepromaroc-v0.8/assets/css/style16.css
  creating: ingepromaroc-v0.8/assets/database/
  inflating: ingepromaroc-v0.8/assets/database/ingepro_maroc (3).sql
  creating: ingepromaroc-v0.8/assets/images/
  inflating: ingepromaroc-v0.8/assets/images/body.png
  inflating: ingepromaroc-v0.8/assets/images/hero-bg.jpg
  inflating: ingepromaroc-v0.8/assets/images/hero-slide-1.jpg
  inflating: ingepromaroc-v0.8/assets/images/hero-slide-2.jpg
  inflating: ingepromaroc-v0.8/assets/images/hero-slide-3.jpg
  inflating: ingepromaroc-v0.8/assets/images/ingepro1.png
  inflating: ingepromaroc-v0.8/assets/images/ingepro1d.png
  inflating: ingepromaroc-v0.8/assets/images/pj1.1.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj1.2.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj2.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj3.1.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj3.2.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj4.1.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj4.2.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj5.1.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj5.2.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj6.1.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj6.2.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj7.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj8.1.jpg
  inflating: ingepromaroc-v0.8/assets/images/pj8.2.jpg
  inflating: ingepromaroc-v0.8/assets/images/stats-bg.jpg
```

Now if you type the public IPV4 address in your browser you should be able to access your website.



XI.

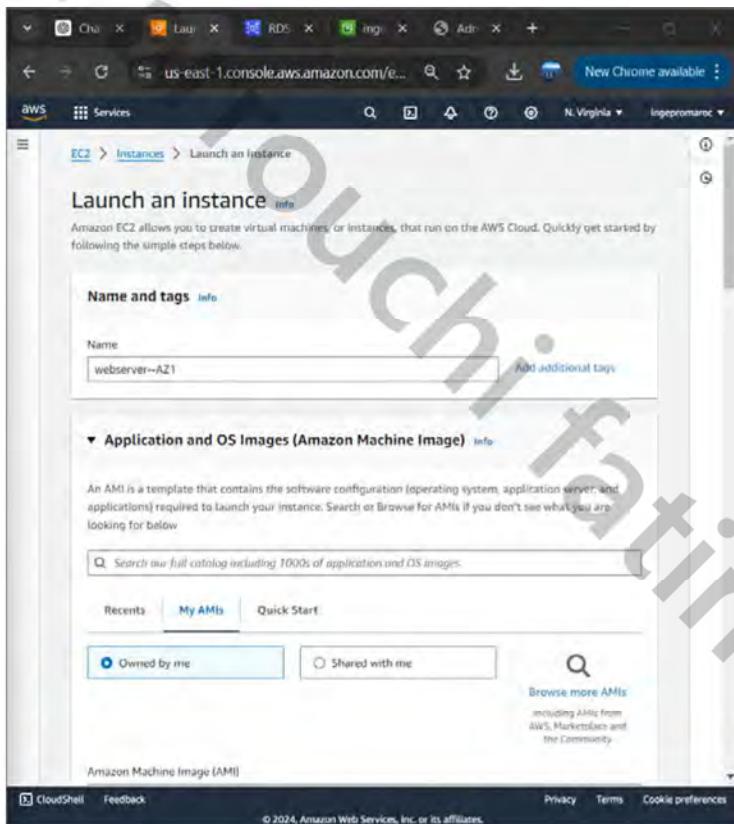
→ Create an AMI:



Give your image a name and use the same name as a description, leave the other options as default then save changes.

XII.

→ launch an EC2 instance:



Name your instance and under AMI choose my AMIs and select owned by me.

The screenshot shows the AWS CloudFormation console with the following details:

- Amazon Machine Image (AMI)**:
 - Name: ingepromaroc AMI
 - Created: 2024-08-07T17:48:55.000Z
 - Virtualization: hvf
 - ENAs enabled: true
 - Root device type: mva
- Description**: ingepromaroc AMI
- Architecture**: x86_64
- AMI ID**: ami-0991adc324a1bdf61
- Instance type**:
 - t2.micro
 - Family: t2
 - 1 vCPU
 - 1 GiB Memory
 - Demand generation
 - Free tier eligible
 - On-Demand Windows base pricing: 0.0182 USD per Hour
 - On-Demand SUSE base pricing: 0.0116 USD per Hour
 - On-Demand RHEL base pricing: 0.026 USD per Hour
 - On-Demand Linux base pricing: 0.0116 USD per Hour
- Key pair (login)**: my-ec2-keypair

Choose your AMI, t2.micro as instance type and you key pair.

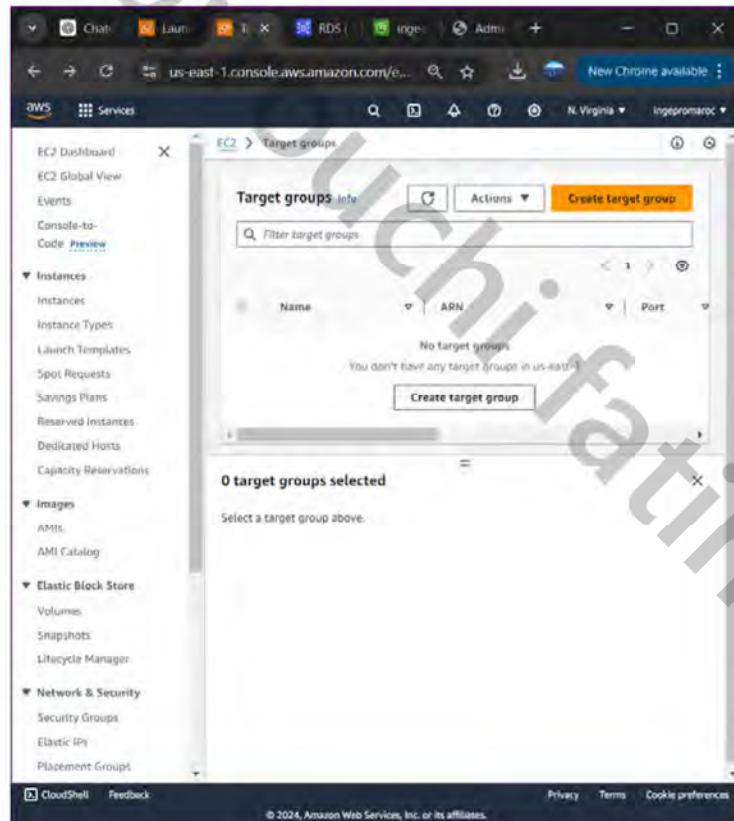
The screenshot shows the AWS CloudFormation console with the following network settings configuration:

- VPC - required**:
 - vpc-Dec9884e3bdd4a21e (dev--vpc)
 - 10.0.0.0/16
- Subnet**:
 - subnet-0debf1ffbbfb6fb5e private--app--subnet--AZ1
 - VPC: vpc-Dec9884e3bdd4a21e Owner: 017820637328
 - Availability Zone: us-east-1a Zone type: Availability Zone
 - IP addresses available: 251 CIDR: 10.0.0.0/24
- Auto-assign public IP**: Disable
- Firewall (security groups)**:
 - Create security group
 - Select existing security group (selected)
- Common security groups**:
 - webserver-security-group sg-0f503ce6aaf22db03
- Configure storage**:
 - 1x 8 GiB gp2 Root volume (Not encrypted)

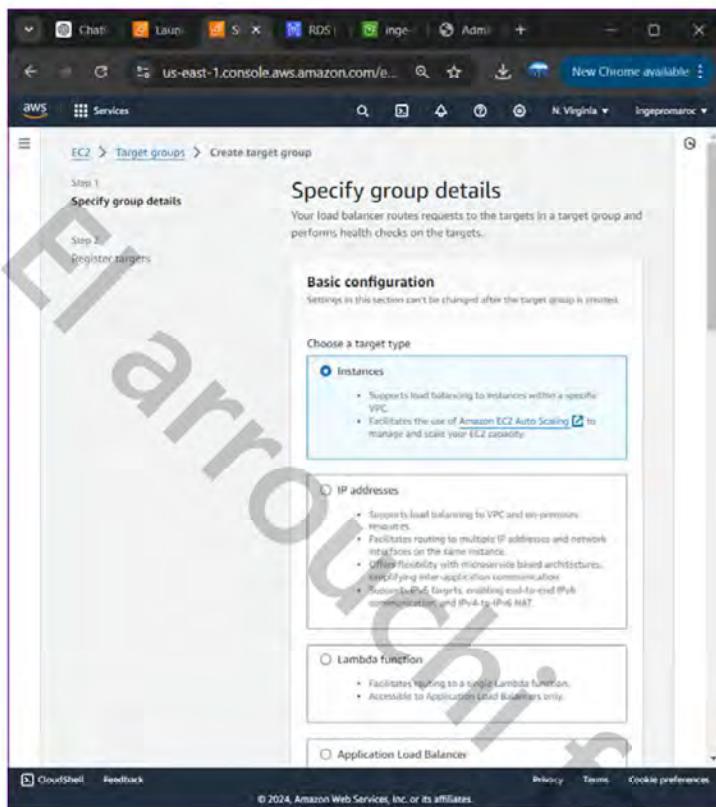
Select your VPC, private app subnet in the first availability zone and webserver security group then hit save.

→ Create a target group:

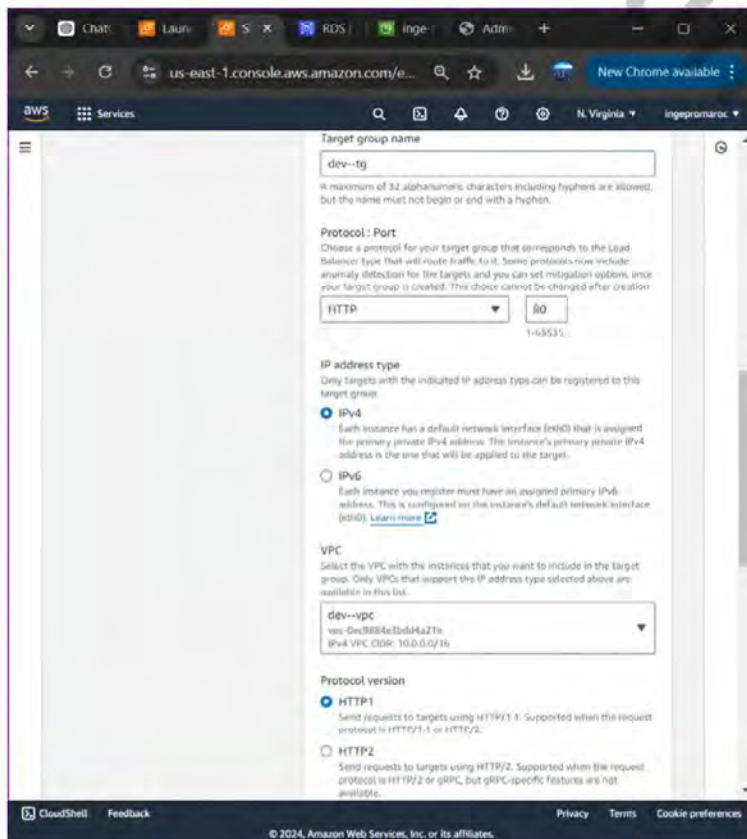
Creating a target group directs traffic to specific resources, optimizing load distribution and performance.



In the same dashboard select target groups then create one

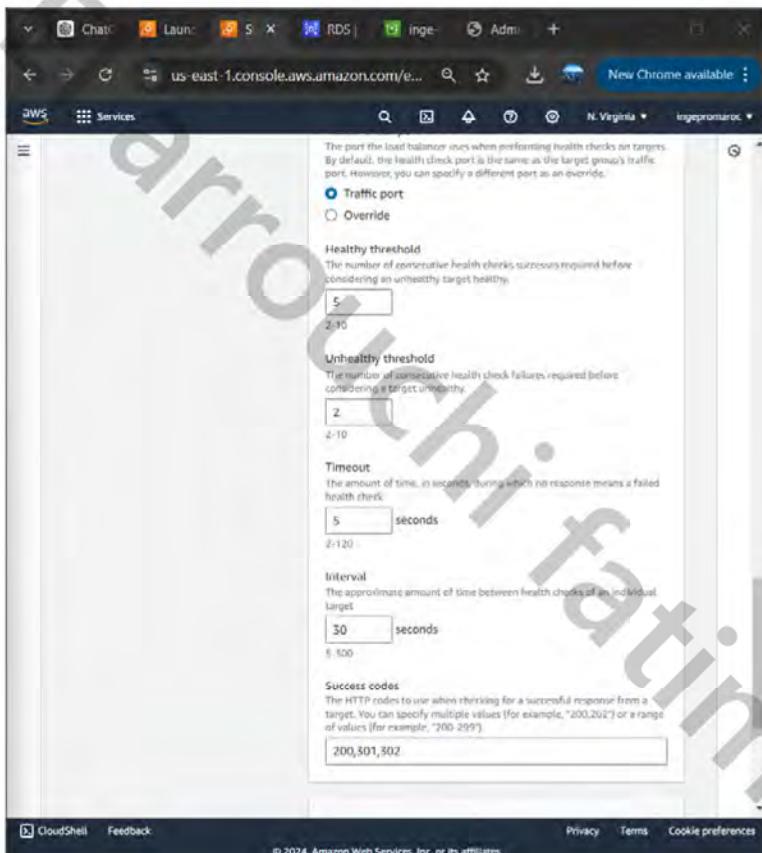


Choose instances as target type to be able to redirect traffic to our EC2 instance.



Give it a name and select http as protocol since it allows the load balancer to route traffic based on standard web requests and enables health checks.

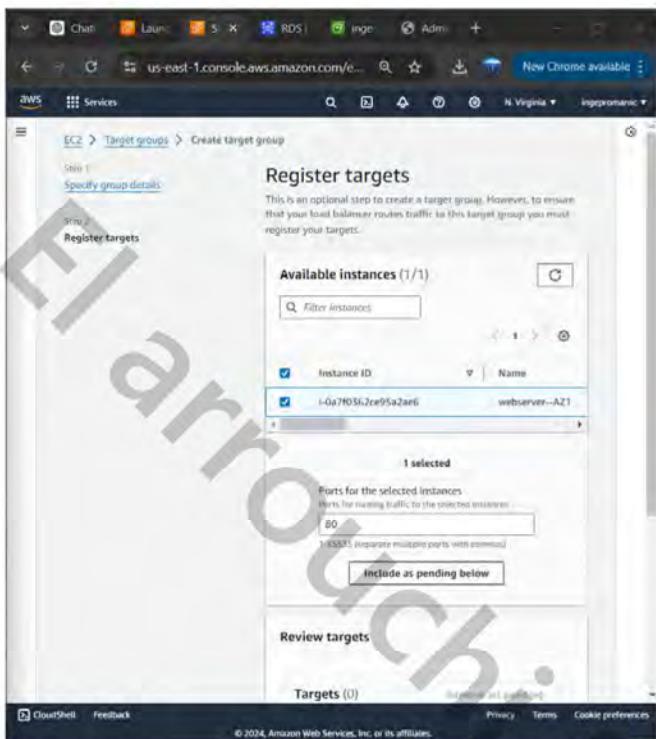
Choose IPV4, your VPC and HTTP1 as version.



Leave all these options as default except success codes add 200,301,302 because :

200 indicates a successful HTTP request.

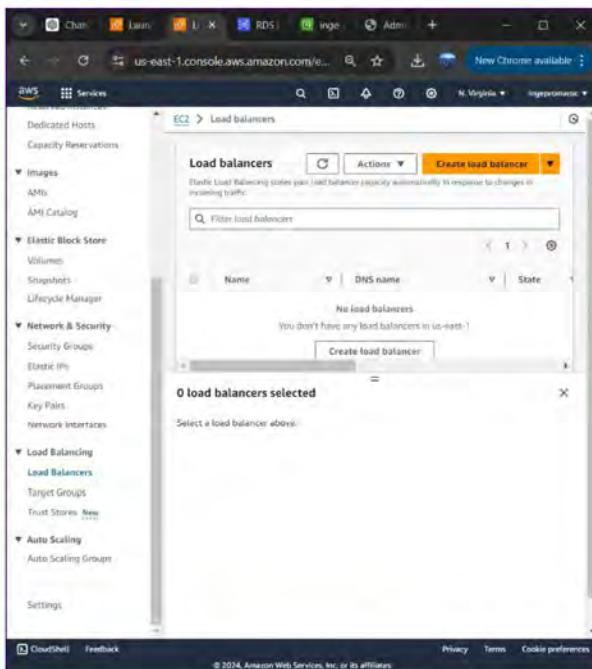
301 and 302 indicate successful URL redirections.



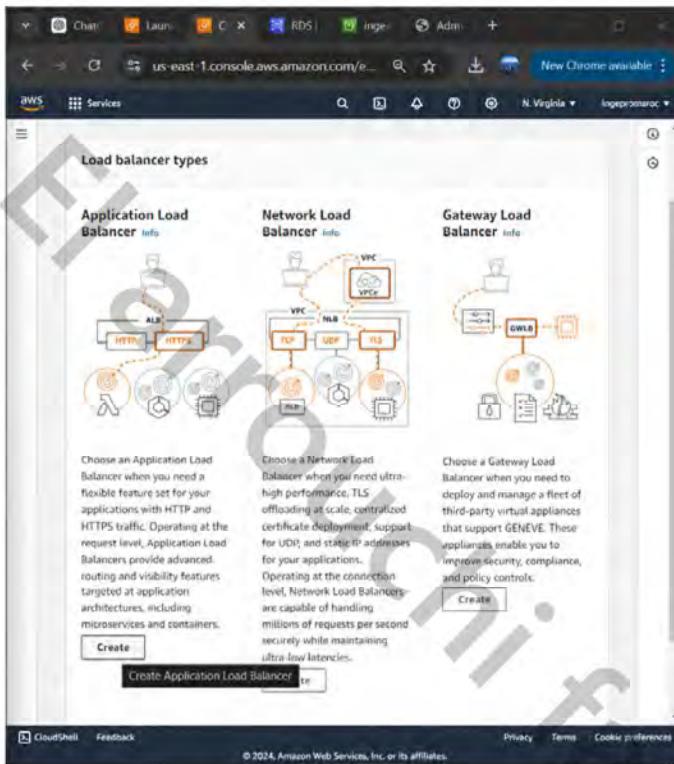
Select webserver AZ1 instance and add port 80 then hit save.

→ Create an application load balancer:

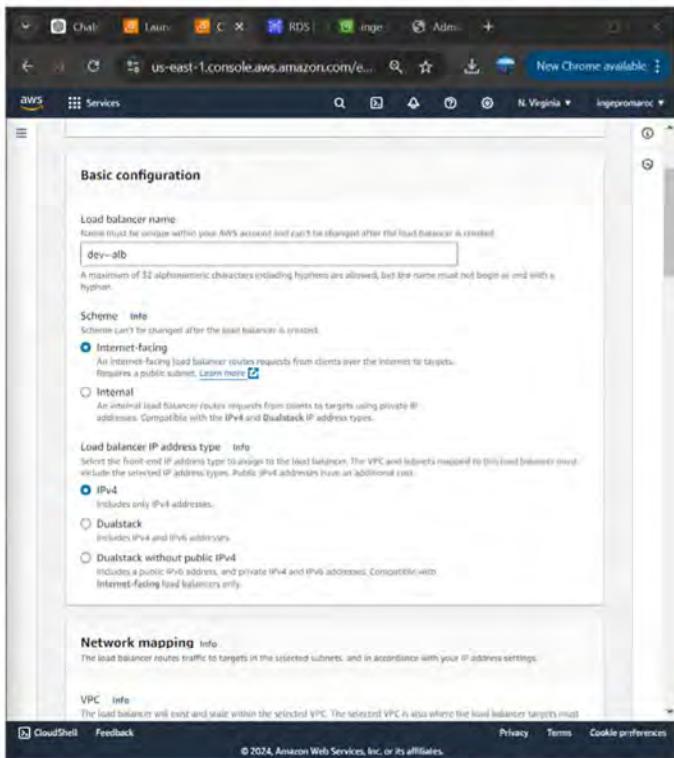
distributes traffic for better availability, scalability, and content-based routing.



In the EC2 dashboard look for load balancers, create one.



For the types choose Application Load Balancer and hit that create button.



Name your ALB and make it internet-facing since we are dealing with public subnets.

The screenshot shows the 'Network mapping' section of the AWS CloudFront console. It displays two public subnets selected for the load balancer:

- Subnet 1:** subnet-03a274c52b28d3a3d (IPv4 subnet CIDR: 10.0.0.0/24)
- Subnet 2:** subnet-026d16a462ae8ddfa (IPv4 subnet CIDR: 10.0.1.0/24)

Both subnets are assigned by AWS.

Select the your VPC, for the availability zones choose the 2 public subnets

The screenshot shows the 'Listeners and routing' configuration for an ALB. It defines a listener for port 80 (HTTP) that forwards traffic to a target group named 'dev-tg'. The target type is set to 'instance'.

Protocol	Port	Default action
HTTP	80	Forward to: dev-tg (Target type: instance, this)

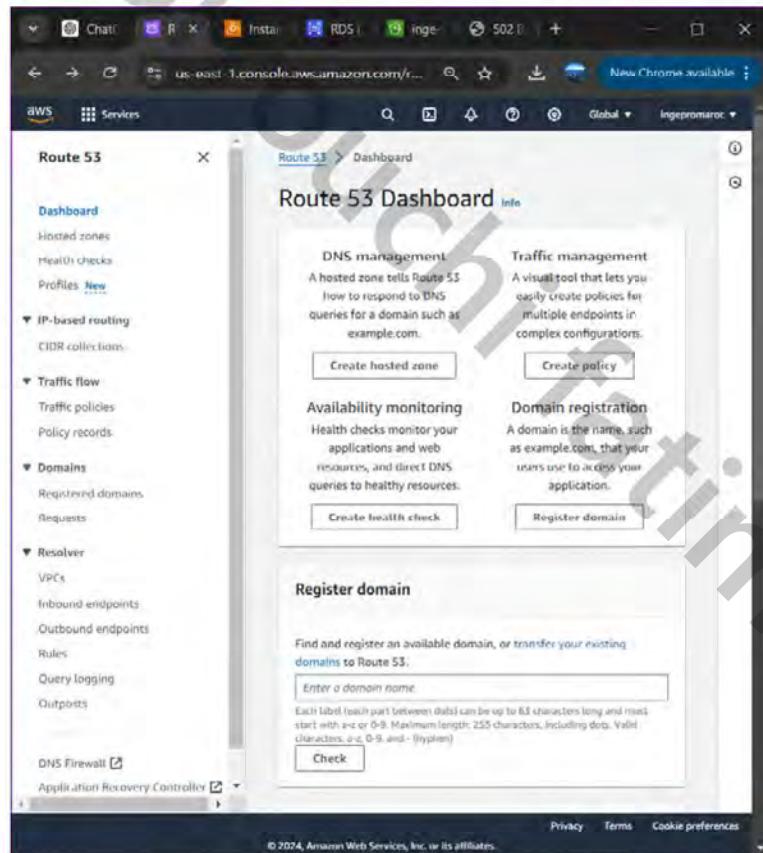
Listener tags - optional: You can add up to 50 more tags.

Choose the application load balancer security group, and port http (target group) finally save changes

XIII.

→ register a domain name:

It gives your website a unique, easily recognizable address.



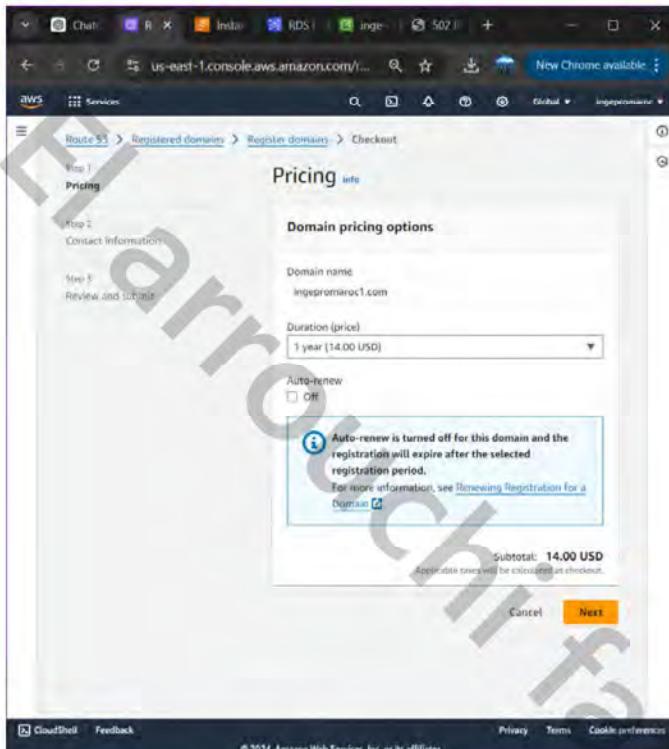
In the aws console search box type Route53.

The screenshot shows the AWS Route 53 service dashboard. On the left, there's a navigation sidebar with various options like Dashboard, Hosted zones, Health checks, IP-based routing, Traffic flow, Domains, Resolver, VPCs, and DNS Firewall. The main content area is titled "Registered domains" and contains a search bar, a "Transfer in" button, and a "Register domains" button. Below these are sections for "Domain name", "Expiration date", "Auto-renew", and "Transfer ID". A message states "No domains to display." At the bottom, there are links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

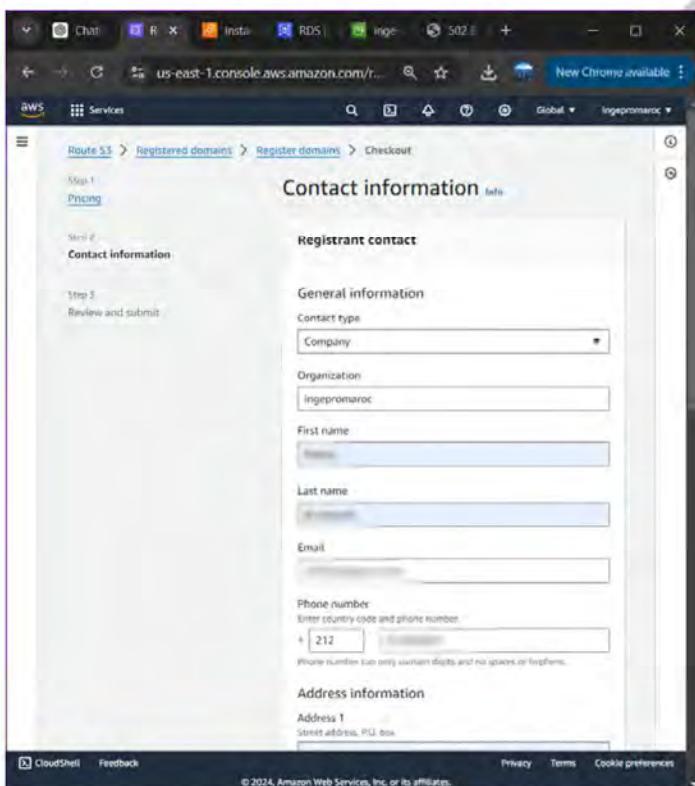
Under registered domains click “register domains”.

This screenshot shows the "Register domains" sub-page within the Route 53 service. It has a header with "Route 53 > Registered domains > Register domains". Below this is a section for "Search for domain" with a search bar containing "ingepromaroc1.com" and a "Search" button. Under "Search result", there's a table showing one entry: "ingepromaroc1.com" with a status of "Selected". In the "Suggested available domains (10)" section, there are ten more entries, each with a "Select" button under the "Actions" column. The domains listed are: ingepromaroc1.co, ingeproturisie1.com, ingepromaroc1.info, ingepromaroc124.com, ingeprosanegat1.com, ingepromaroc1.com, ingepromaroc1.com, ingepromaroc1.com, ingepromaroc1.com, and ingepromaroc1.com.

Check whether the domain name you would like is available or not, if it is select it and checkout.



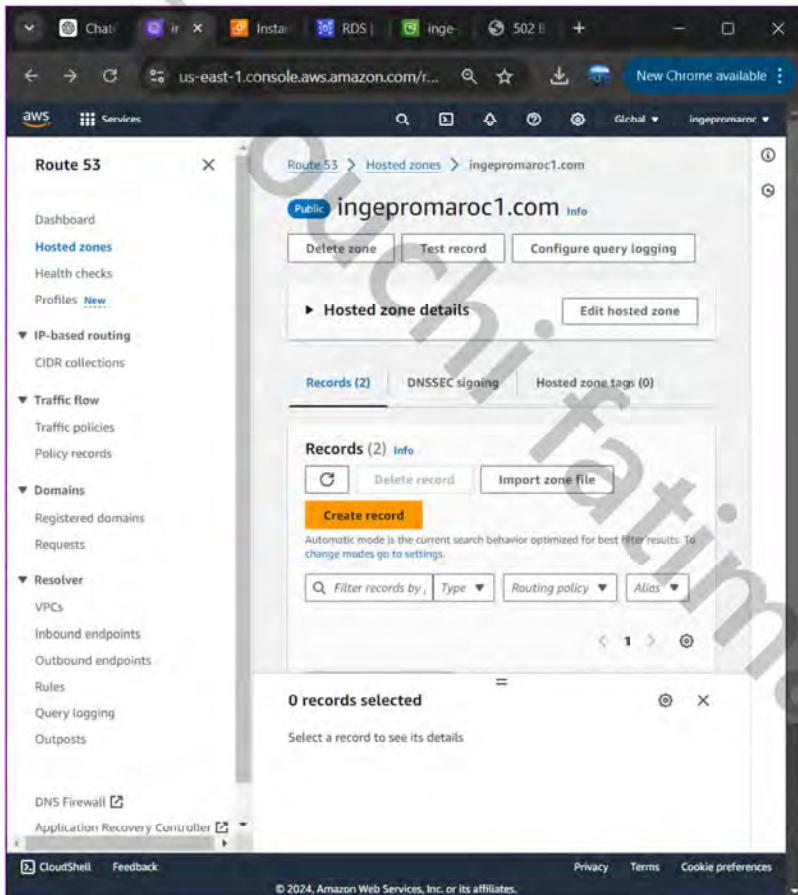
Select the duration compatible for you.



Fill in all your company's informations then hit continue.

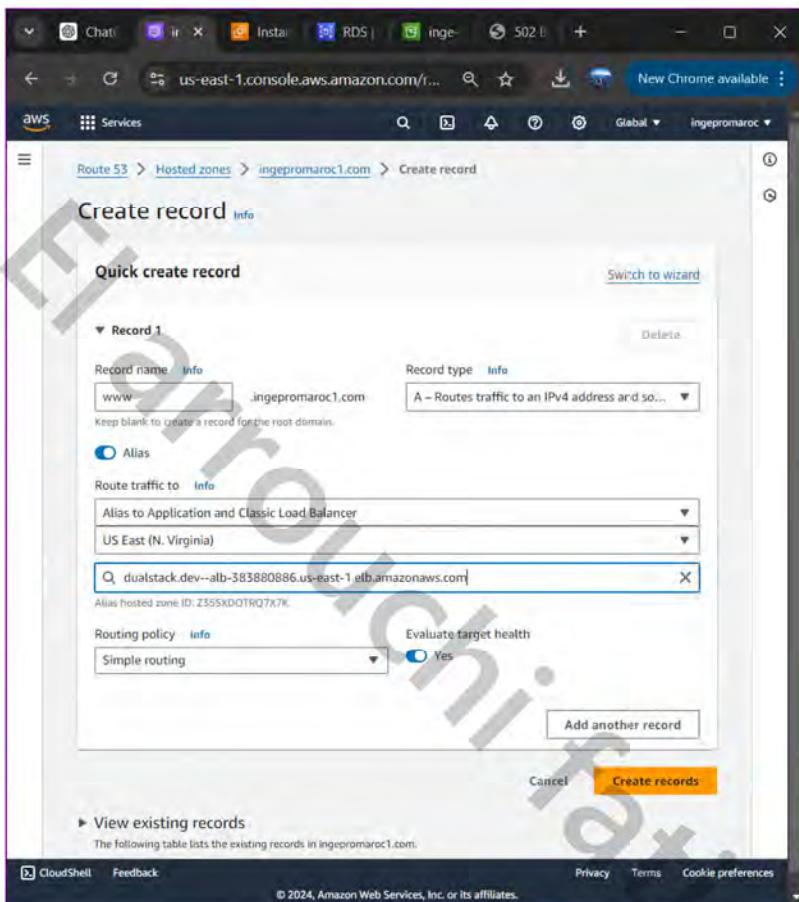
→ Create a record:

To link your domain name to your resources, enabling traffic routing to your website or application.



The screenshot shows the AWS Route 53 service in a web browser. The left sidebar contains navigation links for Dashboard, Hosted zones (selected), Health checks, Profiles, IP-based routing, Traffic flow, Domains, Resolver, DNS Firewall, Application Recovery Controller, CloudShell, and Feedback. The main content area displays the 'Hosted zones > ingepromaroc1.com' page for the domain 'ingepromaroc1.com'. It includes options to Delete zone, Test record, and Configure query logging. Below this is the 'Hosted zone details' section with an 'Edit hosted zone' button. The 'Records (2)' tab is selected, showing two existing records. A prominent orange 'Create record' button is located at the bottom of this section. A note below it states: 'Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.' At the bottom of the page, there are filters for 'Filter records by', 'Type', 'Routing policy', and 'Alias', along with a message: '0 records selected' and 'Select a record to see its details'. The footer includes links for Privacy, Terms, and Cookie preferences, and a copyright notice: '© 2024, Amazon Web Services, Inc. or its affiliates.'

Select your domain name and click “create record”.

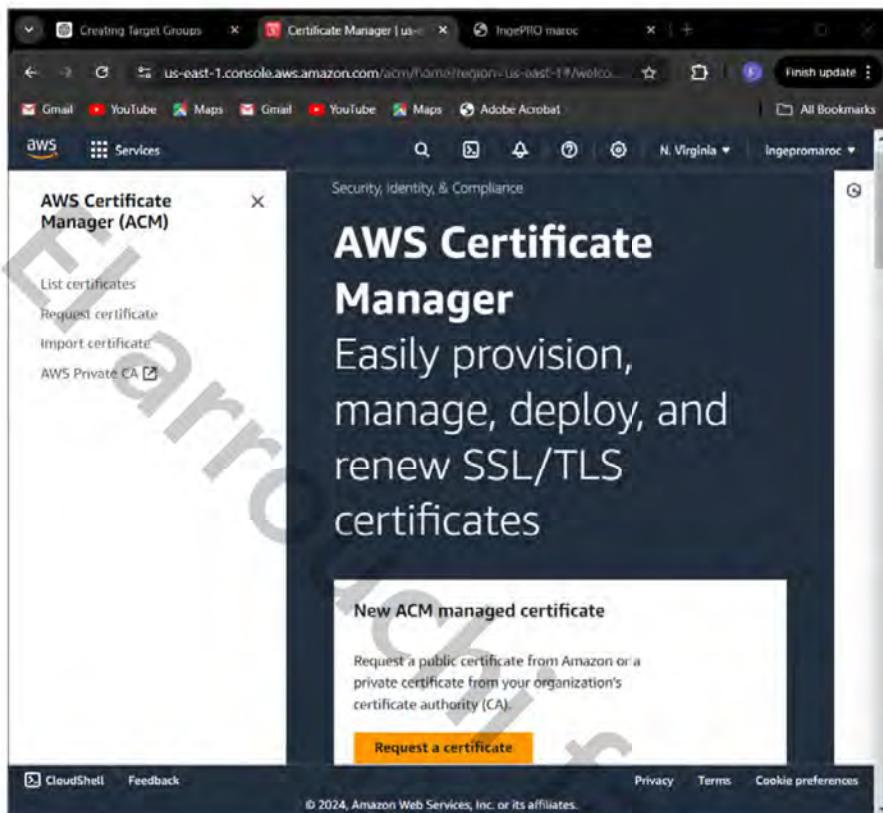


In the record name input type www and set record type to A in order to create an address for "www.ingepromaroc1.com" that directs traffic to your server's IP, Route the traffic to Alias to app and classic load balancer, select the US east and dualstack for the hosted zone.

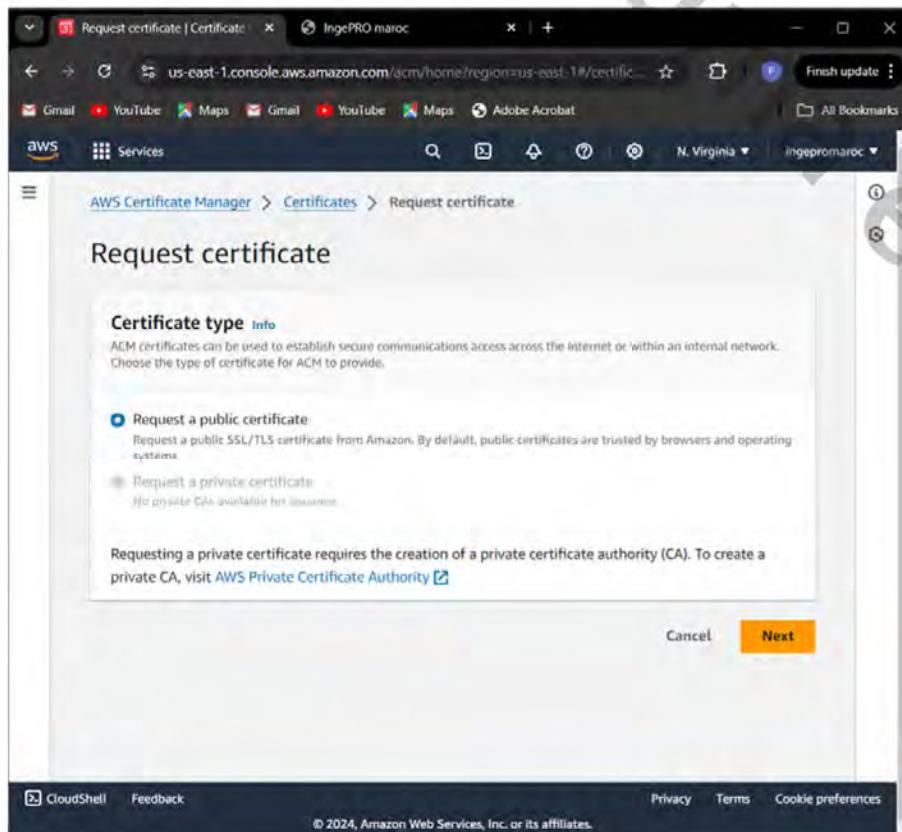
XIV.

→ Register for a SSL certificate:

To encrypt all the communications between the web browser and our website.



In the AWS search box type Certificate Manager and request a certificate.



Select Request a public certificate then click next.

The screenshot shows the 'Request public certificate' step in the AWS Certificate Manager. Under the 'Domain names' section, two domains are listed: 'ingepromaroc.tcom' and '*.ingepromaroc1.com'. There is a link to 'Add another name to this certificate' and a note about adding additional names like 'www.example.com'. The bottom of the page includes standard AWS navigation links and a copyright notice.

Enter your domain name, click add another name to certificate and enter [*.ingepromaroc1.com](http://www.ingepromaroc1.com) to request a certificate for www.ingepromaroc1.com

The screenshot shows the 'Validation method' configuration step. It offers two options: 'DNS validation - recommended' (selected) and 'Email validation'. Below this, the 'Key algorithm' section is shown, with 'RSA 2048' selected. The bottom of the page includes standard AWS navigation links and a copyright notice.

Leave the validation method as DNS validation and key algorithm to RSA then validate.

The screenshot shows the AWS Certificate Manager interface. A certificate named 'cc9601ce-8953-4630-9aad-7b7cdb604b87' is listed. The 'Certificate status' section shows the ARN as 'arn:aws:acm:us-east-1:0117820672226:certificate/cc9601ce-8953-4630-9aad-7b7cdb604b87'. The status is 'Pending validation' with a 'Info' link. Below this, under 'Domains (2)', there are two entries: 'ingepromaroc1.com' and '*.ingepromaroc1.com'. At the bottom, there are buttons for 'Create records in Route 53' and 'Export to CSV'.

as you can see we successfully requested our certificate, the status is still pending validation because we have to create a record set in route53 to improve that this domain name belongs to us.

The screenshot shows the 'Create DNS records in Amazon Route 53 (2/2)' step. It lists two matches: 'Validation status = Pending validation' and 'Is domain in Route 53? = Yes'. There are two items in the table:

Domain	Validation status	Is domain in Route 53?
ingepromaroc1.com	Pending validation	Yes
*.ingepromaroc1.com	Pending validation	Yes

At the bottom, there are 'Cancel' and 'Create records' buttons.

After clicking “create a route53 record” select both your domain name and wild card then click create records.

The screenshot shows the AWS Certificate Details page. At the top, it displays the identifier (cc9601ce-8953-4630-9aad-7b7cdb604b87), status (Issued), and ARN (arn:aws:acm:us-east-1:017820672226:certificate/cc9601ce-8953-4630-9aad-7b7cdb604b87). The Type is listed as Amazon Issued. Below this, under Domains (2), there are two entries: ingepromaroc1.com and *.ingepromaroc1.com, both with a status of Success. There are buttons for "Create records in Route 53" and "Export to CSV".

→ secure our website with ACM SSL certificate:

Securing our website with an AWS ACM SSL certificate protects user data, boosts trust and SEO, ensures compliance, and simplifies management.

The screenshot shows the AWS Load Balancer configuration page. On the left sidebar, the navigation menu includes Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, Load Balancers, Target Groups, Auto Scaling, Launch Configurations, and Auto Scaling Groups. The main content area displays an Application Load Balancer (ALB) configuration. It shows the DNS name (Dev-ALB-251828045.us-east-1.elb.amazonaws.com (A Record)), Status (Active), VPC (vpc-047e879193d0f9347), and Availability Zones (subnet-0987af0d0ede8ce95, us-east-1b (use1-az2), subnet-0be2d9f6b00fb60a, us-east-1a (use1-az6)). The listeners tab is selected, showing one listener for Protocol:Port (HTTP:80) with ARN (Not Applicable), Security policy (Not Applicable), Default SSL cert (Default SSL cert), and Default routing rule (Forward to 1. http://1 (100%), Group-level stickiness: Off).

In the EC2 dashboard, under load balancers there is only one listener for HTTP we should create another one for HTTPS so click “Add listener”.

Add listener

Details

Listener details

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Protocol: HTTPS Port: 443

Default actions: 1. Forward to

Target group: dev--tg Weight (0-999)

Select a target group

Create target group

Choose HTTPS from the protocol dropdown (Port 443).

Secure listener settings

Security policy

Your load balancer uses a Secure Socket Layer (SSL) negotiation configuration, known as a security policy, to negotiate SSL connections with clients.

ELBSecurityPolicy-2016-08

Compare security policies

Default SSL/TLS certificate

Choose your dev—tg

The screenshot shows the 'Add listener' configuration page for an AWS Load Balancer. The 'Secure listener settings' section is active. Under 'Default SSL/TLS certificate', a dropdown menu shows 'From ACM' selected, with the value 'ingepromaroc1.com' and the ID 'f427970e-28a7-4910-8a51-c3fd70fd777f'. There is also a link to 'Request new ACM certificate'. At the bottom of the page, there is a 'Tags - optional' section and an 'Add' button.

From ACM select the SSL certificate then hit "Add"

The screenshot shows the 'Add listener' confirmation page. A green header bar indicates 'Successfully created listener.' Below it, the breadcrumb navigation shows 'EC2 > Load balancers > Dev-ALB > Add listener'. The main content area displays the message 'Add listener' and a 'Suggested next steps' section with a link to 'Review or customize your listener. Edit listener'. At the bottom right is an orange 'View listeners' button.

We have successfully created our listener.

The screenshot shows the AWS Application Load Balancer configuration page. At the top, it displays basic information: Load balancer type: Application Load Balancer, DNS name: Dev-ALB-251828045.us-east-1.elb.amazonaws.com (A Record), Status: Active, and VPC: vpc-047e879193d89f347. Below this, IP address type is set to IPv4 and Scheme is Internet-facing. Availability Zones include subnet-0987af00ede8ce95 (us-east-1b) and subnet-0be2d9f6b00fbfb60a (us-east-1a). The Hosted Zone is Z355XD0TRQ7X7K. The 'Listeners' tab is selected, showing two entries:

Protocol:Port	ARN	Security policy	Default SSL cert	Forward to
HTTP:80	Not Applicable	Not Applicable	Not Applicable	1. Forward to o dev-tg: 1 (100%) o Group-level stickiness: Off
HTTPS:443	ELBSecurityPolicy-2016-08	ingepromaroc1.com (Certificate ID:...)	Not Applicable	1. Forward to o dev-tg: 1 (100%) o Group-level stickiness: Off

As you can see both of our listener are redirecting traffic to our dev—tg, now we have to edit the port 80 listener and redirect its traffic to port 443

The screenshot shows the same AWS Application Load Balancer configuration page, but the port 80 listener is now selected. The 'Actions' dropdown menu is open, showing options: View listener details, Edit listener (which is highlighted in blue), Add SSL certificates, Manage rules, Manage tags, Delete listener, and Get. The 'Edit listener' option is currently active.

Select the port 80 listener click "Actions" and edit listener .

Listener details

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

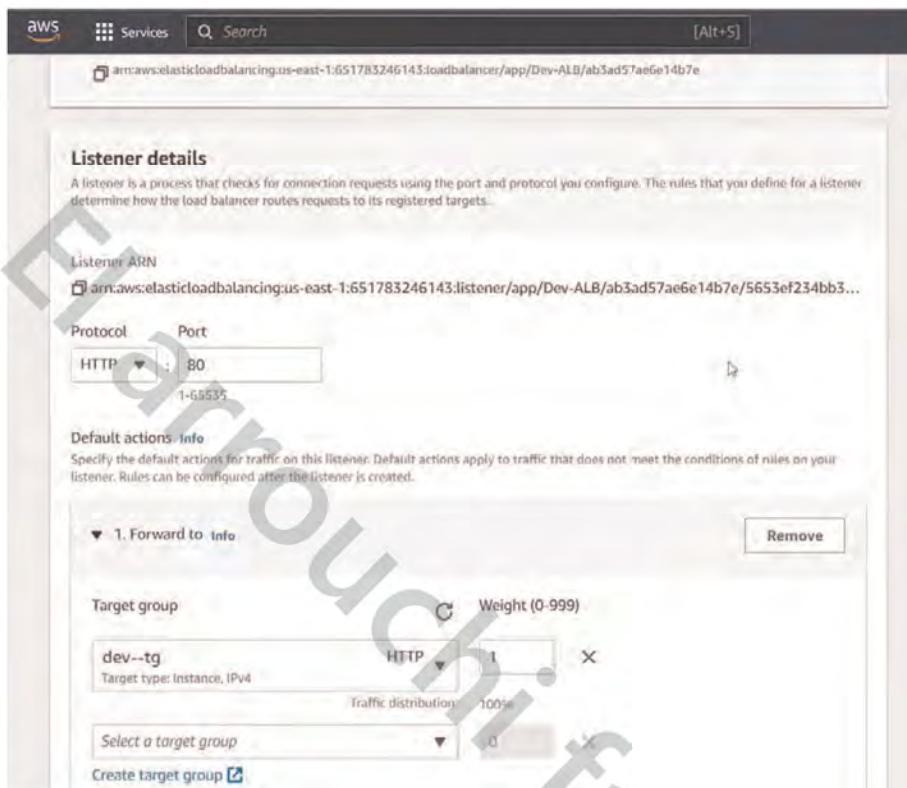
Listener ARN
arn:aws:elasticloadbalancing:us-east-1:651783246143:listener/app/Dev-ALB/ab3ad57ae6e14b7e/5653ef234bb3...

Protocol Port
HTTP ▾ : 80
1-65535

Default actions - Info
Specify the default actions for traffic on this listener. Default actions apply to traffic that does not meet the conditions of rules on your listener. Rules can be configured after the listener is created.

▼ 1. Forward to [Info](#) [Remove](#)

Target group C Weight (0-999)
dev--tg Target type: Instance, IPv4
HTTP ▾ 1 100%
Traffic distribution
Select a target group Create target group [Create target group](#)



In forward to click "Remove".

Details

arn:aws:elasticloadbalancing:us-east-1:651783246143:loadbalancer/app/Dev-ALB/ab3ad57ae6e14b7e/5653ef234bb3...

Listener details

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Listener ARN
arn:aws:elasticloadbalancing:us-east-1:651783246143:listener/app/Dev-ALB/ab3ad57ae6e14b7e/5653ef234bb3...

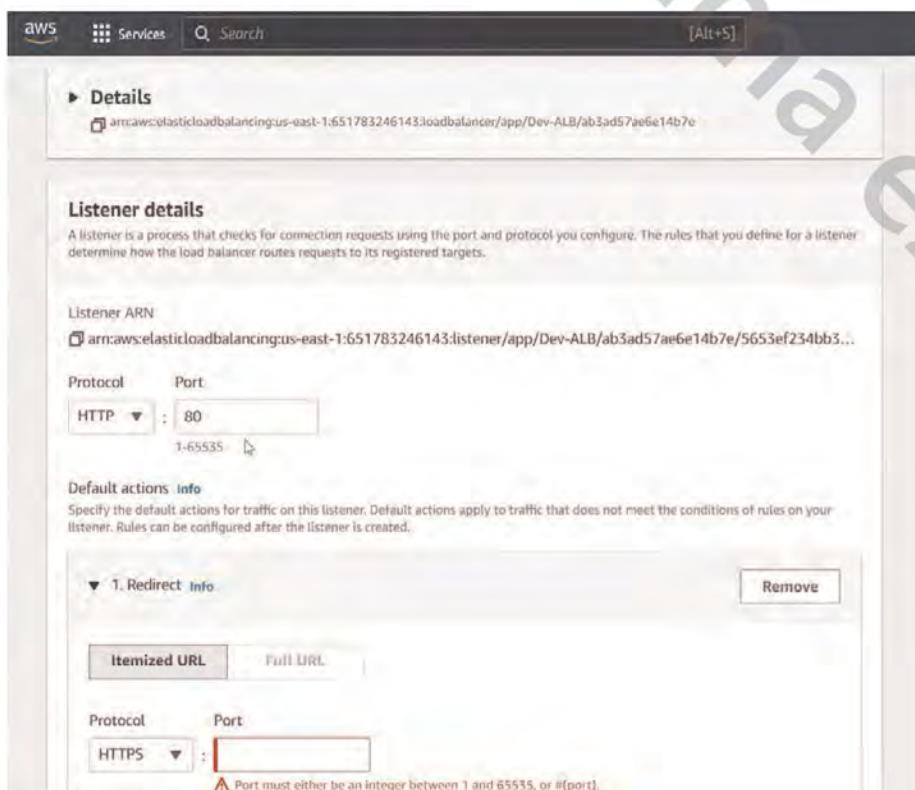
Protocol Port
HTTP ▾ : 80
1-65535

Default actions - Info
Specify the default actions for traffic on this listener. Default actions apply to traffic that does not meet the conditions of rules on your listener. Rules can be configured after the listener is created.

▼ 1. Redirect [Info](#) [Remove](#)

Itemized URL Full URL

Protocol Port
HTTPS ▾ :
⚠ Port must either be an integer between 1 and 65535, or #[port].



From the default actions dropdown select Redirect, enter 443 in the Port.

The screenshot shows the AWS Lambda Listener configuration page. A red box highlights the 'Protocol' dropdown set to 'HTTPS' and the 'Port' input field containing '443'. Below these, the 'Original host, path, query' radio button is selected, and the status code is set to '301 - Permanently moved'. A 'Tags - optional' section is visible at the bottom.

Leave all other options as default then save changes.

The screenshot shows the AWS Application Load Balancer (ALB) configuration page. It displays basic load balancer details like type, DNS name, and VPC. The 'Listeners' tab is active, showing two entries: 'HTTP:80' and 'HTTPS:443'. The 'HTTPS:443' entry has a red box around it, indicating it's the one being configured. The 'Actions' dropdown for this listener is open, showing a 'Redirect to HTTPS://#(host):443/#[path]?#(query)' option with a 'Status code: HTTP_301' sub-option selected.

As you can see we have successfully redirected our traffic to port 443, now our website is going to look just like basic html with no style, in order to fix this we have to SSH into our EC2 instance in the private subnet.

→ SSH into our ec2 instance in the private subnet:

The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. On the left, the 'Name and tags' section has 'Bastion host' entered in the 'Name' field. On the right, the 'Summary' panel shows 'Number of instances' set to 1, 'Software Image (AMI)' as Amazon Linux 2 Kernel 5.10 AMI, 'Virtual server type (instance type)' as t2.micro, and 'Storage (volumes)' as 1 volume(s) - 8 GiB. A tooltip for the free tier is visible, stating: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOPS'. At the bottom right are 'Cancel' and 'Launch instance' buttons.

In the ec2 dashboard launch a new instance, give it a name (bastion host in my case) and amazon linux as AMI.

The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. On the left, the 'Application and OS Images (Amazon Machine Image)' section shows the 'Quick Start' tab selected, displaying various AMI categories like Amazon Linux, macOS, Ubuntu, Windows, and Red Hat. On the right, the 'Summary' panel shows the same configuration as the previous screenshot, including the free tier tooltip. At the bottom right are 'Cancel' and 'Launch instance' buttons.

AWS Services Search [Alt+5] N. Virginia

Key pair (login) Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required
the--ec2--keypair

Network settings Info

Network info
vpc-be482cc3 | default vpc

Subnet info
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

We'll create a new security group called 'launch-wizard-1' with the following rules:

Allow SSH traffic from Anywhere

Helps you connect to your instance

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-000dcdb5067f052a63

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million

Select your key pair

AWS Services Search [Alt+5] N. Virginia

Key pair (login) Info
my-ec2key

Network settings Info

VPC - required Info
vpc-047e879193d89f347 (dev--vpc)
10.0.0.0/16

Subnet info
subnet-0be2d9f6b00fb60a public--subnet--AZ1
VPC vpc-047e879193d89f347 Owner: 651783246143 Availability Zone: us-east-1a IP addresses available: 249 CIDR: 10.0.0.0/24

Auto-assign public IP [Info](#)
Enable

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

Security group name - required
launch-wizard-1

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _~!@#\$%^&*()

Description - required Info
launch-wizard-1 created 2022-12-04T21:23:15.664Z

Summary

Number of instances [Info](#)
1

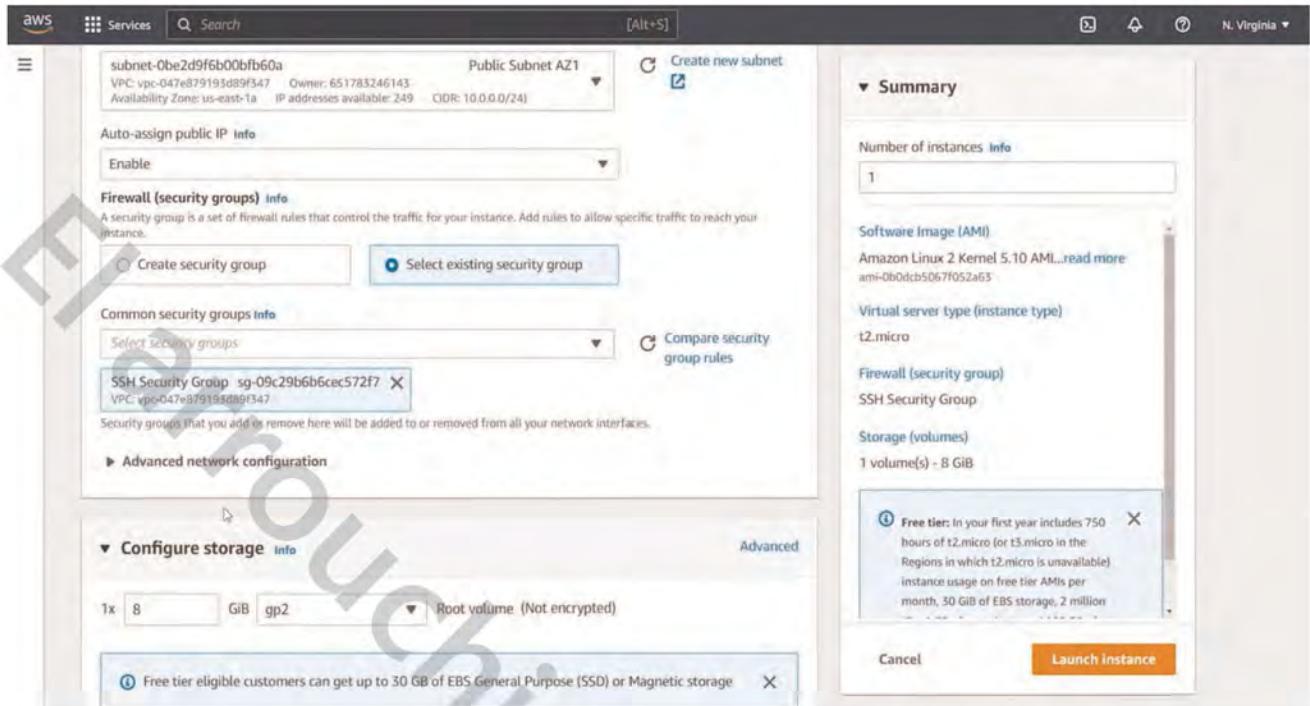
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-000dcdb5067f052a63

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million



Select your VPC and the public subnet AZ1, then choose your SSH security group finally launch instance.

Name	Instance ID	Instance state	Instance type	Status check	Alarm st
webservice--AZ1	i-0d825dcc3a5d9e961	<input checked="" type="checkbox"/> Running	t2.micro	<input checked="" type="checkbox"/> 2/2 checks passed	No alarm
Bastion Host	i-0ccacfb3be051c375	<input checked="" type="checkbox"/> Running	t2.micro	<input type="checkbox"/> Initializing	No alarm

Open powershell as administrator then enter theses commands :

```
# Check if ssh-agent is running
```

```
Get-Service ssh-agent
```

```
# Start the service
```

```
Start-Service ssh-agent
```

```
# This should return a status of Running
```

```
Get-Service ssh-agent
```

```
# Now load your key files into the ssh-agent
```

```
ssh-add C:\Users\fatim\the—ec2--keypair.pem
```

```
close powershell and reopen it normally (not as admin)
```

```
#SSH into the bastion host in the public subnet
```

```
Ssh -A ec2-user@54.198.207.34 {paste public IPV4 address of your bastion host}
```

```
#SSH into webserver az1 in private subnet
```

```
Ssh 10.0.2.204 {paste the private IPV4 address of your webserver AZ1}
```

```
#update the .env file
```

```
Cd /var/www/html
```

```
Sudo vi .env
```

```
{in the .env file type I to go to the insert mode then in the PP_ENV=local remove local  
and type production, for the APP_URL= paste your domain name, hit esc :wq and enter}
```

```
#update the app service provider.php file
```

```
Cd app
```

```
Cd providers/
```

```
Sudo vi AppServiceProvider.php
```

```
{type I to go to the insert mode in the public function boot(){ paste this line of code  
if (env('APP_ENV') === 'production')  
{\Illuminate\Support\Facades\URL::forceScheme('https');}, after that hit esc :wq and  
enter }
```

#restart the Apache server

Sudo service httpd restart

Now our website should load properly and the styles are applied.

→ Create an AMI:

The screenshot shows the AWS EC2 Instances page. There are two instances listed: 'webserver--AZ1' (running, t2.micro) and 'Bastion Host' (running, t2.micro). The 'Actions' dropdown menu is open for the 'webserver--AZ1' instance, with 'Image and templates' selected. Other options in the dropdown include 'Connect', 'View details', 'Manage instance state', 'Instance settings', 'Networking', 'Security', and 'Monitor and troubleshoot'.

Select the webserver AZ1 then under image and templates in the actions dropdown click "create image".

EC2 > Instances > i-0d825dcc3a5d9e961 > Create image

Create image Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Instance ID
i-0d825dcc3a5d9e961 (Webserver AZ1)

Image name
ingepro AMI V2

Maximum 127 characters. Can't be modified after creation.

Image description - optional
ingepro AMI V2

Maximum 255 characters.

No reboot
 Enable

Instance volumes

Volume type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/...	Create new snapshot from...	8	EBS General Purpose S...	100		<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable

Add volume

Give your image a name and use the same name as description.

No reboot
 Enable

Instance volumes

Volume type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/...	Create new snapshot from...	8	EBS General Purpose S...	100		<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable

Add volume

① During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Tag image and snapshots together
Tag the image and the snapshots with the same tag.

Tag image and snapshots separately
Tag the image and the snapshots with different tags.

Key	Value - <small>optional</small>
<input type="text"/> Name	<input type="text"/> ingepro AMI V2

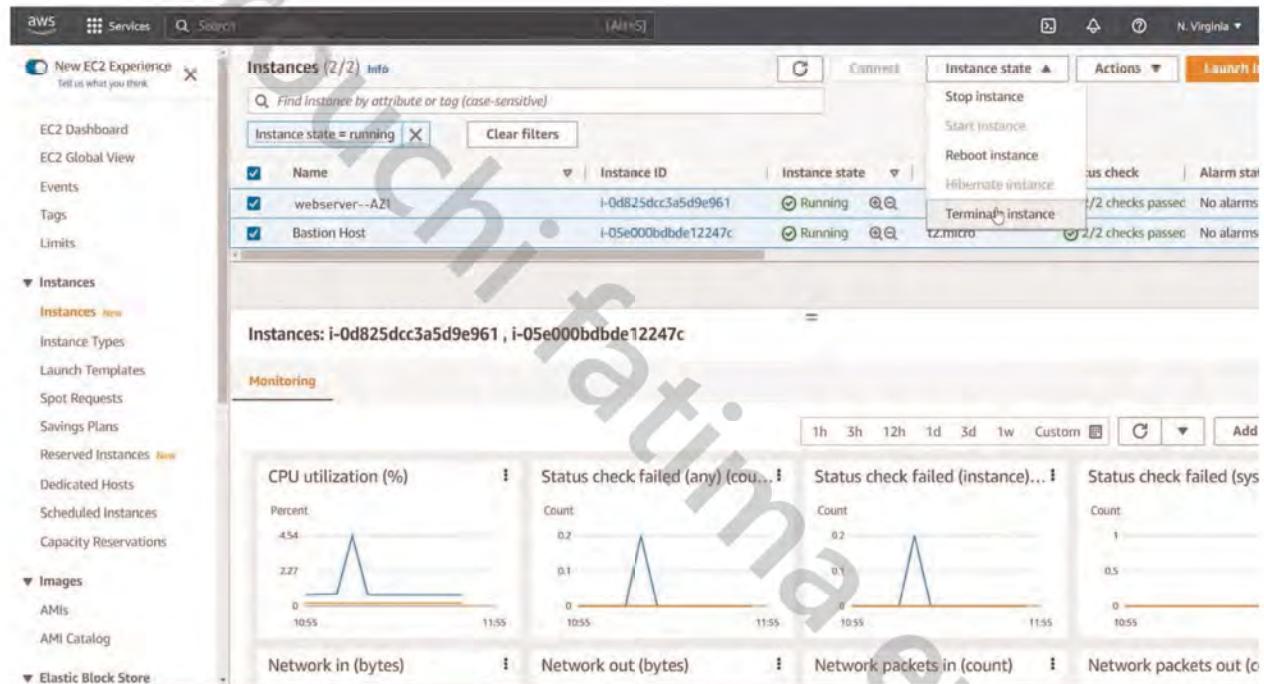
Add new tag

You can add up to 49 more tags.

Add a new tag enter name as key and the AMI name as a value then save changes.

→ Create an Auto-Scaling group:

To dynamically scale our webserver in the private app subnet and it ensures your application remains highly available and performs optimally by automatically adjusting the number of instances based on traffic demand.



First we have terminate the two instances we created manually to avoid conflicts, This ensures the Auto Scaling group works optimally without interference from manually created instances.

Go to instances in the EC2 dashboard, select the two running instances (bastion host & webserver—AZ1) under instance state click “Terminate instance” .

Now we should launch a Template since it defines the EC2 instance configuration used by the Auto Scaling group.

The screenshot shows the AWS EC2 Launch Templates dashboard. On the left, there's a navigation sidebar with options like EC2 Dashboard, EC2 Global View, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates (selected), Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, and Capacity Reservations. Below these are sections for Images, AMIs, and Elastic Block Store. The main content area has a title 'EC2 launch templates' and a subtitle 'Streamline, simplify and standardize instance launches'. It includes a description of launch templates and a 'Create launch template' button. A 'Benefits and features' section lists 'Streamline provisioning' and 'Simplify permissions'. To the right, there's a 'Documentation' section with links to 'Documentation' and 'API reference'.

In the same dashboard select Launch Templates then create launch template.

The screenshot shows the 'Create launch template' wizard. The top navigation bar shows 'EC2 > Launch templates > Create launch template'. The main form has a 'Launch template name and description' section where 'Dev-Launch-Template' is entered into the 'Launch template name - required' field. Below it, 'Launch template for ASG' is entered into the 'Template version description' field. There's also a note about 'Auto Scaling guidance' with an info icon and a checkbox for 'Provide guidance to help me set up a template that I can use with EC2 Auto Scaling'. At the bottom of this section are 'Template tags' and 'Source template' buttons. To the right, there's a 'Summary' section with tabs for 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', and 'Storage (volumes)'. A tooltip for the 'Free tier' is displayed, stating: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.' At the bottom right are 'Cancel' and 'Create launch template' buttons.

Give your launch template a name and description, make sure you checked the auto scaling guidance option.

Select this if you intend to use this template with EC2 Auto Scaling

Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

▶ Template tags

▶ Source template

Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

▼ Application and OS Images (Amazon Machine Image) - required Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recent AMIs My AMIs Quick Start

Owned by me Shared with me

Browse more AMIs

Amazon Machine Image (AMI)

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GiB of bandwidth to the internet.

Create launch template

▼ Application and OS Images (Amazon Machine Image) - required Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recent AMIs My AMIs Quick Start

Owned by me Shared with me

Browse more AMIs

Amazon Machine Image (AMI)

Ingepro AMI V2
ami-09f71ed25dfbc2966
2022-12-05T11:10:21.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

Description
ingepr0 AMI V2

Architecture x86_64 AMI ID ami-09f71ed25dfbc2966

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GiB of bandwidth to the internet.

Create launch template

Under my AMIs select Owned by me then choose the latest AMI we created.

The screenshots show the AWS Lambda console interface. The top screenshot is for a function named 'HelloWorld' and the bottom one is for 'MyFirstLambda'. Both show the configuration tabs and the execution history tab at the bottom.

Make sure t2.micro is select then choose your keypair, under Network settings check select existing sec group and choose the webserver security group, hit “create launch template”.

The screenshot shows the AWS Management Console with the 'Auto Scaling Groups' service selected. The main page features a large banner with the heading 'Amazon EC2 Auto Scaling helps maintain the availability of your applications'. Below the banner, there's a diagram titled 'How it works' illustrating the scaling process. The diagram shows an 'Auto Scaling group' containing four instances. One instance is highlighted with a dashed border, while the others are solid. Below the instances, a horizontal bar indicates the 'Desired capacity' with a midpoint labeled 'Minimum size' and an end point labeled 'Scale out as needed'.

Create Auto Scaling group

Get started with EC2 Auto Scaling by creating an Auto Scaling group.

Create Auto Scaling group

How it works

Pricing

Amazon EC2 Auto Scaling features have no additional fees beyond the service fees for Amazon EC2, CloudWatch (for scaling policies), and the other AWS resources that you use. Visit the pricing page of each service to learn more.

Getting started

What is Amazon EC2 Auto Scaling?

Name

Auto Scaling group name
Enter a name to identify the group.
Dev-ASG
Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info Switch to launch configuration

Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

Dev-Launch-Template

Create a launch template Info

Version

Default (1) C

Create a launch template version Info

Description	Launch template	Instance type
Launch template for ASG	Dev-Launch-Template <small>Info</small> lt-00c85ede7f74fb2fb	t2.micro
AMI ID	ami-09f71ed25dfbc2966	Security groups
Key pair name	-	Request Spot Instances
	-	No
	-	Security group IDs

Click "create auto scaling group" from auto scaling groups, give your ASG a name and choose the launch template we have previously created .

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.
vpc-047e879193d89f347 (dev-vpc)
10.0.0.0/16
Create a VPC

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.
Select Availability Zones and subnets
us-east-1a | subnet-0792629e1c7b0ea0d (Private--- App-Subnet-AZ1)
10.0.2.0/24
us-east-1b | subnet-0b7adde07d53d2897 (Private--- App-Subnet-AZ2)
10.0.3.0/24
Create a subnet

Instance type requirements Info
You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.
Override launch template

Launch template
Dev-Launch-Template
lt-00c85e4e7f74f82fb

Version
Default

Description
Launch template for ASG

Load balancing - optional Info
Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer
Choose from your existing load balancers.

Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer
Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.
Select target groups
dev-tg | HTTP
Application Load Balancer: Dev-ALB

Choose from Classic Load Balancers

Health checks - optional

Select the dev—vpc and the private app subnets in AZ1 and AZ2, check attach to an existing LB and choose the dev—tg .

The screenshot shows the AWS Auto Scaling configuration interface. The top navigation bar includes 'Services' and a search bar. The main content area is titled 'Configure group size and scaling policies'. On the left, a sidebar lists steps: Step 1 (Choose launch template or configuration), Step 2 (Choose instance launch options), Step 3 (optional) Configure advanced options, Step 4 (optional) Configure group size and scaling policies (which is currently selected), Step 5 (optional) Add notifications, Step 6 (optional) Add tags, and Step 7 (Review). The main panel has two sections: 'Health checks - optional' and 'Additional settings - optional'. In 'Health checks - optional', 'EC2' is selected as the health check type. In 'Additional settings - optional', there are sections for 'Monitoring' and 'Default instance warmup'. At the bottom are 'Cancel', 'Previous', 'Skip to review', and 'Next' buttons.

This screenshot continues from the previous one, showing the 'Configure group size and scaling policies' step. The sidebar remains the same. The main panel now focuses on 'Group size - optional'. It specifies a 'Desired capacity' of 2, a 'Minimum capacity' of 1, and a 'Maximum capacity' of 4. Below this is a 'Scaling policies - optional' section, which currently has 'None' selected. The 'Info' link for scaling policies is visible.

In the group size type 2 as desired capacity and set the minimum to 1 and maximum to 4, to allow the number of running instances to adjust based on traffic.

The screenshot shows the 'Add notifications' step of an AWS Auto Scaling configuration wizard. On the left, a sidebar lists steps from 1 to 7. Step 5 is currently selected, 'Add notifications'. The main area displays a 'Notification 1' configuration. It includes fields for 'Send a notification to' (set to 'my-sns-topic') and 'With these recipients' (set to 'chakir.laimina@ingepromaroc.ma'). There is also a 'Use existing topic' button. Under 'Event types', four checkboxes are checked: 'Launch', 'Terminate', 'Fail to launch', and 'Fail to terminate'. A 'Next' button is at the bottom right.

The screenshot shows the 'Add tags' step of the AWS Auto Scaling configuration wizard. The sidebar shows steps 1 through 6, with step 6, 'Add tags', selected. The main area shows a 'Tags (1)' section. A single tag is listed with 'Key' set to 'Name' and 'Value - optional' set to 'ASG-Webserver'. A checkbox labeled 'Tag new instances' is checked. A note at the top right says: 'You can optionally choose to add tags to instances (and their attached EBS volumes) by specifying tags in your launch template. We recommend caution, however, because the tag values for instances from your launch template will be overridden if there are any duplicate keys specified for the Auto Scaling group.' A 'Next' button is at the bottom right.

Add the email where you want to receive notifications and finally give you tag a name and value then save changes.

The screenshot shows the AWS CloudWatch Metrics Targets console. On the left, there's a sidebar with navigation links for Images, Elastic Block Store, Network & Security, Load Balancing, and Auto Scaling. The main area displays target configuration details: Target type (Instance), Protocol / Port (HTTP: 80), Protocol version (HTTP1), VPC (vpc-047e879193d89f347), and a summary of targets (Total targets: 2, Healthy: 2, Unhealthy: 0, Unused: 0, Initial: 0). Below this, tabs for Targets, Monitoring, Health checks, Attributes, and Tags are visible. The Targets tab is selected, showing a table titled "Registered targets (2)". The table includes columns for Instance ID, Name, Port, Zone, Health status, and Health status detail. Two instances are listed: i-035fc0502ee59e5a3 (ASG-Webserver, port 80, us-east-1b, healthy) and i-0b7864b515b8f5b64 (ASG-Webserver, port 80, us-east-1a, healthy).

As you can see we have successfully created an auto scaling group, now we can access our secured website.

The screenshot shows a web browser window displaying the website ingepromaroc.ma. A security certificate dialog box is open, stating "Connection is secure" and "Your information (for example, passwords or credit card numbers) is private when it is sent to this site." It also mentions "Certificate is valid". The main website page has a dark orange header with the text "Crafting project specific solutions with expertise." and "We're a creative company that focuses on establishing long-term relationships with customers." It features two buttons: "Explore Now" and "Contact Us". The background of the website shows a large industrial port with cranes and shipping containers. The browser taskbar at the bottom shows various pinned icons and the date/time as 8/24/2024 3:25 PM.

Chapter 3

→Tools and softwares used:

➤ Front-end:

• Html:

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It defines the content and structure of web content.



• Css:

Cascading Style Sheets (CSS) is a style sheet language used for specifying the presentation and styling of a document written in a markup language such as HTML or XML (including XML dialects such as SVG, MathML or XHTML).



• Javascript:

JavaScript is a high-level programming language that follows the ECMAScript standard. It was originally designed as a scripting language for websites but became widely adopted as a general-purpose programming language.



• Bootstrap:

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.



➤ **Back-end:**

- **Php:**

PHP is a general-purpose scripting language geared towards web development. It was originally created by Danish-Canadian programmer Rasmus Lerdorf in 1993 and released in 1995. The PHP reference implementation is now produced by the PHP Group. PHP was originally an abbreviation of Personal Home Page, but it now stands for the recursive initialism PHP: Hypertext Preprocessor.



- **Mysql:**

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter My, and "SQL", the acronym for Structured Query Language.



➤ Services:

- **PHPmyadmin:**

phpMyAdmin is a free and open source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it has become one of the most popular MySQL administration tools, especially for web hosting services.



- **AWS:**

Amazon Web Services, Inc. (AWS) is a subsidiary of Amazon that provides on-demand cloud computing platforms and APIs to individuals, companies, and governments, on a metered, pay-as-you-go basis. Clients will often use this in combination with autoscaling (a process that allows a client to use more computing in times of high application usage, and then scale down to reduce costs when there is less traffic).



- **EC2:**

Amazon Elastic Compute Cloud (EC2) is a part of Amazon.com's cloud-computing platform, Amazon Web Services (AWS), that allows users to rent virtual computers on which to run their own computer applications. EC2 encourages scalable deployment of applications by providing a web service through which a user can boot an Amazon Machine Image (AMI) to configure a virtual machine, which Amazon calls an "instance", containing any software desired.



- **Route53:**

Amazon Route 53 (Route 53) is a scalable and highly available Domain Name System (DNS) service. Released on 5 December 2010 (13 years ago), it is part of Amazon.com's cloud computing platform, Amazon Web Services (AWS). The name is a possible reference to U.S. Routes, and "53" is a reference to the TCP/UDP port 53, where DNS server requests are addressed.

In addition to being able to route users to various AWS services, including EC2 instances, Route 53 also enables AWS customers to route users to non-AWS infrastructure and to monitor the health of their application and its endpoints.



- **S3:**

Amazon S3 (Amazon Simple Storage Service) is a file hosting site offered by Amazon Web Services. Amazon S3 offers storage services through web services (REST, SOAP, and BitTorrent).



- **VPC :**

A virtual private cloud (VPC) is an on-demand configurable pool of shared resources allocated within a public cloud environment, providing a certain level of isolation between the different organizations (denoted as users hereafter) using the resources.



- **Nat gateway:**

A NAT gateway is a Network Address Translation (NAT) service. You can use a NAT gateway so that instances in a private subnet can connect to services outside your VPC but external services cannot initiate a connection with those instances.



- **Security Groups:**

A security group controls the traffic that is allowed to reach and leave the resources that it is associated with. For example, after you associate a security group with an EC2 instance, it controls the inbound and outbound traffic for the instance.

- **RDS:**

Amazon Relational Database Service (Amazon RDS) est un service de base de données relationnelle facile à gérer, optimisé pour le coût total de possession. Il est simple à configurer, à utiliser et se met à l'échelle à la demande.



- **IAM :**

Identity and Access Management (IAM) is a foundational control of cloud security as it authenticates users and regulates access to systems, networks and data. Cloud identity manager grants users' access and entitlements across a wide range of cloud and on-premises applications and services.



- **ALB:**

Application Load Balancer operates at the request level (layer 7), routing traffic to targets (EC2 instances, containers, IP addresses, and Lambda functions) based on the content of the request. Ideal for advanced load balancing of HTTP and HTTPS traffic, Application Load Balancer provides advanced request routing targeted at delivery of modern application architectures, including microservices and container-based applications. Application Load Balancer simplifies and improves the security of your application, by ensuring that the latest SSL/TLS ciphers and protocols are used at all times.



- **AWS Certificate Manager (ACM):**

AWS Certificate Manager (ACM) is a service that lets you easily provision, manage, and deploy public and private Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates for use with AWS services and your internal connected resources.



- **Auto Scaling group:**

An Auto Scaling group contains a collection of EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management. An Auto Scaling group also lets you use Amazon EC2 Auto Scaling features such as health check replacements and scaling policies.



➤ Softwares:

- **Vs code:**

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor developed by Microsoft for Windows, Linux, macOS and web browsers. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded version control with Git.



- **XAMPP:**

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible.



- **Windows PowerShell:**

Windows PowerShell is a task automation and configuration management framework from Microsoft, consisting of a command-line shell and the associated scripting language. It is designed to automate the administration of various system tasks, such as batch processing, and to provide a more flexible and powerful interface than the traditional Windows Command Prompt.



- **MySQL workbench:**

MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system. It is the successor to DBDesigner 4 from fabFORCE.net, and replaces the previous package of software, MySQL GUI Tools Bundle.



Chapter 4

→Conclusion:

The completion of this internship project has been a significant milestone in both personal and professional development. By successfully creating and deploying a dynamic website for IngePro Maroc, I was able to apply and expand my knowledge of web development and cloud computing. The project not only achieved its goals of improving the company's digital footprint but also demonstrated the practical applications of theoretical knowledge gained during my studies. This experience has equipped me with a comprehensive understanding of full-stack development and cloud hosting, skills that will undoubtedly be valuable in my future career endeavors. The positive impact on IngePro Maroc underscores the importance of digital transformation in the engineering sector, and I am proud to have contributed to this initiative.