

# Raport Projektu Chmury 1

Mateusz Staszków, #263949, IST PSI

## 1. Wprowadzenie

Celem projektu było stworzenie aplikacji webowej (np. prostego chatu grupowego) z podziałem na moduły backend oraz frontend, które są hostowane niezależnie. Realizacja projektu obejmowała:

- Implementację co najmniej 2 endpointów GET oraz 2 endpointów POST (w tym jeden do przesyłania plików multimedialnych).
- Dockerizację obu modułów oraz konfigurację Dockera.
- Synchronizację aplikacji z usługami AWS, w tym Elastic Beanstalk, RDS, S3, CloudWatch, Cognito oraz Lambda.
- Automatyzację wdrożenia infrastruktury przy użyciu Terraform.

## 2. Opis aplikacji

Aplikacja została pierwotnie przetestowana lokalnie (na localhost) z wykorzystaniem kontenerów Docker, a następnie zsynchronizowana z AWS. Główne funkcjonalności aplikacji to:

- Rejestracja i uwierzytelnianie użytkowników z wykorzystaniem AWS Cognito (wraz z automatycznym potwierdzaniem kont przy użyciu funkcji Lambda).
- Możliwość wysyłania wiadomości między użytkownikami oraz przesyłania plików multimedialnych.
- Prosty interfejs umożliwiający pobieranie przesłanych plików oraz podgląd wiadomości zarówno wysłanych, jak i odebranych

Kod całego projektu znajduje się pod linkiem: <https://github.com/stvshy/chat-app-aws/tree/using-terraform>

Repozytorium zostało podzielone na dwa branche: master (konfiguracja manualna) oraz using-terraform (konfiguracja z terraformem)

## 3. Implementacja i konfiguracja

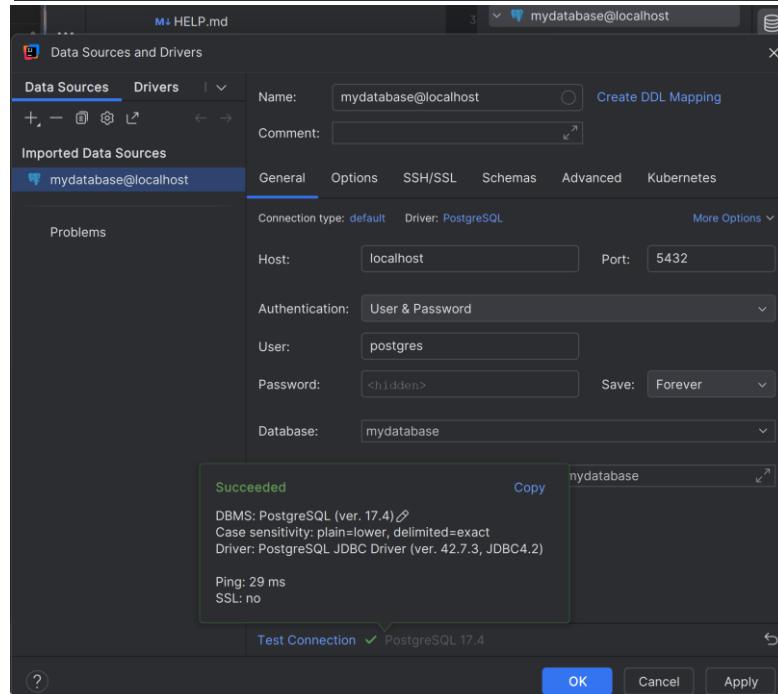
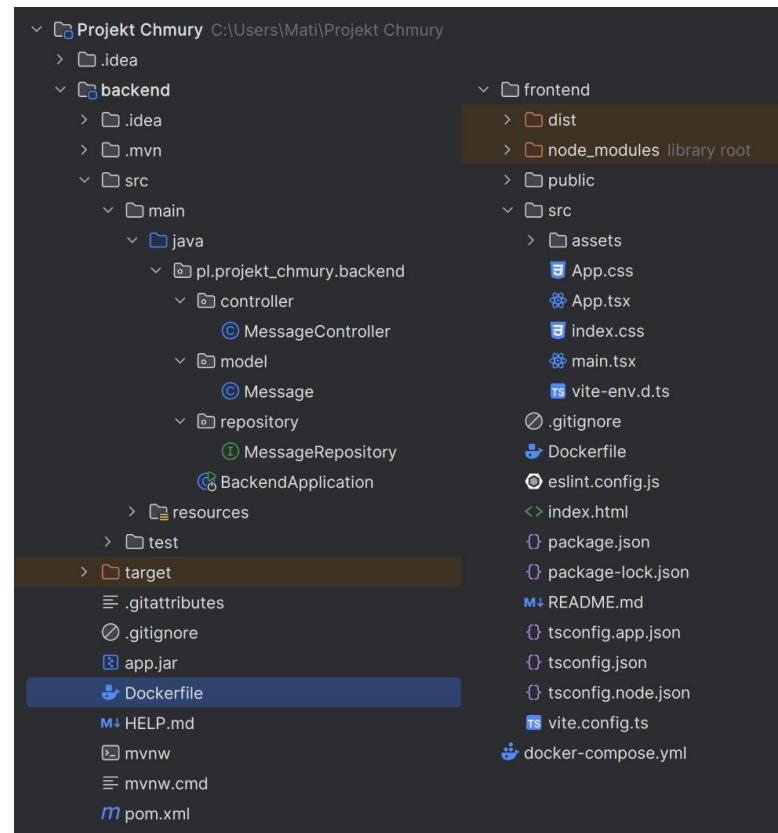
### 3.1 Dockerizacja

- Wszystkie moduły (frontend i backend) zostały zdockerowane, co umożliwiło uruchomienie aplikacji lokalnie przed wdrożeniem na AWS.

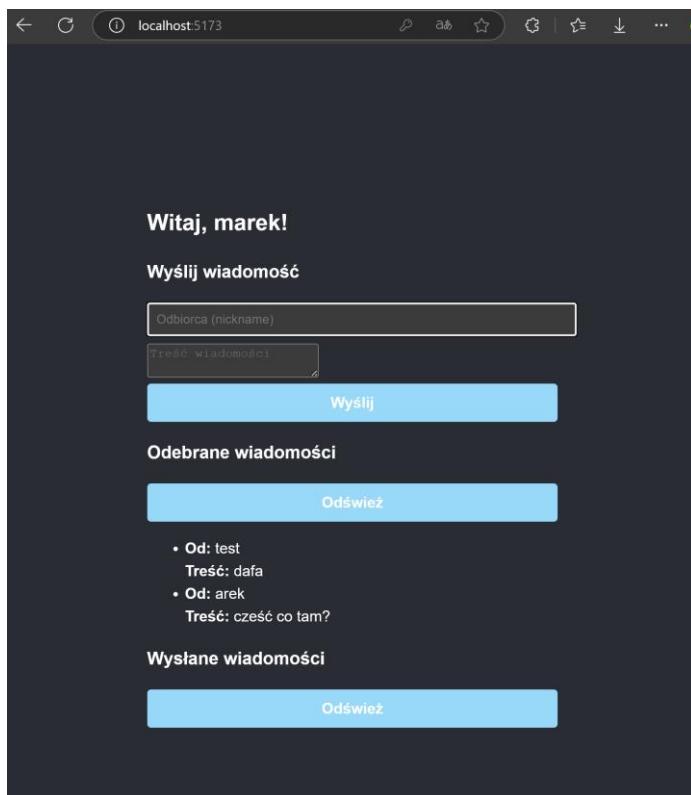
```
PS C:\Users\Mati\Projekt Chmury> docker-compose up -d
time="2025-03-18T11:56:38+01:00" level=warning msg="C:\\\\Users\\\\Mati\\\\Projekt Chmury\\\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/3
  ✓ Container postgres_container  Started                         1.5s
  ✓ Container backend-container   Started                         1.8s
  ✓ Container frontend-container Started                         0.8s
```

CONTAINER ID	IMAGE NAMES	COMMAND	CREATED	STATUS	PORTS
6ad59f74eb92	projektchmury-backend backend-container	"java -jar app.jar"	25 seconds ago	Up 24 seconds	0.0.0.0:8081->8080/tcp
4e05b1c359cb	postgres:latest postgres_container	"docker-entrypoint.s..."	25 seconds ago	Up 25 seconds	0.0.0.0:5432->5432/tcp
4fde22e07aa0	projektchmury-frontend frontend-container	"docker-entrypoint.s..."	58 seconds ago	Up 25 seconds	0.0.0.0:5173->5173/tcp

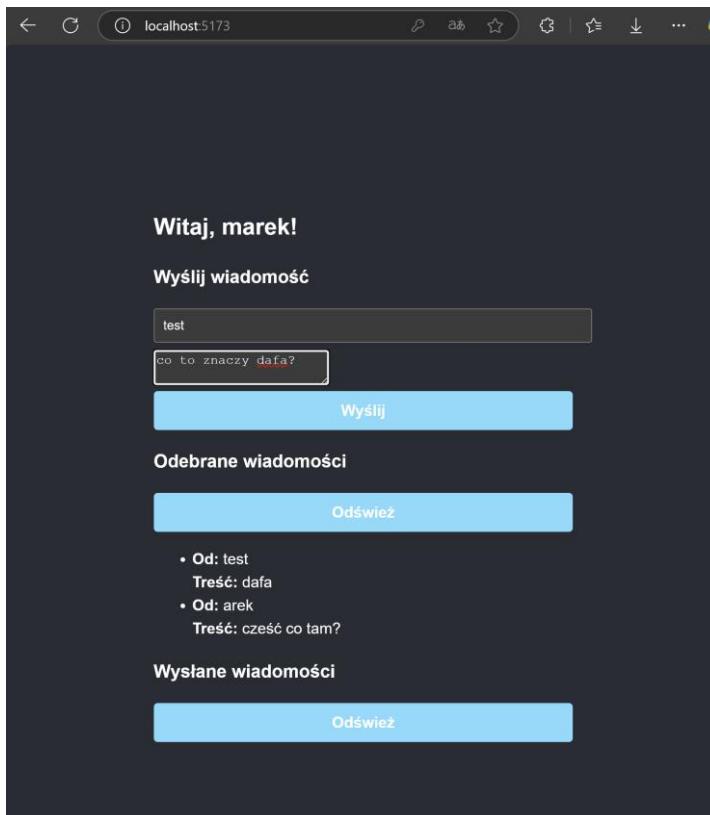
- Po potwierdzeniu poprawności działania na localhost, przystąpiono do synchronizacji z chmurą. Tak na tamten moment wyglądała struktura projektu:

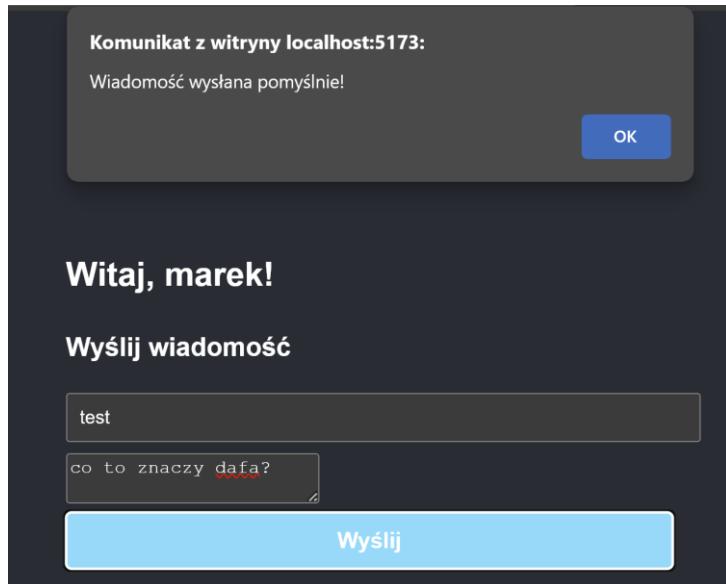


- Tak natomiast wyglądało na tamten moment UI:



- Poprawne działanie na localhoście:





**Witaj, marek!**

**Wyślij wiadomość**

test

co to znaczy dafa?

Wyślij

**Witaj, marek!**

**Wyślij wiadomość**

Odbiorca (nickname)

Treść wiadomości

Wyślij

**Odebrane wiadomości**

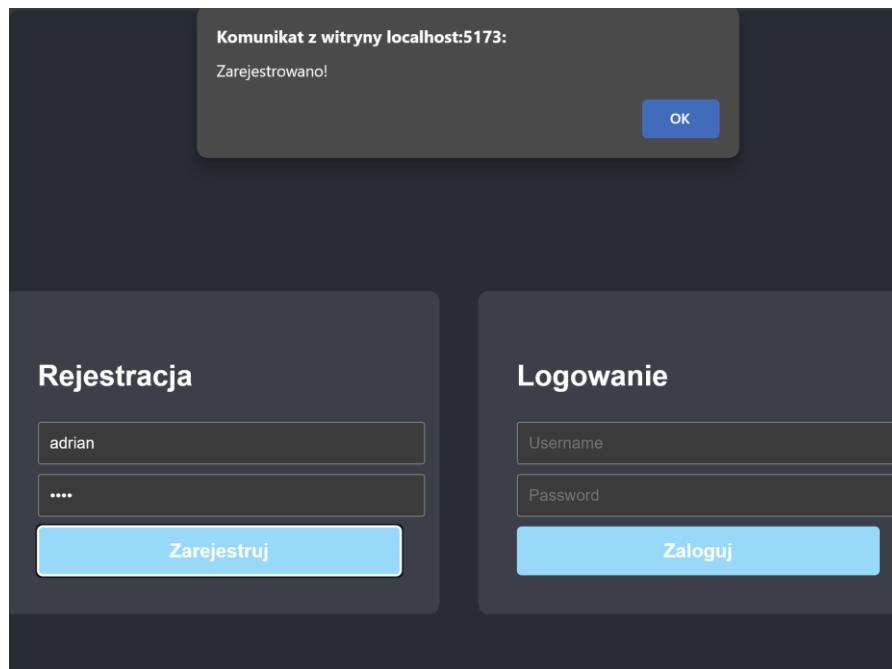
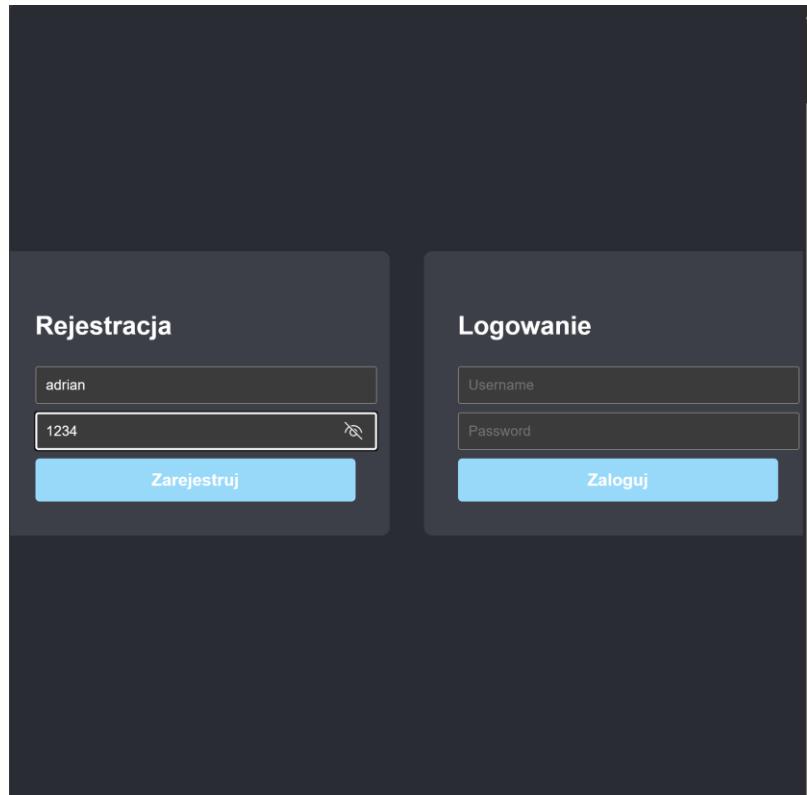
Odśwież

- Od: test  
Treść: dafa
- Od: arek  
Treść: cześć co tam?

**Wysłane wiadomości**

Odśwież

- Do: test  
Treść: co to znaczy dafa?



The screenshot shows a database interface with two tables displayed:

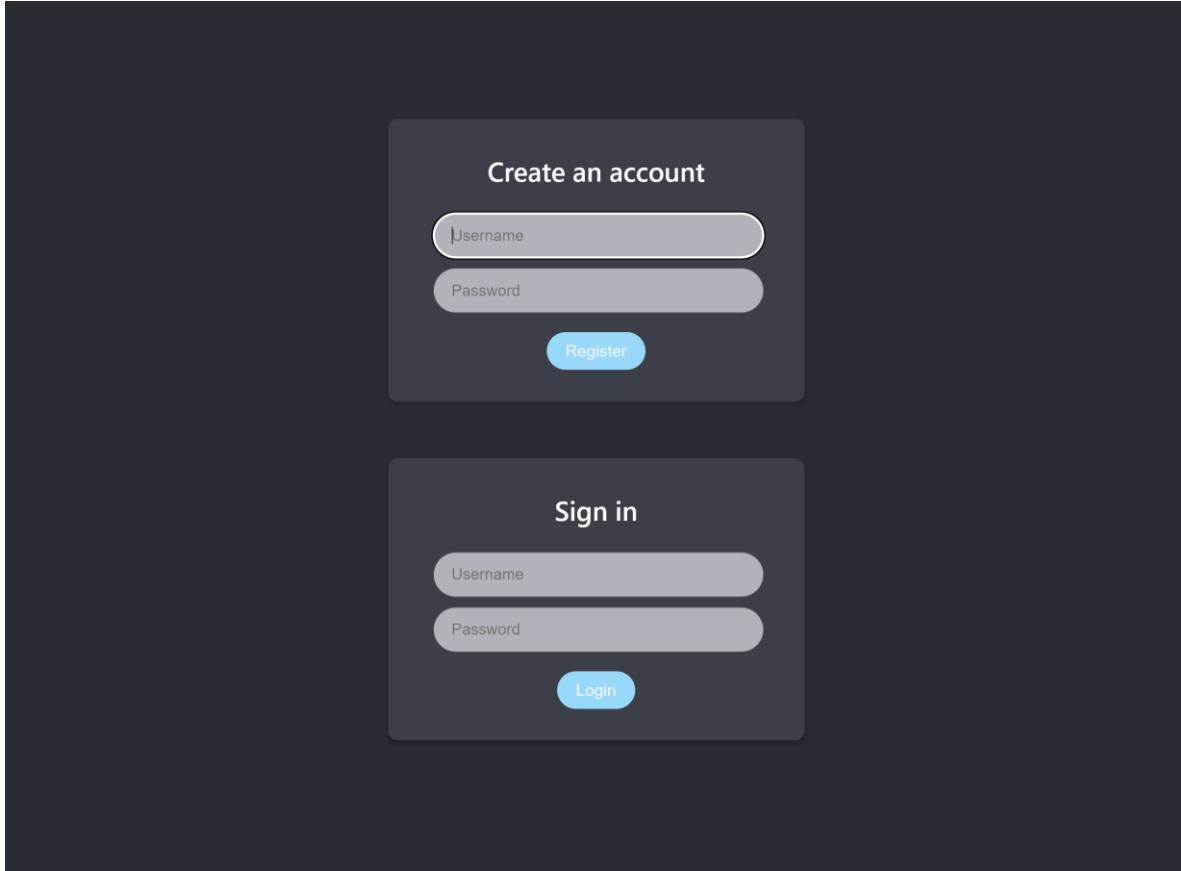
### users

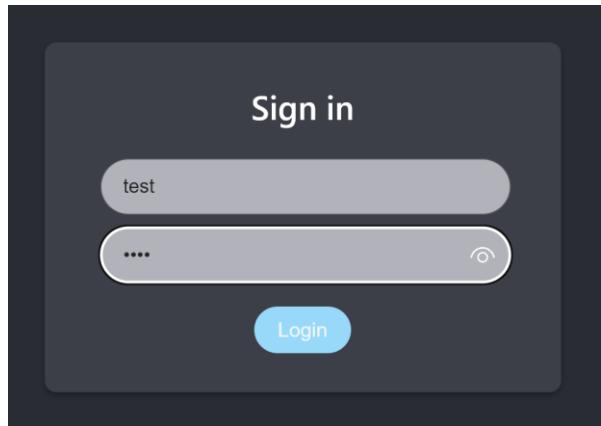
	id	password	username
1	1	\$2a\$10\$FhLljdusao...	marek
2	2	\$2a\$10\$ILDG2epHAp...	test
3	52	\$2a\$10\$1My.L1DCYG...	arek
4	53	\$2a\$10\$agoIRgV00F...	adrian

### message

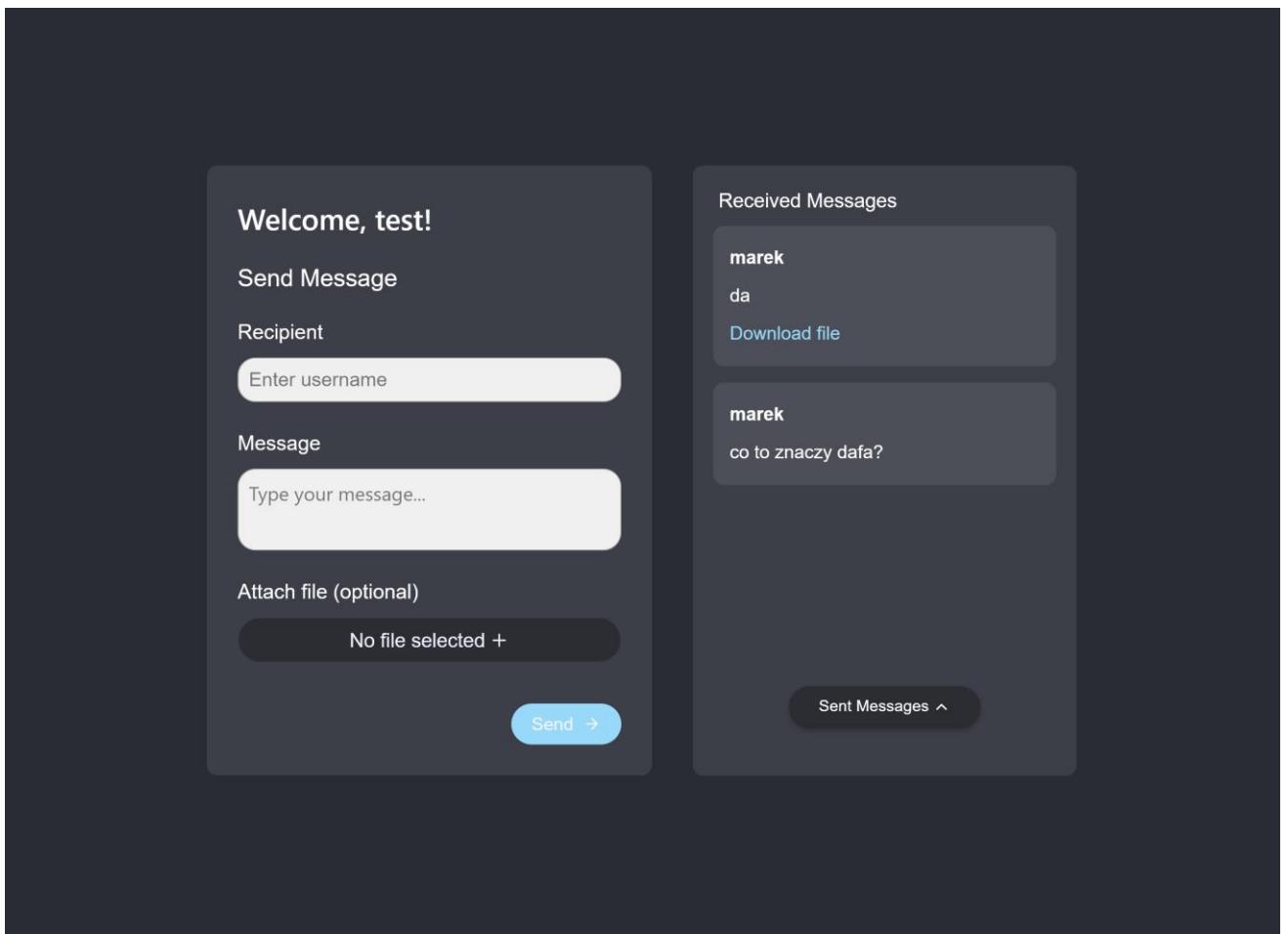
	id	content	author_id	recipient_id
1	1	dafa		2
2	2	cześć co tam?		52
3	3	co to znaczy dafa?		1

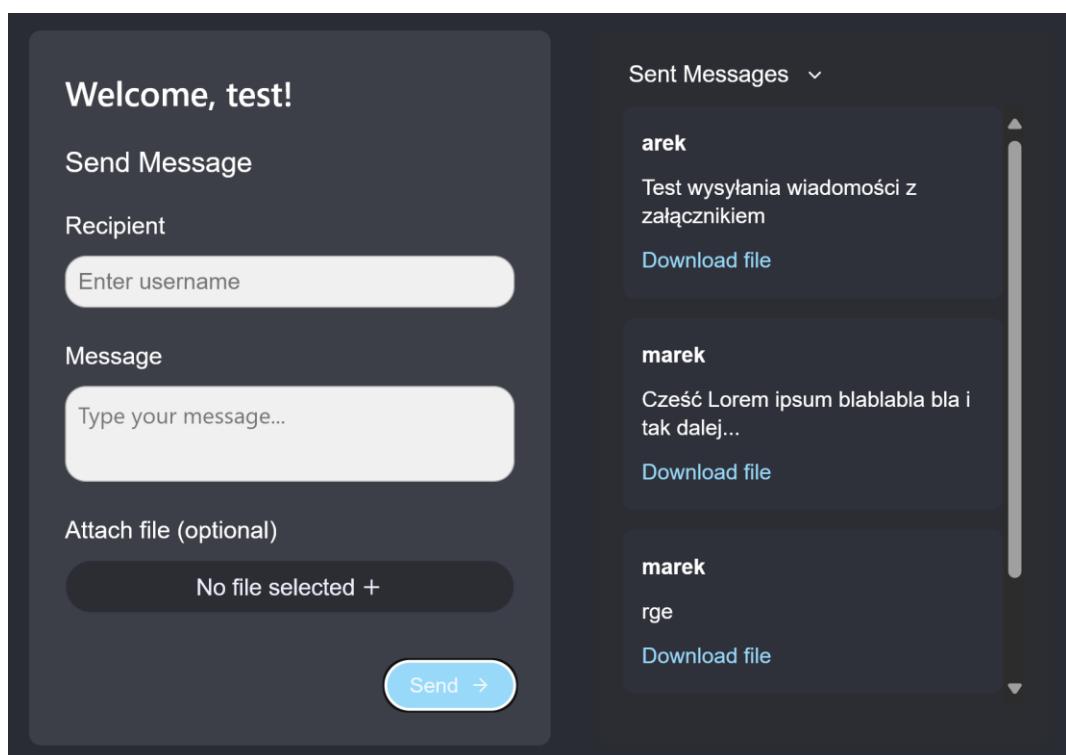
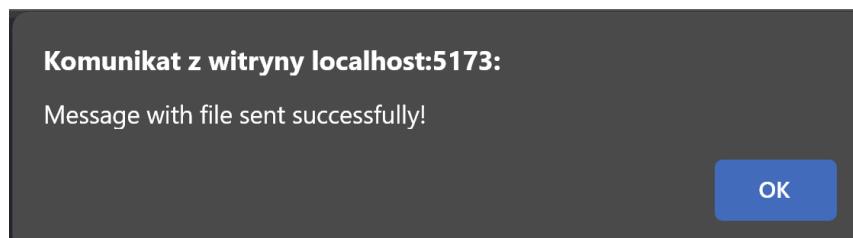
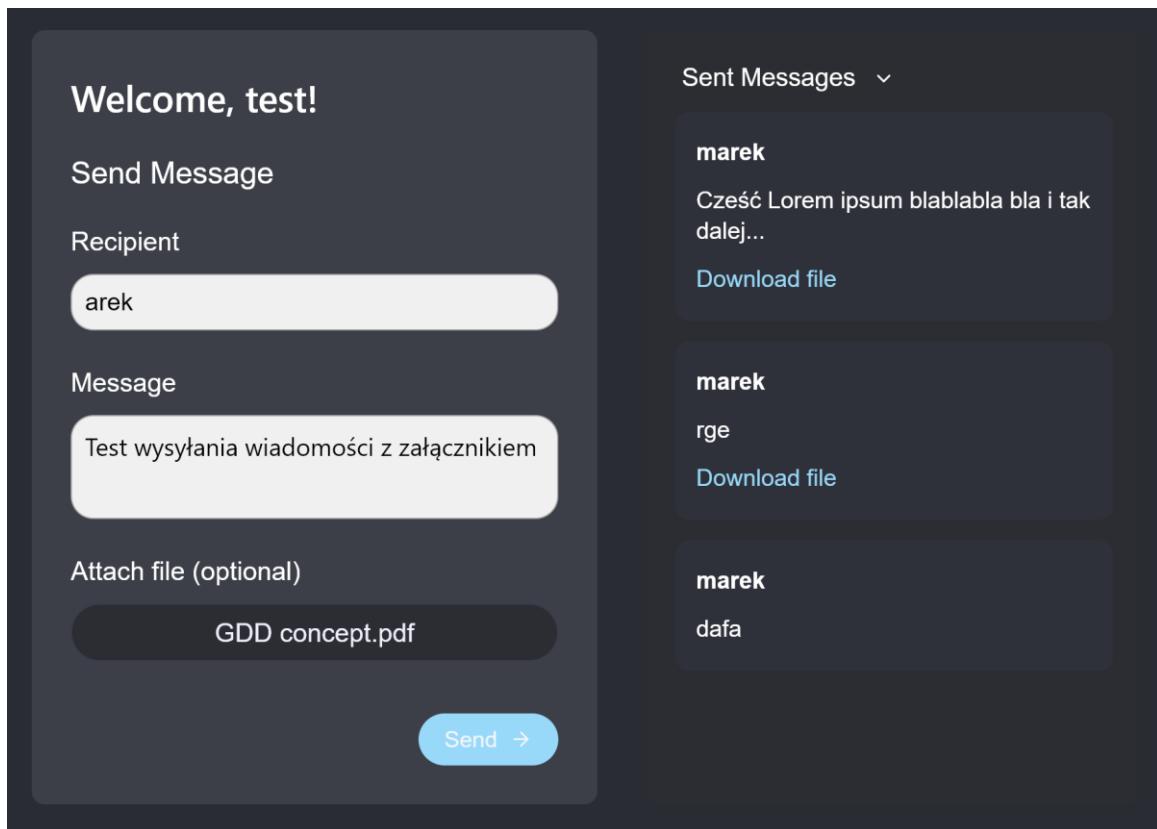
- Nowe UI logowania:



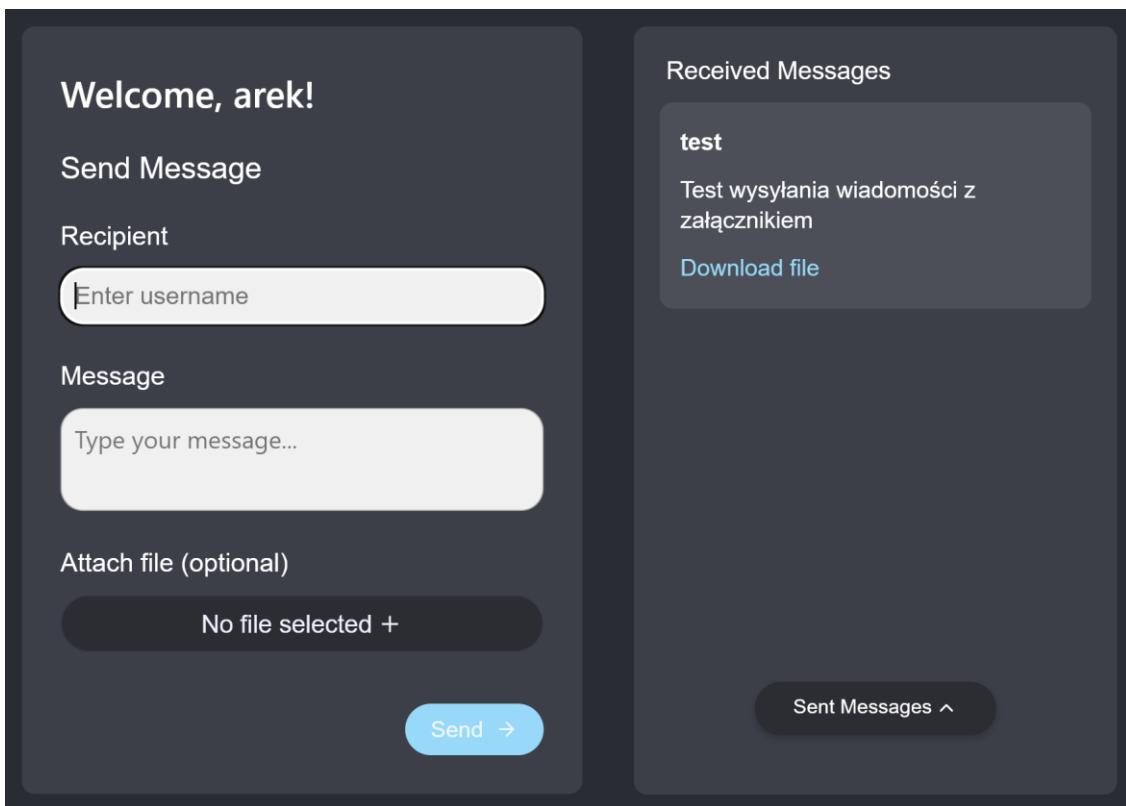


- Nowy widok czatu:

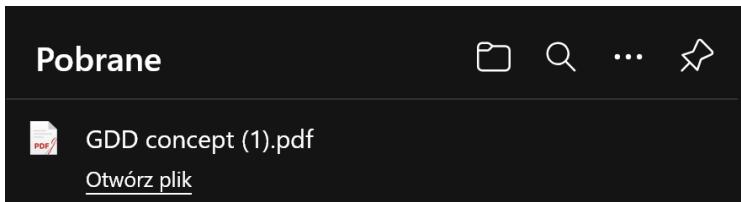




Teraz sprawdzamy u arka



Po kliknięciu na przycisk download file:



### 3.2 Konfiguracja AWS przez interfejs webowy

Najpierw wykonana została przeze mnie manualna konfiguracja poprzez konsolę AWS. W raporcie znajdują się zrzuty ekranu przedstawiające między innymi:

- Konfigurację backendu z CloudWatch.
- Integrację frontendu z CloudWatch.
- Konfiguracje wszystkich wymaganych serwisów AWS
- Testowe logowanie oraz przesyłanie wiadomości z załącznikami.
- Weryfikację poprawności działania aplikacji (np. pobieranie plików, wyświetlanie wiadomości w różnych kontach użytkowników).

- Backend w Elastic Beanstalk:

**Environments (3) [Info](#)**

Environment name	Health	Application	Platform	Domain	Running version	Tier name	Date created	Last modified
backend-env	Suspended	projektch...	Docker ru...	projektchmury-backend.us-eas...	1.0.2	WebServer	March 23, 2025	March 24, 2025
projektchmury-backend-env	Ok	projektch...	Docker ru...	projekt-chmury-backend.us-ea...	1.0.24	WebServer	March 23, 2025	April 4, 2025
projektchmury-frontend-env	Ok	projektch...	Docker ru...	projektchmury-frontend.us-ea...	1.0.15	WebServer	March 24, 2025	April 4, 2025

**projektchmury-backend-env [Info](#)**

**Environment overview**

Health Ok	Environment ID e-ymrckaxmj
Domain projekt-chmury-backend.us-east-1.elasticbeanstalk.com	Application name projektchmury-backend

**Platform**

Platform Docker running on 64bit Amazon Linux 2/4.1.0	Change version
Running version 1.0.24	Platform state Supported

**Events** | **Health** | **Logs** | **Monitoring** | **Alarms** | **Managed updates** | **Tags**

**Events (100) [Info](#)**

Time	Type	Details
April 4, 2025 18:29:40 (UTC+2)	INFO	Environment health has transitioned from No Data to Ok.
April 4, 2025 18:28:40 (UTC+2)	INFO	Added instance [i-037f078128e8a7ded] to your environment.
April 4, 2025 18:28:40 (UTC+2)	INFO	Environment health has transitioned from Ok to No Data. 1 instance online which meets Auto Scaling group desired capacity of 1. All instances are in same availability zone (us-east-1c).
April 4, 2025 18:28:40 (UTC+2)	INFO	Removed instance [i-0ea0cc976c9a7053f] from your environment.
April 4, 2025 17:21:45 (UTC+2)	INFO	Environment health has transitioned from Info to Ok.
April 4, 2025 17:20:11 (UTC+2)	INFO	Environment update completed successfully.
April 4, 2025 17:20:11 (UTC+2)	INFO	Successfully deployed new configuration to environment.
April 4, 2025 17:20:11 (UTC+2)	INFO	New application version was deployed to running EC2 instances.

**Configuration [Info](#)**

**Service access [Info](#)**

Service role arn:aws:iam::107378568397:role/LabRole	EC2 key pair vokey	EC2 Instance profile LabInstanceProfile
--	-----------------------	--

**Networking and database [Info](#)**

Network	Public IP address true	Instance subnets subnet-03e74672cba510707
---------	---------------------------	--

**Instance traffic and scaling [Info](#)**

Instances	EC2 Security Groups sg-0339eb242986f6986,sg-082624d59e6c0396d,sg-0fbff690ca8991550	On-demand base 0
Capacity	Fleet composition On-Demand instance	Scaling cooldown 360
Processor type	Capacity rebalancing Deactivated	AMI ID ami-0f78189a679d18874
x86_64	Instance types t3.micro, t3.small	

**Updates, monitoring, and logging** [Info](#)

Define when and how Elastic Beanstalk deploys changes to your environment. Manage your application's monitoring and logging settings, instances, and other environment resources.

**Monitoring**

Log group	System	Cloudwatch custom metrics - instance
[Alt+S]	United States (N. Virginia)  vclabs/user3928724=Matheusz_Staszek_w @ 1073-7856-859	

[projektchmury-backend-env](#) > Configuration

Activated 7

**Lifecycle**  
false

**Updates**

Managed updates	Deployment batch size	Deployment batch size type
Activated	100	Percentage

Command timeout	Deployment policy	Health threshold
600	AllAtOnce	Ok

Ignore health check	Instance replacement
false	false

**Platform software**

Lifecycle	Log groups	Log streaming
false	/aws/elasticbeanstalk/projektchmury-backend-env	Activated

Proxy server	Logs retention	Rotate logs
none	7	Deactivated

Update level	X-Ray enabled
minor	Deactivated

**Environment properties**

Source	Key	Value
Plain text	SPRING_DATASOURCE_PASSWORD	admin1234
Plain text	SPRING_DATASOURCE_URL	jdbc:postgresql://baza-chmury.cdle9v0rnyed.us-east-1.rds.amazonaws.com:5432/baza-chmury?sslmode=require
Plain text	SPRING_DATASOURCE_USERNAME	postgres

#### Instance metadata service (IMDS)

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

##### IMDSv1

With the current setting, the environment enables only IMDSv2.

Deactivated

#### EC2 security groups

Select security groups to control traffic.

**EC2 security groups (6)**

<input type="text"/> Filter security groups	Group name	Group ID	Name
<input checked="" type="checkbox"/> awseb-e-muct4ccfvm-stack-AWSEBSecurityGroup-iUjAJGVd7mNK	sg-082624d59e6c0396d		backend-env
<input type="checkbox"/> awseb-e-qp85mykyic-stack-AWSEBSecurityGroup-HN1oB1NMSpId	sg-07c2ccb59a910cf09		projektchmury-frontend-env
<input checked="" type="checkbox"/> awseb-e-ymrccrakxmj-stack-AWSEBSecurityGroup-QSUoebzqA2Ml	sg-0fbff690ca8991550		projektchmury-backend-env
<input checked="" type="checkbox"/> default	sg-0339eb242986f6986		
<input type="checkbox"/> ec2-rds-1	sg-082f9d6e474396f0c		
<input type="checkbox"/> rds-ec2-1	sg-01897eb9d86089958		

## Konfiguracja backendu z CloudWatch:

▼ **Monitoring** Info

**Health reporting**

Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The **EnvironmentHealth** custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#)

**System**

Basic  
 Enhanced

**CloudWatch Custom Metrics - Instance**

**CloudWatch Custom Metrics - Environment**

**Health event streaming to CloudWatch Logs**

Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

**Log streaming**

Activated (standard CloudWatch charges apply.)

Log group: [/aws/elasticbeanstalk/projektmury-backend-env/environment-health.log](#)

**Retention**

7

**Lifecycle**

Keep logs after terminating environment

**Instance log streaming to CloudWatch logs**

Configure the instances in your environment to stream logs to CloudWatch logs. You can set the retention up to 10 years and configure Elastic Beanstalk to delete the logs when you terminate your environment. [Learn more](#)

Log groups: [/aws/elasticbeanstalk/projektmury-backend-env](#)

**Log streaming**  
(standard CloudWatch charges apply.)

Activated

**Retention**

7

**Lifecycle**

Keep logs after terminating ...

▼ **Managed platform updates** Info

Activate managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.

**Managed updates**

Activated

**Managed actions role**

arn:aws:iam::107378568397:role/LabRole

Weekly update window

Thursday at 23 : 19 UTC

**Update level**

Minor and patch

## Frontend w Elastic Beanstalk:

**projektmury-frontend-env** [Info](#)

[Actions](#) [Upload and deploy](#)

**Environment overview**

Health	<span>Ok</span>
Domain	projektmury-frontend.us-east-1.elasticbeanstalk.com
Environment ID	e-qpb5mykyic
Application name	projektmury-frontend

**Platform**

Platform	Docker running on 64bit Amazon Linux 2/4.1.0
Running version	1.0.15
Platform state	<span>Supported</span>

[Events](#) [Health](#) [Logs](#) [Monitoring](#) [Alarms](#) [Managed updates](#) [Tags](#)

**Events (100)** [Info](#)

Time	Type	Details
April 5, 2025 15:53:49 (UTC+2)	INFO	Environment health has transitioned from Info to Ok.
April 5, 2025 15:52:58 (UTC+2)	INFO	Environment update completed successfully.
April 5, 2025 15:52:58 (UTC+2)	INFO	Successfully deployed new configuration to environment.
April 5, 2025 15:52:49 (UTC+2)	INFO	Environment health has transitioned from Ok to Info. Command is executing on all instances.
April 5, 2025 15:52:20 (UTC+2)	INFO	Instance deployment completed successfully.
April 5, 2025 15:51:44 (UTC+2)	INFO	Updating environment projektmury-frontend-env's configuration settings.
April 5, 2025 15:51:43 (UTC+2)	INFO	Created Amazon CloudWatch log group named: /aws/elasticbeanstalk/projektmury-frontend-env/var/log/eb-hooks.log
April 5, 2025 15:51:42 (UTC+2)	INFO	Created Amazon CloudWatch log group named: /aws/elasticbeanstalk/projektmury-frontend-env/var/log/eb-logs-stdout

**Service access** [Info](#)

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

Service role	<a href="#">arn:aws:iam::107378568397:role/LabRole</a>	EC2 key pair	vockey	EC2 instance profile	<a href="#">Edit</a>
				LabInstanceProfile	

**Networking and database** [Info](#)

Configure VPC settings, and subnets for your environment's EC2 instances and load balancer. Set up an Amazon RDS database that's integrated with your environment.

Network	<a href="#">Edit</a>				
VPC	true	Public IP address	true	Instance subnets	subnet-03e74672cba510707
arn:aws:vpc:08e2f7b9bd2c1ab27					

**Instance traffic and scaling** [Info](#)

Customize the capacity and scaling for your environment's instances. Select security groups to control instance traffic. Configure the software that runs on your environment's instances by setting platform-specific options.

Instances	<a href="#">Edit</a>				
IMDSv1	Deactivated	EC2 Security Groups	sg-0339eb242986f6986,sg-082624d59e6c0396d,sg-0fbff690ca891550,sg-07c2cc59a910cf9		
Environment type	Single instance	Fleet composition	On-Demand instance	On-demand base	0
On-demand above base	0	Capacity rebalancing	Deactivated	Scaling cooldown	360
Processor type	x86_64	Instance types	t3.micro, t3.small	AMI ID	ami-0f78189a679d18874

**Updates, monitoring, and logging** [Info](#)

Define when and how Elastic Beanstalk deploys changes to your environment. Manage your application's monitoring and logging settings, instances, and other environment resources.

**Monitoring**

<b>Log group</b> <a href="#">/aws/elasticbeanstalk/projektchmury-frontend-env/environment-health.log</a>	<b>System</b> enhanced	<b>Cloudwatch custom metrics - instance</b> —
<b>Cloudwatch custom metrics - environment</b> —	<b>Log streaming</b> Activated	<b>Retention</b> 7
<b>Lifecycle</b> false		
<b>Updates</b>		
<b>Managed updates</b> Activated	<b>Deployment batch size</b> 100	<b>Deployment batch size type</b> Percentage
<b>Command timeout</b> 600	<b>Deployment policy</b> AllAtOnce	<b>Health threshold</b> Ok
<b>Ignore health check</b> false	<b>Instance replacement</b> false	
<b>Platform software</b>		
<b>Lifecycle</b> false	<b>Log groups</b> <a href="#">/aws/elasticbeanstalk/projektchmury-frontend-env</a>	<b>Log streaming</b> Activated

none / © 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

**Update level**  
minor **X-Ray enabled**  
Deactivated

**Environment properties**

Source	Key	Value
Plain text	VITE_API_URL	projekt-chmury-backend.us-east-1.elasticbeanstalk.com

**Instance metadata service (IMDS)**

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

**IMDSv1**  
With the current setting, the environment enables only IMDSv2.  
 Deactivated

**EC2 security groups**

Select security groups to control traffic.

Group name	Group ID	Name
awsb-e-muct4ccfvm-stack-AWSEBSecurityGroup-IUJAJGWd7rnNK	sg-082624d59e6c0396d	backend-env
awseb-e-qpb5mykyic-stack-AWSEBSecurityGroup-HN1oB1NMSpld	sg-07c2ccb59a910fce9	projektchmury-frontend-env
awseb-e-ymrckraxmj-stack-AWSEBSecurityGroup-QSUoebzqA2Ml	sg-0fbff690ca8991550	projektchmury-backend-env
default	sg-0339eb242986f6986	
ec2-rds-1	sg-082f9d6e474396f0c	
rds-ec2-1	sg-01897eb9d86089958	

**Instance types**

Add instance types for your environment with your preferred launch order. The order preference only applies to On-Demand Instances.

1. t3.micro	▼	▲	▼
2. t3.small	▼	▲	▼

[Add instance type](#)

**AMI ID**

Elastic Beanstalk selects a default Amazon Machine Image (AMI) for your environment based on the Region, platform version, and environment.

ami-0f78189a679d18874
-----------------------

## Integracja frontenu z Cloudwatch:

**Amazon CloudWatch monitoring**  
The time interval between when metrics are reported from the EC2 instances

**Monitoring interval**  
5 minute ▾

**Instance metadata service (IMDS)**  
Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#) ⓘ

**IMDSv1**  
With the current setting, the environment enables only IMDSv2.  
 Deactivated

**▼ Monitoring Info**

**Health reporting**  
Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The **EnvironmentHealth** custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#) ⓘ

**System**  
 Basic  
 Enhanced

**CloudWatch Custom Metrics - Instance**  
Choose metrics ▾

**CloudWatch Custom Metrics - Environment**  
Choose metrics ▾

**Health event streaming to CloudWatch Logs**  
Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

**Log streaming**  
 Activated (standard CloudWatch charges apply.)  
Log group: /aws/elasticbeanstalk/projektczhmury-frontend-env/environment-health.log ⓘ

**Retention**  
7 ▾

**Lifecycle**  
Keep logs after terminating environment ▾

**Instance log streaming to CloudWatch logs**  
Configure the instances in your environment to stream logs to CloudWatch logs. Yes

**Log groups:** /aws/elasticbeanstalk/projektczhmury-frontend-env ⓘ

**Log streaming**  
(standard CloudWatch charges apply.)  
 Activated

**Retention**  
7 ▾

**Lifecycle**  
Keep logs after terminating ... ▾

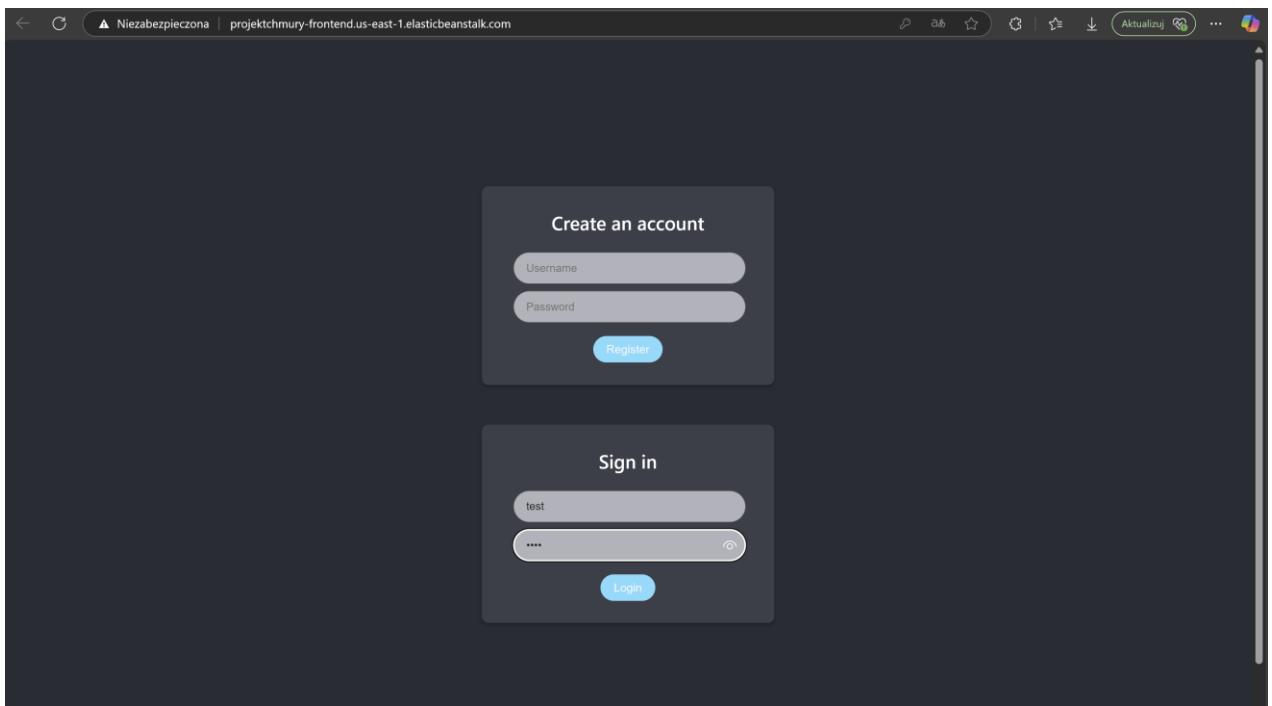
**▼ Managed platform updates** ⓘ  
Activate managed platform updates to apply platform patches.

**Managed updates**  
 Activated

**Managed actions role**  
arn:aws:iam::107378568397:role/LabRole

**Weekly update window**  
Monday at 22 : 48 UTC

**Update level**  
Minor and patch



Działający frontend i backend na AWS wchodząc a link dla frontenu, po zalogowaniu wysyłamy wiadomość do innego użytkownika (**test** do **marek**).

Welcome, test!

Send Message

Recipient

marek

Message

Cześć, przesyłam ci plik

Attach file (optional)

GDD concept (3).pdf

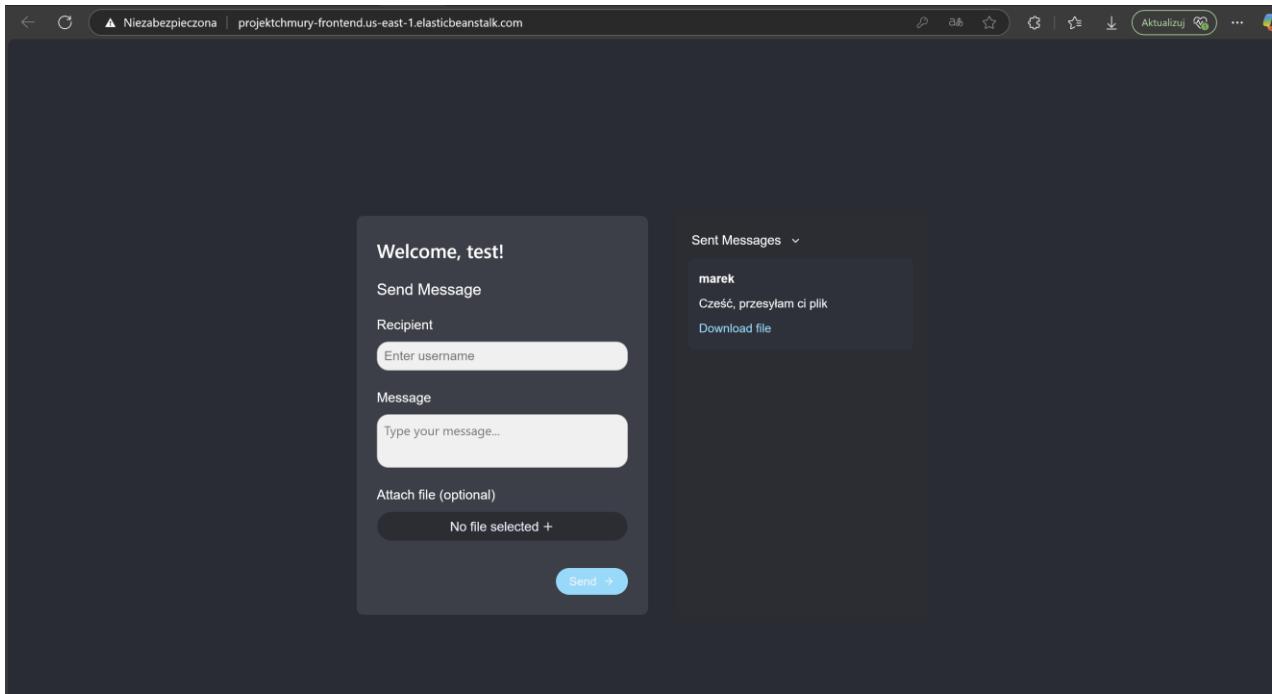
Send →

Komunikat z witryny projektchmury-east-1.elasticbeanstalk.com:

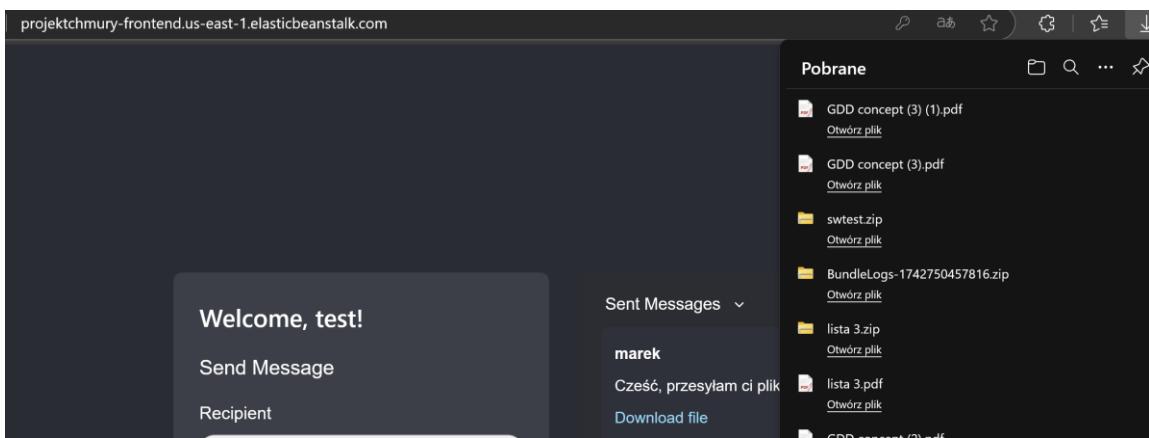
Message with file sent successfully!

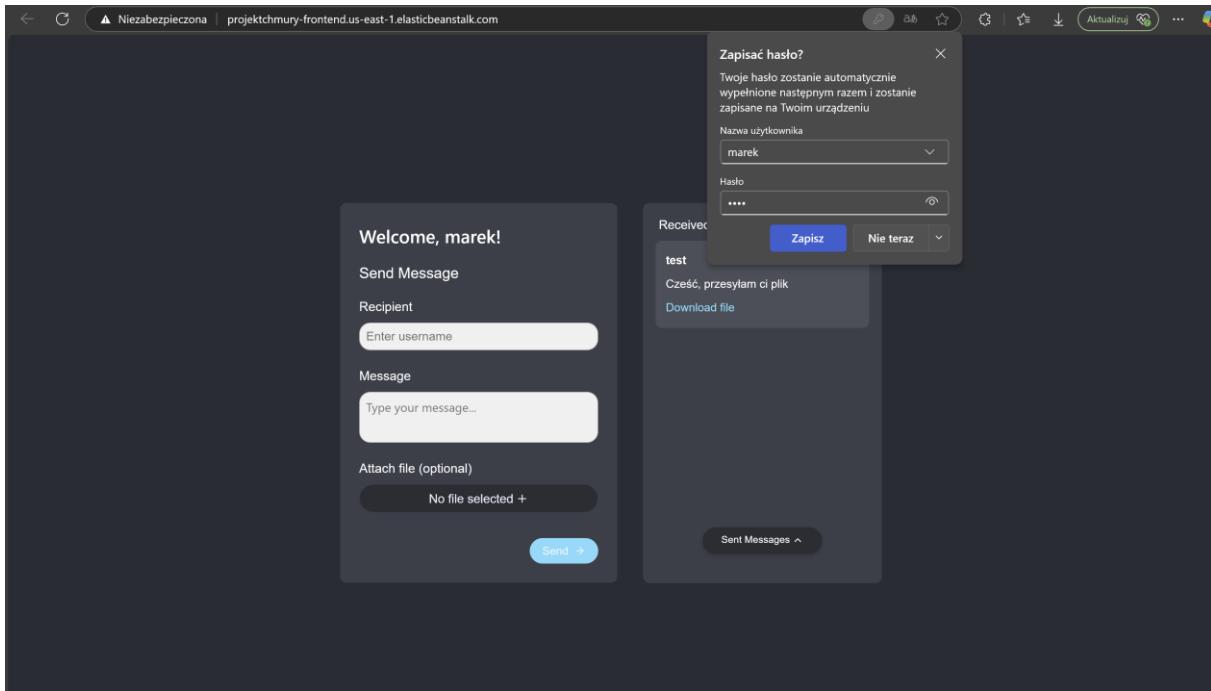
Received Messages

Sent Messages ^

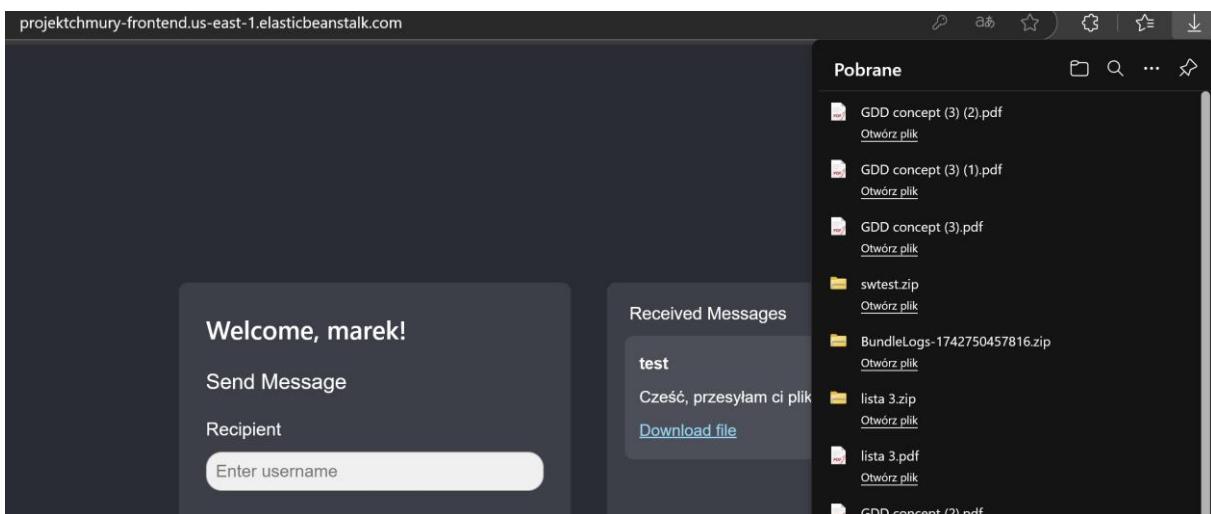


Po kliknięciu Download file plik się pobiera





Po zalogowaniu na drugie konto mamy wiadomość od test w otrzymanych wiadomościach



I również plik pobiera się bez problemu

- Przesyłane pliki przechowują się w S3:

This screenshot shows the Amazon S3 console for the 'projekt-chmury-uploads' bucket. The left sidebar lists various AWS services like General purpose buckets, Directory buckets, Table buckets, etc. The main area displays the contents of the S3 bucket. A table titled 'Objects (1)' shows a single file entry: '1742911229967-GDD concept (3) (2).pdf'. The file is a PDF type, last modified on March 25, 2025, at 15:00:31 (UTC+01:00), and is 86.5 KB in size. It is stored in the 'Standard' storage class. The table has columns for Name, Type, Last modified, Size, and Storage class.

**Bucket overview**

AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN) <a href="#">arn:aws:s3:::projekt-chmury-uploads</a>	Creation date March 18, 2025, 19:41:54 (UTC+01:00)
---	---	---

**Bucket Versioning**

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

**Bucket Versioning**  
Disabled

**Multi-factor authentication (MFA) delete**

An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Disabled

**Bucket overview**

AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN) <a href="#">arn:aws:s3:::projekt-chmury-uploads</a>	Creation date March 18, 2025, 19:41:54 (UTC+01:00)
---	---	---

**Block public access (bucket settings)**

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

**Block all public access**  
Off

► Individual Block Public Access settings for this bucket

- RDS:

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

- Databases (1)**: One database instance named **baza-chmury**.
- Status**: Available.
- Role**: Instance.
- Engine**: PostgreSQL.
- Region**: us-east-1c.
- Size**: db.t3.micro.
- Recommendations**: 6 informational.
- CPU**: 3.58%.
- Current activity**: 10 Connections.

## Connectivity & security

Endpoint & port	Networking	Security
<b>Endpoint</b> <a href="#">baza-chmury.cdle9v0rnyed.us-east-1.rds.amazonaws.com</a>	<b>Availability Zone</b> us-east-1c	<b>VPC security groups</b> awseb-e-qpb5mykyic-stack-AWSEBSecurityGroup-HN1oB1NMSPId (sg-07c2ccb59a910fce9) Active
<b>Port</b> 5432	<b>VPC</b> <a href="#">vpc-08e2f7b9bd2c1ab27</a>	rds-ec-2-1 (sg-01897eb9d86089958) Active
	<b>Subnet group</b> default	default (sg-0339eb242986f6986) Active
	<b>Subnets</b> <a href="#">subnet-0c6e24690d329a7e7</a> <a href="#">subnet-054fc5687b1b5f979</a> <a href="#">subnet-03e74672cba510707</a> <a href="#">subnet-067e210886b5cdd71</a> <a href="#">subnet-0dd7000c3bab5adbd</a> <a href="#">subnet-0fee47b94f4f1394f</a>	awseb-e-ymrcrkaxmj-stack-AWSEBSecurityGroup-QSUoebzA2Ml (sg-0fbff690ca8991550) Active
	<b>Network type</b> IPv4	awseb-e-muct4ccfvml-stack-AWSEBSecurityGroup-iUjAJGVd7mNK (sg-082624d59e6c0396d) Active
		<b>Publicly accessible</b> Yes
		<b>Certificate authority</b> <a href="#">Info</a> rds-ca-rsa2048-g1
		<b>Certificate authority date</b> May 26, 2061, 01:34 (UTC+02:00)
		<b>DB instance certificate expiration date</b> March 24, 2026, 00:17 (UTC+01:00)

## Security group rules (7)

Security group rules		Type	Rule
<a href="#">awseb-e-muct4ccfvm-stack-AWSEBSecurityGroup-iUjAJGVd7mNK (sg-082624d59e6c0396d)</a>	CIDR/IP - Outbound	0.0.0.0/0	
<a href="#">awseb-e-qpb5mykyic-stack-AWSEBSecurityGroup-HN1oB1NNMSpld (sg-07c2ccb59a910cfe9)</a>	CIDR/IP - Outbound	0.0.0.0/0	
<a href="#">awseb-e-ymrckaxmj-stack-AWSEBSecurityGroup-QSUoebqzA2Ml (sg-0fbff690ca8991550)</a>	CIDR/IP - Outbound	0.0.0.0/0	
<a href="#">default (sg-0339eb242986f6986)</a>	EC2 Security Group - Inbound	sg-0339eb242986f6986	
<a href="#">default (sg-0339eb242986f6986)</a>	CIDR/IP - Outbound	0.0.0.0/0	
<a href="#">rds-ec2-1 (sg-01897eb9d86089958)</a>	CIDR/IP - Inbound	109.243.146.84/32	
<a href="#">rds-ec2-1 (sg-01897eb9d86089958)</a>	EC2 Security Group - Inbound	sg-082f9d6e474396f0c	

## Instance

Configuration	Instance class	Storage	Monitoring
<b>DB instance ID</b> aza-chmury	<b>Instance class</b> db.t3.micro	<b>Encryption</b> Not enabled	<b>Monitoring type</b> Database Insights - Standard
<b>Engine version</b> 14.15	<b>vCPU</b> 2	<b>Storage type</b> General Purpose SSD (gp2)	<b>Performance Insights</b> Disabled
<b>RDS Extended Support</b> Enabled	<b>RAM</b> 1 GB	<b>Storage</b> 20 GiB	<b>Enhanced Monitoring</b> Disabled
<b>DB name</b> mydatabase	<b>Availability</b>	<b>Provisioned IOPS</b> -	<b>DevOps Guru</b> Disabled
<b>License model</b> Postgresql License	<b>Master username</b> postgres	<b>Storage throughput</b> -	
<b>Option groups</b> <a href="#">default:postgres-14</a> ⓘ In sync	<b>Master password</b> *****	<b>Storage autoscaling</b> Disabled	
<b>Amazon Resource Name (ARN)</b> <a href="#">arn:aws:rds:us-east-1:107378568397:db:aza-chmury</a>	<b>IAM DB authentication</b> Not enabled	<b>Storage file system configuration</b> Current	
<b>Resource ID</b> db-CWQQ3NTB7BQ7HZFXY7VBDRG3M	<b>Multi-AZ</b> No		
<b>Created time</b> March 18, 2025, 18:52 (UTC+01:00)	<b>Secondary Zone</b> -		
<b>DB instance parameter group</b> <a href="#">default:postgres14</a> ⓘ In sync			
<b>Deletion protection</b> Disabled			
<b>Architecture settings</b> Non-multitenant architecture			

Potwierdzenie poprawnej konfiguracji z RDS i prawidłowego przechowywania danych łącząc się z RDS poprzez psql:

```

SQL Shell (psql)  x  +  ~

Server [localhost]: psql --host=baza-chmury.cdle9v0rnyed.us-east-1.rds.amazonaws.com --port=5432 --username=postgres --dbname=mydatabase
Database [postgres]:
Port [5432]:
Username [postgres]:
Hasło użytkownika postgres:

psql (17.4, serwer 14.15)
OSTRZEŻENIE: strona kodowa konsoli (852) jest różna od kodowania Windows (1250)
8-bitowe znaki mogą nie wyglądać poprawnie. Przejrzyj odnośną
stronę "Notes for Windows users" by poznac szczegóły.
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, compression: wyłączone, ALPN: none)
Wpisz "help" by uzyskać pomoc.

postgres=> \c mydatabase
psql (17.4, serwer 14.15)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, compression: wyłączone, ALPN: none)
Jestes obecnie połączony do bazy danych "mydatabase" jako użytkownik "postgres".
mydatabase=> \dt
      Lista relacji
Schemat | Nazwa | Typ | Właściciel
-----+-----+-----+-----
public | message | tabela | postgres
public | users   | tabela | postgres
(2 wiersze)

mydatabase=> SELECT * FROM users;
 id | password | username
---+-----+-----
 1 | $2a$10$ybVR08wNFK3xAS09mwXw6uTgjBsKZsh7A4GfwkNFFZ4LpxqH/G7Di | se
 2 | $2a$10$V8DAnEmQT5Gtse1nt0pIaeHTChPFDjRlyvK8b10Q95qE8sZunoU.a | test
 3 | $2a$10$lsVI5ghCmNanbrAyUpxQ0YUcf0cfXjUckrjAZGxhCsn/FxSgiJz0 | marek
 52 | $2a$10$GdyHExpXR88KiesuMFQmmONzyT3fsy2MeTwIfB806Dh0/GhOoYZ0a | siema
(4 wiersze)

mydatabase=> SELECT * FROM message;
 id | content | file | author_id | recipient_id
---+-----+-----+-----+-----+
 1 | Cześć, przesyłam ci plik | /uploads/GDD concept (3).pdf |           |           |
 2 | Testtest | https://projekt-chmury-uploads.s3.amazonaws.com/1742911229967-GDD concept (3) (2).pdf | 2 | 3
(2 wiersze)

```

- Cognito:

The screenshot shows the Amazon Cognito User Pools Overview page for the 'projekt-chmury-user-pool'. The left sidebar shows the current user pool as 'projekt-chmury-user-pool'. The main area displays the following information:

- User pool information:**
  - User pool name: projekt-chmury-user-pool
  - User pool ID: us-east-1\_wF3wVlY9V
  - ARN: arn:aws:cognito-idp:us-east-1:107378568397:userpool/us-east-1\_wF3wVlY9V
  - Token signing key URL: [https://cognito-idp.us-east-1.amazonaws.com/us-east-1\\_wF3wVlY9V/.well-known/jwks.json](https://cognito-idp.us-east-1.amazonaws.com/us-east-1_wF3wVlY9V/.well-known/jwks.json)
  - Estimated number of users: 6
  - Created time: March 18, 2025 at 19:44 GMT+1
  - Last updated time: March 28, 2025 at 00:03 GMT+1
  - Feature plan: Essentials
- App clients and analytics:**
  - App clients (1):** Shows one client named 'projekt-chmury-client' with Client ID: 6cqk2jchr76qrq14ggumonicap.

**App client**

Configure app clients. App clients are the user pool authentication resources attached to your app. Select an app client to configure the permitted authentication actions for an app.

**App client name:** Info  
Enter a friendly name for your app client.

**project-chmury-client**

**Choice-based sign-in: ALLOW\_USER\_AUTH**  
Your user pool responds to sign-in requests with a list of available methods. Users can choose options like one-time passwords, biometric devices and security keys, and password-based sign-in with MFA.

**Sign in with username and password: ALLOW\_USER\_PASSWORD\_AUTH**  
Users can sign in with a username and password. This method sends the username and password directly to your user pool.

**Sign in with secure remote password (SRP): ALLOW\_USER\_SRP\_AUTH**  
Users can sign in with username and password. Your application uses SRP libraries in server-side or client-side sign-in operations to pass a password hash and verifier.

**Sign in with server-side administrative credentials: ALLOW\_ADMIN\_USER\_PASSWORD\_AUTH**  
Users can sign in with username and password in server-side authentication operations. This feature is not supported in HostedUI.

**Sign in with custom authentication flows from Lambda triggers: ALLOW\_CUSTOM\_AUTH**  
Users can sign in, optionally with username and password, and respond to custom challenges that you design in Lambda functions.

**Get new user tokens from existing authenticated sessions: ALLOW\_REFRESH\_TOKEN\_AUTH**  
Your application can store a longer-lived refresh token that renews user sessions without additional user prompts.

**Authentication flow session duration:** Info  
3 minutes  
Must be between 3 and 15 minutes.

**Refresh token expiration:** Info  
30 days 0 minutes  
Must be between 60 minutes and 10 years.

**Access token expiration:** Info  
0 days 60 minutes  
Must be between 5 minutes and 1 day. Value cannot be greater than refresh token expiration.

**ID token expiration:** Info  
0 days 60 minutes  
Must be between 5 minutes and 1 day. Value cannot be greater than refresh token expiration.

**Advanced security configurations - optional**

**Enable token revocation:** Info  
Amazon Cognito will add new claims to access and id tokens to enable revocation. This increases the size of tokens.

**Prevent user existence errors:** Info  
Amazon Cognito authentication APIs return a generic authentication failure response, indicating either the user name or password is incorrect, instead of indicating that the user was not found.

Warto zaznaczyć, że żeby automatycznie potwierdzać nowo założone konta (bez potwierdzenia konta nie będzie się dało na nie zalogować) musimy napisać własną funkcję Lambda, którą dodamy jako Extention do Cognito User Pool:

Lambda > Functions > auto-confirm-user

**auto-confirm-user**

**Function overview:** Info

**Description:**  
Last modified 1 week ago

**Function ARN:** arn:aws:lambda:us-east-1:107378568397:function:auto-confirm-user

**Function URL:** Info

**Code:** Test | Monitor | Configuration | Aliases | Versions

**Code source:** Info

**index.js:**

```

1 exports.handler = async (event) => {
2   event.response.autoConfirmUser = true;
3   return event;
4 };

```

**Extensions:** Info

**Lambda triggers (1):** Info

Add and configure Lambda triggers for your user pool. Cognito can invoke Lambda functions in your account to customize authentication actions. Lambda triggers allow you to customize how you register and confirm users, authenticate users, send messages, and generate tokens.

**Lambda triggers:** Attached Lambda trigger | Trigger event version

Pre sign-up Lambda trigger: auto-confirm-user

Spróbujmy założyć nowego użytkownika i zobaczymy czy pojawi się jego instancja na stronie AWS Cognito

Users (5) <small>info</small>					
View, edit, and create users in your user pool. Users that are enabled and confirmed can sign in to your user pool.					
Property:	User name	Email address	Email verified	Confirmation status	Status
<input type="radio"/>	testowy	-	No	Confirmed	<input checked="" type="checkbox"/> Enabled
<input type="radio"/>	nmbjjhb	-	No	Confirmed	<input checked="" type="checkbox"/> Enabled
<input type="radio"/>	anita	-	No	Confirmed	<input checked="" type="checkbox"/> Enabled
<input type="radio"/>	marek	-	No	Confirmed	<input checked="" type="checkbox"/> Enabled

### Create an account

test\_cognito

\*\*\*\*\*

**Register**

projektchmury-frontend.us-east-1.elasticbeanstalk.com

Komunikat z witryny projektchmury-frontend.us-east-1.elasticbeanstalk.com:

Your account has been created!

**OK**

Create an account

test\_cognito

\*\*\*\*\*

**Register**

Amazon Cognito		[Alt+S]	Search	United States (N. Virginia)	vodlabs/user5928724=Mateusz_Staszek_w @ 1073-7856-8397																																				
Amazon Cognito	View all																																								
Current user pool projekt-chmury-user-pool																																									
Overview																																									
<b>Applications</b> <ul style="list-style-type: none"> <li>App clients <a href="#">New</a></li> </ul>																																									
<b>User management</b> <ul style="list-style-type: none"> <li><a href="#">Users</a></li> <li>Groups</li> </ul>																																									
<b>Authentication</b> <ul style="list-style-type: none"> <li>Authentication methods</li> <li>Sign-in <a href="#">New</a></li> <li>Sign-up</li> <li>Social and external providers</li> </ul>																																									
<b>Users</b> <a href="#">Info</a>																																									
<p>View, edit, and create users in your user pool. Users that are enabled and confirmed can sign in to your user pool.</p>																																									
<table border="1"> <thead> <tr> <th>Property:</th> <th>User name</th> <th>Email address</th> <th>Email verified</th> <th>Confirmation status</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td></td> <td><a href="#">testowy</a></td> <td>-</td> <td>No</td> <td>Confirmed</td> <td>Enabled</td> </tr> <tr> <td></td> <td><a href="#">test_cognito</a></td> <td>-</td> <td>No</td> <td>Confirmed</td> <td>Enabled</td> </tr> <tr> <td></td> <td><a href="#">nmjbjjh</a></td> <td>-</td> <td>No</td> <td>Confirmed</td> <td>Enabled</td> </tr> <tr> <td></td> <td><a href="#">anita</a></td> <td>-</td> <td>No</td> <td>Confirmed</td> <td>Enabled</td> </tr> <tr> <td></td> <td><a href="#">marek</a></td> <td>-</td> <td>No</td> <td>Confirmed</td> <td>Enabled</td> </tr> </tbody> </table>						Property:	User name	Email address	Email verified	Confirmation status	Status		<a href="#">testowy</a>	-	No	Confirmed	Enabled		<a href="#">test_cognito</a>	-	No	Confirmed	Enabled		<a href="#">nmjbjjh</a>	-	No	Confirmed	Enabled		<a href="#">anita</a>	-	No	Confirmed	Enabled		<a href="#">marek</a>	-	No	Confirmed	Enabled
Property:	User name	Email address	Email verified	Confirmation status	Status																																				
	<a href="#">testowy</a>	-	No	Confirmed	Enabled																																				
	<a href="#">test_cognito</a>	-	No	Confirmed	Enabled																																				
	<a href="#">nmjbjjh</a>	-	No	Confirmed	Enabled																																				
	<a href="#">anita</a>	-	No	Confirmed	Enabled																																				
	<a href="#">marek</a>	-	No	Confirmed	Enabled																																				

Konto pojawia się w Cognito User Pool i automatycznie jest potwierdzone

Niezabezpieczona | projektchmury-frontend.us-east-1.elasticbeanstalk.com

Welcome, test\_cognito!

Send Message

Recipient: anita

Message: Przesyłam ci wiadomość z załącznikiem przez Cognito

Attach file (optional): PizzaShapes.txt

Send →

Sent Messages

**Komunikat z witryny projektchmury-frontend.us-east-1.elasticbeanstalk.com:**

Message with file sent successfully!

OK

Welcome, test\_cognito!

Sent Messages ▾

Send Message

Recipient

anita

Message

Przesyłam ci wiadomość z załącznikiem  
przez Cognito

Attach file (optional)

PizzaShapes.txt

Send →

Sent Messages ▾

**anita**

Przesyłam ci wiadomość z  
załącznikiem przez Cognito

Download file

I jak zalogujemy się na konto anita to również mamy wiadomość od test\_cognito

The screenshot shows a dark-themed web application interface. On the left, there's a 'Send Message' form with fields for 'Recipient' (placeholder 'Enter username') and 'Message' (placeholder 'Type your message...'). Below these is an 'Attach file (optional)' section with a button 'No file selected +'. A 'Send' button with a right-pointing arrow is at the bottom. To the right of the form is a sidebar titled 'Received Messages' containing two messages from 'test\_cognito' and 'marek'. Each message includes a 'Download file' link. The 'test\_cognito' message is timestamped 'Przesyłam ci wiadomość z załącznikiem przez Cognito'. The 'marek' message is timestamped 'Wysyłam ci załącznik z użyciem autentykacji przez Cognito'. At the bottom of the sidebar is a 'Sent Messages' section with a single message from 'marek' with the text 'Hej'.

projekt-chmury-backend.us-east-1.elasticbeanstalk.com/api/files/download/11

I oczywiście plik się pobiera, a więc wszystko działa tak jak należy

This screenshot is similar to the previous one but shows a different state. The 'Received Messages' sidebar now lists multiple messages from 'test\_cognito' and 'marek', each with a 'Download file' link. The messages from 'test\_cognito' are identical to the ones in the previous screenshot. The messages from 'marek' are: '1743779203268-GDD concept (1).pdf', '1743780318856-GDD concept (1).pdf', '174377221469-(1).gitignore', '1743731848866-.gitignore', '1743775677111-.gitignore', '174377221469-.gitignore', '1743770304062-.gitignore', '1743774484356-.gitignore', and '1743779203268-GDD concept.pdf'. The 'Sent Messages' section is still present at the bottom of the sidebar.

- Cloudwatch i pojawiające się logi potwierdzające poprawne podłączenie:

The screenshot shows the AWS CloudWatch 'Live Tail' interface. At the top, there are buttons for Filter, Actions, Clear, Cancel, and Start. A status bar indicates '0 events/sec, 0% displayed' and the time '00:20:30'. Below this is a search bar labeled 'Highlight term' with the placeholder 'Highlight up to 5 terms (Not case sensitive)'. The main area displays a table of log entries with columns: Timestamp (UTC), Message, Log group, and Log stream. The logs show several DEBUG-level messages from a backend service, indicating the handling of an OPTIONS /api/auth/login request and a POST /api/auth/login request, along with a successful mapping of a RequestHandlerMapping to an AuthController#loginUser method. The log entries are timestamped from 2025-04-06T20:04:28.800Z to 2025-04-06T20:04:28.600Z.

Timestamp (UTC)	Message	Log group	Log stream
2025-04-06T20:04:28.800Z	2025-04-06T20:04:28.472Z DEBUG 1 --- [back... 107378568397:/aws/elastice... Link ↗		
2025-04-06T20:04:28.472Z DEBUG 1 --- [backend] [nio-8080-exec-3] o.s.security.web.FilterChainProxy : Securing OPTIONS /api/auth/login			
2025-04-06T20:04:28.800Z	2025-04-06T20:04:28.602Z DEBUG 1 --- [back... 107378568397:/aws/elastice... Link ↗		
2025-04-06T20:04:28.602Z DEBUG 1 --- [backend] [nio-8080-exec-4] o.s.security.web.FilterChainProxy : Securing POST /api/auth/login			
2025-04-06T20:04:28.800Z	2025-04-06T20:04:28.603Z DEBUG 1 --- [back... 107378568397:/aws/elastice... Link ↗		
2025-04-06T20:04:28.603Z DEBUG 1 --- [backend] [nio-8080-exec-4] o.s.security.web.FilterChainProxy : Secure POST /api/auth/login			
2025-04-06T20:04:28.800Z	2025-04-06T20:04:28.605Z DEBUG 1 --- [back... 107378568397:/aws/elastice... Link ↗		
2025-04-06T20:04:28.605Z DEBUG 1 --- [backend] [nio-8080-exec-4] o.s.web.servlet.DispatcherServlet : POST "/api/auth/login", parameters= {}			
2025-04-06T20:04:28.800Z	2025-04-06T20:04:28.605Z DEBUG 1 --- [back... 107378568397:/aws/elastice... Link ↗		
2025-04-06T20:04:28.605Z DEBUG 1 --- [backend] [nio-8080-exec-4] s.w.s.m.a.RequestMappingHandlerMapping : Mapped to pl.projekt_chmury.backend.controller.AuthController#loginUser			
2025-04-06T20:04:28.800Z	2025-04-06T20:04:28.606Z DEBUG 1 --- [back... 107378568397:/aws/elastice... Link ↗		
2025-04-06T20:04:28.606Z DEBUG 1 --- [backend] [nio-8080-exec-4] m.m.a.RequestResponseMethodProcessor : Read "application/json;charset=UTF-8" to [pl.projekt_chmury.backend.model.User@57aa621f]			

### 3.3 Infrastruktura za pomocą Terraformu

- Za pomocą takiego main.tf:

```
provider "aws" {
  region = "us-east-1"
}

#####
# Losowy sufiks do nazw #
#####

resource "random_string" "suffix" {
  length = 4
  special = false
  upper  = false
}

#####
# RDS — PostgreSQL instance #
#####

resource "aws_db_instance" "mydb" {
  identifier      = "terraform-mydb-${random_string.suffix.result}"
  allocated_storage = 20
  engine          = "postgres"
  engine_version   = "14.15"
  instance_class    = "db.t3.micro"
  db_name          = "mydatabase"
}
```

```

username      = "postgres"
password      = "admin1234"
parameter_group_name = "default.postgres14"
skip_final_snapshot = true
publicly_accessible = true

# Włącz eksport logów do CloudWatch
enabled_cloudwatch_logs_exports = ["postgresql", "upgrade"]
}

output "db_endpoint" {
  value = aws_db_instance.mydb.address
}

#####
# S3 — File uploads & App Versions
#####
resource "aws_s3_bucket" "upload_bucket" {
  bucket = "terraform-projekt-chmury-uploads-${random_string.suffix.result}"
}

# Przesyłanie plików do S3
resource "aws_s3_object" "backend_app_zip" {
  bucket = aws_s3_bucket.upload_bucket.bucket
  key    = "backend-app.zip"
  source = "../backend/backend-app.zip"
  etag   = filemd5("../backend/backend-app.zip")
}

resource "aws_s3_object" "frontend_app_zip" {
  bucket = aws_s3_bucket.upload_bucket.bucket
  key    = "frontend-app.zip"
  source = "../frontend/frontend-app.zip"
  etag   = filemd5("../frontend/frontend-app.zip")
}

#####
# CloudWatch — Log Group #
#####
resource "aws_cloudwatch_log_group" "app_logs" {
  name      = "/terraform/projekt-chmury/app-logs-${random_string.suffix.result}"
  retention_in_days = 30
}

#####
# Cognito — User Pool and Client
#####

resource "aws_cognito_user_pool_client" "chat_pool_client" {
  name      = "terraform-projekt-chmury-client-${random_string.suffix.result}"
  user_pool_id = aws_cognito_user_pool.chat_pool.id
  explicit_auth_flows = ["ALLOW_USER_PASSWORD_AUTH", "ALLOW_REFRESH_TOKEN_AUTH"]
}

#####
# ECR Repositories
#####

```

```

resource "aws_ecr_repository" "backend_repo" {
  name = "terraform-projekt-chmury-backend-${random_string.suffix.result}"
}

resource "aws_ecr_repository" "frontend_repo" {
  name = "terraform-projekt-chmury-frontend-${random_string.suffix.result}"
}

#####
# Elastic Beanstalk — Backend Application
#####

resource "aws_elastic(beanstalk_application" "backend_app" {
  name      = "terraform-backend-app-${random_string.suffix.result}"
  description = "Backend for Projekt Chmury"
}

resource "aws_elastic(beanstalk_application_version" "backend_app_version" {
  name      = "terraform-backend-app-v1-${filemd5("../backend/backend-app.zip")}"
  application = aws_elastic(beanstalk_application.backend_app.name
  bucket     = aws_s3_bucket.upload_bucket.bucket
  key        = aws_s3_object.backend_app_zip.key
  description = "Backend application version 1"
}

resource "aws_elastic(beanstalk_environment" "backend_env" {
  name      = "terraform-backend-env-${random_string.suffix.result}"
  application      = aws_elastic(beanstalk_application.backend_app.name
  solution_stack_name = "64bit Amazon Linux 2 v4.0.8 running Docker"
  version_label    = aws_elastic(beanstalk_application_version.backend_app_version.name

setting {
  namespace = "aws:elasticbeanstalk:environment"
  name      = "ServiceRole"
  value     = "arn:aws:iam::107378568397:role/LabRole"
}

setting {
  namespace = "aws:autoscaling:launchconfiguration"
  name      = "iamInstanceProfile"
  value     = "LabInstanceProfile"
}

setting {
  namespace = "aws:elasticbeanstalk:application:environment"
  name      = "SPRING_DATASOURCE_URL"
  value     = "jdbc:postgresql://${aws_db_instance.mydb.address}:5432/mydatabase"
}

setting {
  namespace = "aws:elasticbeanstalk:application:environment"
  name      = "SPRING_DATASOURCE_USERNAME"
  value     = aws_db_instance.mydb.username
}

setting {
  namespace = "aws:elasticbeanstalk:application:environment"
  name      = "SPRING_DATASOURCE_PASSWORD"
}

```

```

    value = aws_db_instance.mydb.password
}

setting {
  namespace = "aws:elasticbeanstalk:application:environment"
  name     = "S3_BUCKET_NAME"
  value    = aws_s3_bucket.upload_bucket.bucket
}

# Przekazanie konfiguracji Cognito do backendu
setting {
  namespace = "aws:elasticbeanstalk:application:environment"
  name     = "aws.cognito.userPoolId"
  value    = aws_cognito_user_pool.chat_pool.id
}
setting {
  namespace = "aws:elasticbeanstalk:application:environment"
  name     = "aws.cognito.clientId"
  value    = aws_cognito_user_pool_client.chat_pool_client.id
}
# Ustawienia log streamingu do CloudWatch
setting {
  namespace = "aws:elasticbeanstalk:cloudwatch:logs"
  name     = "StreamLogs"
  value    = "true"
}
setting {
  namespace = "aws:elasticbeanstalk:cloudwatch:logs"
  name     = "DeleteOnTerminate"
  value    = "true"
}
setting {
  namespace = "aws:elasticbeanstalk:cloudwatch:logs"
  name     = "RetentionInDays"
  value    = "7"
}
wait_for_ready_timeout = "30m"
}

#####
# Elastic Beanstalk — Frontend Application
#####

resource "aws_elastic(beanstalk_application" "frontend_app" {
  name     = "terraform-frontend-app-${random_string.suffix.result}"
  description = "Frontend for Projekt Chmury"
}

resource "aws_elastic(beanstalk_application_version" "frontend_app_version" {
  name     = "terraform-frontend-app-v1-${filemd5("../frontend/frontend-app.zip")}"
  application = aws_elastic(beanstalk_application.frontend_app.name
  bucket    = aws_s3_bucket.upload_bucket.bucket
  key       = aws_s3_object.frontend_app_zip.key
  description = "Frontend application version 1"
}

resource "aws_elastic(beanstalk_environment" "frontend_env" {

```

```

name      = "terraform-frontend-env-${random_string.suffix.result}"
application = aws_elastic(beanstalk_application.frontend_app.name
solution_stack_name = "64bit Amazon Linux 2 v4.0.8 running Docker"
version_label = aws_elastic(beanstalk_application_version.frontend_app_version.name

setting {
  namespace = "aws:elasticbeanstalk:application:environment"
  name     = "VITE_API_URL"
  value    = "http://${aws_elastic(beanstalk_environment.backend_env.cname).api}"
}

# Usuwamy blok "Image" – wersja aplikacji (zip) określa już obraz
setting {
  namespace = "aws:autoscaling:launchconfiguration"
  name     = "IamInstanceProfile"
  value    = "LabInstanceProfile"
}
# Ustawienia log streamingu do CloudWatch
setting {
  namespace = "aws:elasticbeanstalk:cloudwatch:logs"
  name     = "StreamLogs"
  value    = "true"
}
setting {
  namespace = "aws:elasticbeanstalk:cloudwatch:logs"
  name     = "DeleteOnTerminate"
  value    = "true"
}
setting {
  namespace = "aws:elasticbeanstalk:cloudwatch:logs"
  name     = "RetentionInDays"
  value    = "7"
}
wait_for_ready_timeout = "30m"
}

#####
# Lambda function for Cognito triggers
#####

resource "aws_lambda_function" "auto_confirm_user" {
  function_name = "auto-confirm-user"
  runtime      = "python3.9"
  handler     = "lambda_function.lambda_handler"
  role        = aws_iam_role.lambda_cognito_triggers.arn

  # Plik zip z kodem Lambdy (np. w folderze lambda/)
  filename     = "${path.module}/lambda/auto_confirm_user.zip"
  source_code_hash = filebase64sha256("${path.module}/lambda/auto_confirm_user.zip")
}

resource "aws_iam_role" "lambda_cognito_triggers" {
  name          = "lambda_cognito_triggers"
  assume_role_policy = data.aws_iam_policy_document.lambda_assume_role_policy.json
}

```

```

data "aws_iam_policy_document" "lambda_assume_role_policy" {
  statement {
    actions = ["sts:AssumeRole"]
    principals {
      type    = "Service"
      identifiers = ["lambda.amazonaws.com"]
    }
  }
}

resource "aws_iam_role_policy_attachment" "lambda_basic_execution" {
  role      = aws_iam_role.lambda_cognito_triggers.name
  policy_arn = "arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole"
}

# Sam user pool:
resource "aws_cognito_user_pool" "chat_pool" {
  name = "terraform-projekt-chmury-user-pool-${random_string.suffix.result}"

  # Podłączamy naszą Lambdę w lambda_config
  lambda_config {
    pre_sign_up = aws_lambda_function.auto_confirm_user.arn
  }

  # Polityka haseł
  password_policy {
    minimum_length    = 6
    require_lowercase = true
    require_numbers   = true
    require_symbols   = false
    require_uppercase = true
    temporary_password_validity_days = 7
  }

  auto_verified_attributes = ["email"]
}

#####
# Outputs
#####
output "backend_url" {
  value = "http://${aws_elastic(beanstalk_environment.backend_env).cname}"
}

output "frontend_url" {
  value = "http://${aws_elastic(beanstalk_environment.frontend_env).cname}"
}

output "database_endpoint" {
  value = aws_db_instance.mydb.address
}

output "s3_bucket" {
  value = aws_s3_bucket.upload_bucket.bucket
}

```

```

output "cognito_user_pool_id" {
  value = aws_cognito_user_pool.chat_pool.id
}

output "cognito_client_id" {
  value = aws_cognito_user_pool_client.chat_pool_client.id
}

```

Udało się pomyślnie postawić całą działającą infrastrukturę korzystając z Terraforma (poprzez terraform init, terraform plan, a następnie terraform apply). Oczywiście oprócz tego musielibyśmy wykonać parę zmian w kodzie, w plikach takich jak .env, vite.config.js dla frontendu, czy application.properties oraz SecurityConfig dla backendu. Tak prezentuje się działająca infrastruktura na stronie AWS:

- Elastic Beanstalk (frontend, backend):

Environment name	Health	Application	Platform	Domain	Running ver...	Tier name	Date created	Last modified
backend-env	Suspended	projektchmury...	Docker runnin...	projektchmury-backend.us-eas...	1.0.2	WebServer	March 23, 202...	March 24, 202...
projektchmury-backend-env	Degraded	projektchmury...	Docker runnin...	projekt-chmury-backend.us-ea...	1.0.24	WebServer	March 23, 202...	April 4, 2025 1...
projektchmury-frontend-env	Ok	projektchmury...	Docker runnin...	projektchmury-frontend.us-ea...	1.0.15	WebServer	March 24, 202...	April 5, 2025 1...
terraform-backend-env-474	Ok	terraform-bac...	Docker runnin...	terraform-backend-env-474x.e...	terraform-bac...	WebServer	April 6, 2025 0...	April 6, 2025 2...
terraform-frontend-env-474x	Ok	terraform-fron...	Docker runnin...	terraform-frontend-env-474x...	terraform-fron...	WebServer	April 6, 2025 0...	April 6, 2025 2...

#### Środowisko frontendu:

**Service access**

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

<b>Service role</b> arn:aws:iam::107378568397:role/aws-service-role/elasticbeanstalk.amazonaws.com/AWSServiceRoleForElasticBeanstalk	<b>EC2 key pair</b> vokey	<b>EC2 instance profile</b> LabInstanceProfile
---	------------------------------	---

**Networking and database**

Configure VPC settings, and subnets for your environment's EC2 instances and load balancer. Set up an Amazon RDS database that's integrated with your environment.

No options configured

**Instance traffic and scaling**

Customize the capacity and scaling for your environment's instances. Select security groups to control instance traffic. Configure the software that runs on your environment's instances by setting platform-specific options.

<b>Instances</b>	<b>EC2 Security Groups</b>	<b>Max instances</b>
IMDSv1 Activated	awseb-e-mjbjfhkmx-stack-AWSEBLoadBalancerSecurityGroup-LMw8sqkJNjPP,default,awseb-e-mjbjfhkmx-stack-AWSEBSecurityGroup-LyepzvqoBwp	4
<b>Capacity</b>	<b>On-demand base</b>	<b>On-demand above base</b>
Environment type Load balanced	0	70
Fleet composition On-Demand instances	Scaling cooldown	Processor type
Deactivated	360	x86_64
Instance types t3.micro, t3.small	AMI ID	Availability Zones
	ami-03d1621701100ca43	Any

Metric	Statistic	Unit
NetworkOut	Average	Bytes
Period	Breach duration	Upper threshold
5	5	6000000
Scale up increment	Lower threshold	Scale down increment
1	2000000	-1
<b>Load balancer</b>		
Load balancer visibility	Load balancer type	Cross-zone load balancing
public	classic	Deactivated
Connection draining	Connection draining timeout	Health check timeout
Deactivated	20	5
Health check interval	Health check unhealthy threshold	Health check healthy threshold
10	5	3

#### Updates, monitoring, and logging [Info](#)

Define when and how Elastic Beanstalk deploys changes to your environment. Manage your application's monitoring and logging settings, instances, and other environment resources.

[Edit](#)

##### Monitoring

Log group <a href="#">/aws/elasticbeanstalk/terraform-frontend-env-474x/environment-health.log</a>	System enhanced	Cloudwatch custom metrics - instance —
---	--------------------	---

##### Cloudwatch custom metrics - environment

Log streaming	Retention
Activated	7

##### Lifecycle

false

##### Updates

Managed updates	Deployment batch size	Deployment batch size type
Deactivated	100	Percentage
Command timeout	Deployment policy	Health threshold
600	AllAtOnce	Ok

##### Ignore health check

false

##### Instance replacement

false

##### Platform software

Lifecycle	Log groups <a href="#">/aws/elasticbeanstalk/terraform-frontend-env-474x</a>	Log streaming Activated
Proxy server	Logs retention 7	Rotate logs Deactivated

##### X-Ray enabled

Deactivated

##### Environment properties

Source	▼   Key	▲   Value
Plain text	VITE_API_URL	<a href="http://terraform-backend-env-474x.eba-hppnpfaf.us-east-1.elastic...">http://terraform-backend-env-474x.eba-hppnpfaf.us-east-1.elastic...</a>

## Backend:

#### Configuration [Info](#)

[Cancel](#)

##### Service access [Info](#)

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

Service role	EC2 key pair	EC2 instance profile
<a href="#">arn:aws:iam::107378568397:role/LabRole</a>	vockey	<a href="#">LabInstanceProfile</a>

##### Networking and database [Info](#)

Configure VPC settings, and subnets for your environment's EC2 instances and load balancer. Set up an Amazon RDS database that's integrated with your environment.

No options configured

### Instance traffic and scaling [Info](#)

Customize the capacity and scaling for your environment's instances. Select security groups to control instance traffic. Configure the software that runs on your environment's instances by setting platform-specific options.

#### Instances

##### IMDSv1

Activated

##### EC2 Security Groups

awseb-e-pwmvkqzti-stack-AWSEBLoadBalancerSecurityGroup-c9sssT7DHpHQ,default,awseb-e-pwmvkqzti-stack-AWSEBSecurityGroup-LNmXThRUUpafy

#### Capacity

##### Environment type

Load balanced

##### Min instances

1

##### Max instances

4

##### Fleet composition

On-Demand instances

##### On-demand base

0

##### On-demand above base

70

##### Capacity rebalancing

Deactivated

##### Scaling cooldown

360

##### Processor type

x86\_64

##### Instance types

t3.micro, t3.small

##### AMI ID

ami-03d1621701100ca43

##### Availability Zones

Any

##### Metric

NetworkOut

##### Statistic

Average

##### Unit

Bytes

##### Period

5

##### Breach duration

5

##### Upper threshold

6000000

##### Scale up increment

1

##### Lower threshold

2000000

##### Scale down increment

-1

#### Load balancer

##### Load balancer visibility

public

##### Load balancer type

classic

##### Cross-zone load balancing

Deactivated

##### Connection draining

Deactivated

##### Connection draining timeout

20

##### Health check timeout

5

##### Health check interval

10

##### Health check unhealthy threshold

5

##### Health check healthy threshold

3

### Updates, monitoring, and logging [Info](#)

Define when and how Elastic Beanstalk deploys changes to your environment. Manage your application's monitoring and logging settings, instances, and other environment resources.

#### Monitoring

##### Log group

/aws/elasticbeanstalk/terraform-backend-env-474x/environment-health.log



##### System

enhanced

##### Cloudwatch custom metrics - instance

—

##### Cloudwatch custom metrics - environment

—

##### Log streaming

Activated

##### Retention

7

#### Lifecycle

false

#### Updates

##### Managed updates

Deactivated

##### Deployment batch size

100

##### Deployment batch size type

Percentage

##### Command timeout

600

##### Deployment policy

AllAtOnce

##### Health threshold

Ok

##### Ignore health check

false

##### Instance replacement

false

#### Platform software

##### Lifecycle

false

##### Log groups

/aws/elasticbeanstalk/terraform-backend-env-474x

##### Log streaming

Activated

##### Proxy server

nginx

##### Logs retention

7

##### Rotate logs

Deactivated

##### X-Ray enabled

Deactivated

#### Environment properties

Source	Key	Value
Plain text	aws.cognito.clientId	3cu3kld5u5fqj55630a89o94v
Plain text	aws.cognito.userPoolId	us-east-1_FkuffQRKs
Plain text	S3_BUCKET_NAME	terraform-projekt-chmury-uploads-474x
Plain text	SPRING_DATASOURCE_PASSWORD	admin1234
Plain text	SPRING_DATASOURCE_URL	jdbc:postgresql://terraform-mydb-474x.cdle9v0myed.us-east-1.rds.amazonaws.com:5432/terraform-mydb-474x?sslmode=require
Plain text	SPRING_DATASOURCE_USERNAME	postgres

- RDS:

**Databases (2)**

DB identifier	Status	Role	Engine	Region ...	Size	Recommendations	CPU	Current activity	Mi
baza-chmury	Available	Instance	PostgreSQL	us-east-1c	db.t3.micro	6 Informational	4.68%	10 Connections	av.
terraform-mydb-474x	Available	Instance	PostgreSQL	us-east-1f	db.t3.micro	5 Informational	4.64%	10 Connections	no

**terraform-mydb-474x**

**Summary**

DB identifier terraform-mydb-474x	Status <span style="color: green;">Available</span>	Role Instance	Engine PostgreSQL	Recommendations <span style="color: blue;">5 Informational</span>
CPU  4.39%	Class db.t3.micro	Current activity  10 Connections	Region & AZ us-east-1f	

**Connectivity & security**

<b>Endpoint &amp; port</b>	<b>Networking</b>	<b>Security</b>
Endpoint  <a href="#">terraform-mydb-474x.cde9v0rnyed.us-east-1.rds.amazonaws.com</a>	Availability Zone us-east-1f	VPC security groups <a href="#">rds-ec2-1 (sg-01897eb9d86089958)</a>  Active <a href="#">default (sg-0339eb242986f6986)</a>  Active <a href="#">ec2-rds-1 (sg-082f9d6e474396f0c)</a>  Active
Port 5432	VPC <a href="#">vpc-08e2f7b9bd2c1ab27</a>	Subnet group default
	Subnets <a href="#">subnet-0c6e24690d329a7e7</a> <a href="#">subnet-054fc5687b1b5f979</a> <a href="#">subnet-03e74672cba510707</a> <a href="#">subnet-067e21088655cd7d1</a> <a href="#">subnet-0dd7000c3bab5adbd</a> <a href="#">subnet-0fee47b94f4f1394f</a>	Network type IPv4
		Publicly accessible Yes
		Certificate authority <a href="#">rds-ca-rsa2048-g1</a>
		Certificate authority date May 26, 2061, 01:34 (UTC+02:00)
		DB instance certificate expiration date April 06, 2026, 02:19 (UTC+02:00)

**Security group rules (6)**

Security group	Type	Rule
<a href="#">default (sg-0339eb242986f6986)</a>	EC2 Security Group - Inbound	sg-0339eb242986f6986
<a href="#">default (sg-0339eb242986f6986)</a>	CIDR/IP - Outbound	0.0.0.0/0
<a href="#">ec2-rds-1 (sg-082f9d6e474396f0c)</a>	EC2 Security Group - Outbound	sg-01897eb9d86089958
<a href="#">rds-ec2-1 (sg-01897eb9d86089958)</a>	CIDR/IP - Inbound	109.243.146.84/32
<a href="#">rds-ec2-1 (sg-01897eb9d86089958)</a>	CIDR/IP - Inbound	0.0.0.0/0
<a href="#">rds-ec2-1 (sg-01897eb9d86089958)</a>	EC2 Security Group - Inbound	sg-082f9d6e474396f0c

### Instance

#### Configuration

DB instance ID  
terraform-mydb-474x

Engine version  
14.15

RDS Extended Support  
Enabled

DB name  
mydatabase

License model  
Postgresql License

Option groups  
default:postgres-14 In sync

Amazon Resource Name (ARN)  
 arn:aws:rds:us-east-1:107378568397:db:terraform-mydb-474x

Resource ID  
db-HCWTOXZRLPLEPAU3YRIW7K5Q5E

Created time  
April 05, 2025, 19:22 (UTC+02:00)

DB instance parameter group  
default:postgres14 In sync

Deletion protection  
Disabled

Architecture settings  
Non-multitenant architecture

#### Instance class

Instance class  
db.t3.micro

vCPU  
2

RAM  
1 GB

#### Availability

Master username  
postgres

Master password  
\*\*\*\*\*

IAM DB authentication  
Not enabled

Multi-AZ  
Yes

Secondary Zone  
us-east-1c

#### Storage

Encryption  
Not enabled

Storage type  
General Purpose SSD (gp2)

Storage  
20 GiB

Provisioned IOPS  
-

Storage throughput  
-

Storage autoscaling  
Disabled

Storage file system configuration  
Current

#### Monitoring

Monitoring type  
Database Insights - Standard

Performance Insights  
Disabled

Enhanced Monitoring  
Disabled

DevOps Guru  
Disabled

- S3:

Amazon S3

**Amazon S3**

- General purpose buckets
- Directory buckets
- Table buckets
- Access Grants
- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- IAM Access Analyzer for S3

**Account snapshot - updated every 24 hours** All AWS Regions

Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. [Learn more](#)

[View Storage Lens dashboard](#)

**General purpose buckets** (3) [Info](#) All AWS Regions

Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
elasticbeanstalk-us-east-1-107378568397	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	March 18, 2025, 19:41:54 (UTC+01:00)
projekt-chmury-uploads	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	March 18, 2025, 19:41:54 (UTC+01:00)
terraform-projekt-chmury-uploads-474x	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	April 6, 2025, 00:32:37 (UTC+02:00)

[Create bucket](#)

**terrafarm-projekt-chmury-uploads-474x** [Info](#)

[Objects](#) [Metadata](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

**Objects (3)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permission.

Name	Type	Last modified	Size	Storage class
1743970536279-.gitattributes	gitattributes	April 6, 2025, 22:15:38 (UTC+02:00)	38.0 B	Standard
backend-app.zip	zip	April 6, 2025, 21:56:59 (UTC+02:00)	110.0 MB	Standard
frontend-app.zip	zip	April 6, 2025, 21:56:59 (UTC+02:00)	44.7 MB	Standard

### Jak widać obrazy aplikacji w postaci plików .zip prawidłowo trafiły do S3 za pomocą terraforma

Amazon Cognito

User pools

**Amazon Cognito**

- User pools
- Identity pools

**User pools (1) [Info](#)**

View and configure your user pools. User pools are directories of federated and local user profiles. They provide authentication options for your users.

User pool name	User pool ID	Created time	Last updated time
terraform-projekt-chmury-user-pool-474x	us-east-1_FkuffQRK5	Yesterday	3 hours ago

- Cognito:

## Overview: terraform-projekt-chmury-user-pool-474x [Info](#)

[Rename](#)

### User pool information

**User pool name**  
terraform-projekt-chmury-user-pool-474x

**Token signing key URL**  
[https://cognito-idp.us-east-1.amazonaws.com/us-east-1\\_FkuffQRkS/.well-known/jwks.json](https://cognito-idp.us-east-1.amazonaws.com/us-east-1_FkuffQRkS/.well-known/jwks.json)

**Created time**  
April 6, 2025 at 00:32 GMT+2

**User pool ID**  
[us-east-1\\_FkuffQRkS](#)

**Last updated time**  
April 6, 2025 at 22:11 GMT+2

**ARN**  
[arn:aws:cognito-idp:us-east-1:107378568397:userpool/us-east-1\\_FkuffQRkS](#)

**Estimated number of users**  
6

**Feature plan**  
[Essentials](#)

## App client: terraform-projekt-chmury-client-474x [Info](#)

[Delete](#)

### App client information

**App client name**  
terraform-projekt-chmury-client-474x

**Authentication flow session duration**  
3 minutes

**Created time**  
April 6, 2025 at 00:32 GMT+2

**Client ID**  
[3cu3kld5u5fqL55630a89o94v](#)

**Refresh token expiration**  
30 day(s)

**Last updated time**  
April 6, 2025 at 00:32 GMT+2

**Client secret**  
-

**Access token expiration**  
1 hour(s)

**Authentication flows**  
[Username and password](#)  
Get user tokens from existing authenticated sessions

**ID token expiration**  
1 hour(s)

**Advanced authentication settings**  
Enable token revocation

## Authentication methods [Info](#)

### Email [Info](#)

Configure how your user pool sends email messages to users.

[Edit](#)

**Email provider**  
Send email with Cognito

**FROM email address**  
no-reply@verificationemail.com

**SES Region**  
US East (N. Virginia)

**REPLY-TO email address**  
-

### SMS [Info](#)

Configure how your user pool sends SMS messages to your users. Recipient message and data rates apply.

[Edit](#)

**IAM role ARN**  
-

**SNS Region**  
-

**⚠ Configure AWS service dependencies to complete your SMS message setup**

To send SMS messages from this user pool, you must complete the following additional steps if you haven't already done so. [Learn more](#)

The service links that follow may redirect you to a different AWS Region.

- ▶ Request an Amazon SNS spending limit increase
- ▶ Move to Amazon SNS production environment
- ▶ Set up an Amazon Pinpoint originating identity

## Password policy Info

[Edit](#)

Create a password policy to define the length and complexity of the passwords your users can set.

### Password minimum length

6 character(s)

Temporary passwords set by administrators expire in  
7 day(s)

Allow reuse of previous passwords

### Password requirements

Contains at least 1 lowercase letter

## Passkey Info

[Edit](#)

Configure sign-in with biometrics, hardware keys, and authenticator apps.

User verification  
Preferred

Domain for relying party ID  
Third-party domain

Third-party domain

## Extensions Info

### Lambda triggers (1) Info

Add and configure Lambda triggers for your user pool. Cognito can invoke Lambda functions in your a register and confirm users, authenticate users, send messages, and generate tokens.

Lambda triggers	Attached Lambda function
<input type="radio"/> Pre sign-up Lambda trigger	<a href="#">auto-confirm-user</a>

## Cloudwatch:

CloudWatch > Live Tail

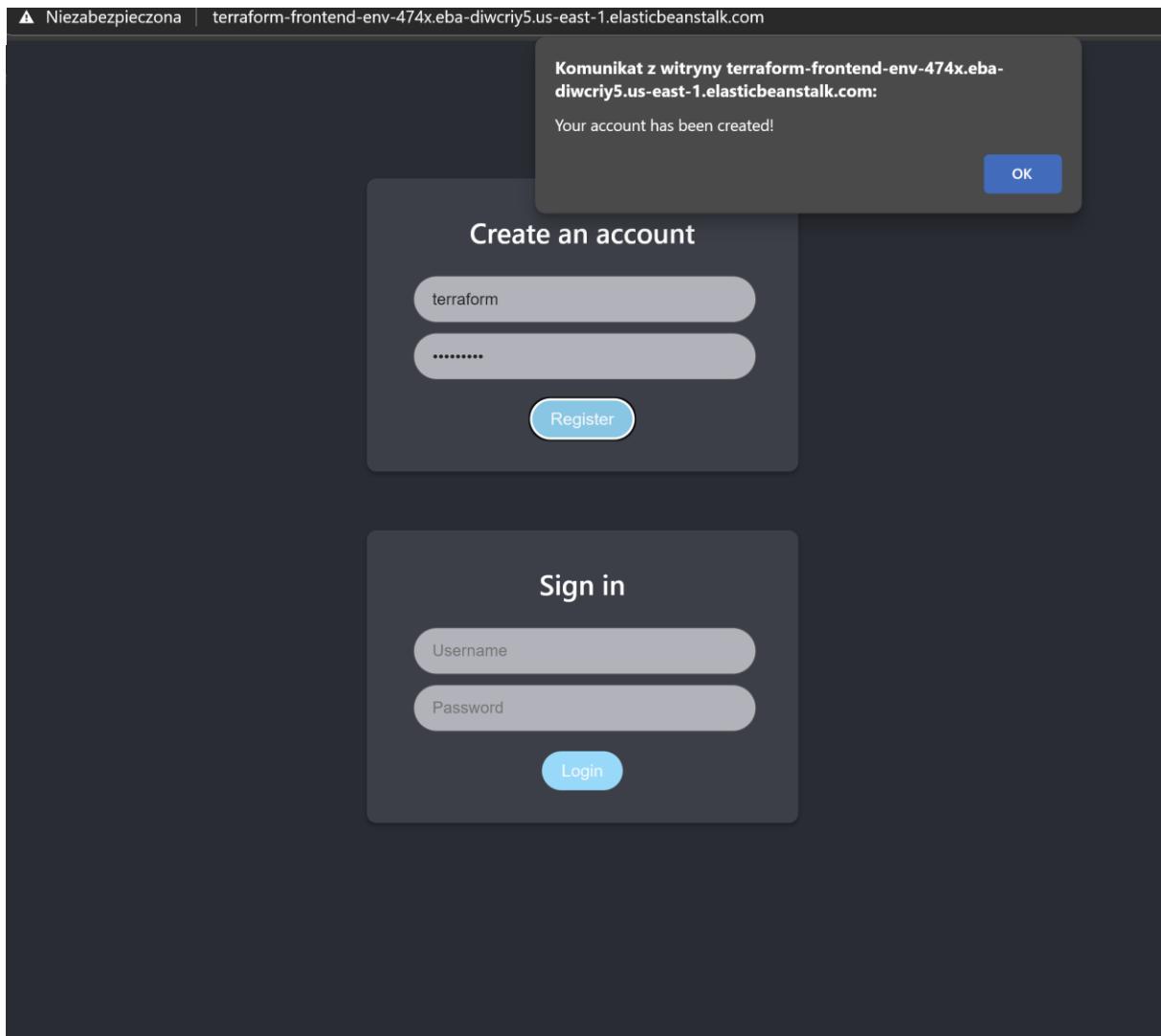
Highlight term  Highlight up to 5 terms (Not case sensitive)

Filter Actions Clear Cancel Start 0 events/sec, 100% displayed 00:04:33 View in columns

Timestamp (UTC)	Message	Log group	Log stream	
2025-04-06T23:48:48.289Z DEBUG 1	[backend] [nio-8080-exec-4] o.s.s.oauth2.jwt.JwtTimestampValidator : Jwt expired at 2025-04-06T23:16:12Z	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:48:48:52.598Z	[backend] [nio-8080-exec-4] o.s.s.o.s.r.a.JwtAuthenticationProvider : Failed to authenticate since the Jwt was invalid	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:48:48.291Z DEBUG 1	[backend] [nio-8080-exec-4] o.s.s.o.s.r.a.JwtAuthenticationProvider : Failed to authenticate since the Jwt was invalid	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:48:48:395Z	[172.31.8.160 - - [06/Apr/2025:23:48:48 +0000] "OPTIONS /api/messages HTTP/1.1" 200 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:48:48:395Z	[172.31.8.160 - - [06/Apr/2025:23:48:48 +0000] "POST /api/messages HTTP/1.1" 401 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:48:48:395Z	[172.31.8.160 - - [06/Apr/2025:23:48:48 +0000] "OPTIONS /api/messages/with-file HTTP/1.1" 200 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com..." "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:48:48:395Z	[172.31.8.160 - - [06/Apr/2025:23:48:48 +0000] "OPTIONS /api/messages/with-file HTTP/1.1" 200 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com..." "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:09.093Z	[172.31.8.160 - - [06/Apr/2025:23:49:09 +0000] "OPTIONS /api/messages/with-file HTTP/1.1" 200 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com..." "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:09.093Z	[172.31.8.160 - - [06/Apr/2025:23:49:09 +0000] "OPTIONS /api/messages/with-file HTTP/1.1" 200 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com..." "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:14.590Z	[172.31.8.160 - - [06/Apr/2025:23:49:09 +0000] "POST /api/messages/with-file HTTP/1.1" 401 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com..." "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:14.590Z	[172.31.8.160 - - [06/Apr/2025:23:49:09 +0000] "POST /api/messages/with-file HTTP/1.1" 401 0 "http://terraform-frontend-env-474x.eba-dicwriy5.us-east-1.elasticbeanstalk.com..." "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/1537.36 Edg/134.0.0.0." "109.243.146.84"]	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:10.893Z	[2025-04-06T23:49:09:627Z DEBUG 1	[backend] [nio-8080-exec-5] o.s.security.web.FilterChainProxy : Securing OPTIONS /api/messages/with-file	107378568397:/aws/elasticbe...	<a href="#">Link</a>
2025-04-06T23:49:09.627Z DEBUG 1	[backend] [nio-8080-exec-5] o.s.security.web.FilterChainProxy : Securing OPTIONS /api/messages/with-file	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:09:767Z DEBUG 1	[backend] [nio-8080-exec-6] o.s.security.web.FilterChainProxy : Securing POST /api/messages/with-file	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:09:767Z DEBUG 1	[backend] [nio-8080-exec-6] o.s.security.web.FilterChainProxy : Securing POST /api/messages/with-file	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:09:772Z DEBUG 1	[2025-04-06T23:49:09:772Z DEBUG 1	[backend] [nio-8080-exec-4] o.s.s.oauth2.JwtJwtTimestampValidator : Jwt expired at 2025-04-06T23:16:12Z	107378568397:/aws/elasticbe...	<a href="#">Link</a>
2025-04-06T23:49:09:772Z DEBUG 1	[backend] [nio-8080-exec-4] o.s.s.oauth2.JwtJwtTimestampValidator : Jwt expired at 2025-04-06T23:16:12Z	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:09:772Z DEBUG 1	[2025-04-06T23:49:09:772Z DEBUG 1	[backend] [nio-8080-exec-4] o.s.s.oauth2.JwtJwtTimestampValidator : Jwt expired at 2025-04-06T23:16:12Z	107378568397:/aws/elasticbe...	<a href="#">Link</a>
2025-04-06T23:49:09:772Z DEBUG 1	[backend] [nio-8080-exec-4] o.s.s.oauth2.JwtJwtTimestampValidator : Jwt expired at 2025-04-06T23:16:12Z	107378568397:/aws/elasticbe...	<a href="#">Link</a>	
2025-04-06T23:49:14.590Z	[2025-04-06T23:49:09:772Z DEBUG 1	[backend] [nio-8080-exec-4] o.s.s.oauth2.JwtAuthenticationProvider : Failed to authenticate since the Jwt was invalid	107378568397:/aws/elasticbe...	<a href="#">Link</a>
2025-04-06T23:49:09:772Z DEBUG 1	[backend] [nio-8080-exec-6] o.s.s.oauth2.JwtAuthenticationProvider : Failed to authenticate since the Jwt was invalid	107378568397:/aws/elasticbe...	<a href="#">Link</a>	

No to teraz sprawdźmy działanie

Spróbujmy stworzyć użytkownika terraform:



W Cognito:

**Users (7) Info**

View, edit, and create users in your user pool. Users that are enabled and confirmed can sign in to your user pool.

Property:

User name	Email address	Email verified	Confirmation status	Status
<a href="#">anita</a>	-	No	Confirmed	Enabled
<a href="#">terraform</a>	-	No	Confirmed	Enabled
<a href="#">xerito</a>	-	No	Unconfirmed	Enabled
<a href="#">xerito1</a>	-	No	Unconfirmed	Enabled
<a href="#">dawdqdwg</a>	-	No	Confirmed	Enabled
<a href="#">fansff</a>	-	No	Unconfirmed	Enabled
<a href="#">marek</a>	-	No	Confirmed	Enabled

Spróbujmy się zalogować:

The screenshot displays a mobile application interface. At the top, there is a dark-themed 'Sign in' screen with two input fields: one for 'username' containing 'terraform' and another for 'password' containing '.....'. A blue 'Login' button is positioned below the password field. Below this, the main application area shows a dark-themed messaging interface. On the left, a 'Send Message' screen is visible with fields for 'Recipient' (labeled 'Enter username') and 'Message' (labeled 'Type your message...'). Below these is an 'Attach file (optional)' section with a placeholder 'No file selected +'. At the bottom of this screen is a blue 'Send →' button. To the right, a 'Received Messages' screen shows a single message entry labeled 'Sent Messages ^'. Above the messaging screens, a small modal window titled 'Zapisz hasło?' (Save password?) is open, asking if the user wants to save their password automatically. It contains fields for 'Nazwa użytkownika' (username) set to 'terraform' and 'Hasło' (password) set to '.....'. There are two buttons at the bottom of this modal: a blue 'Zapisz' (Save) button and a grey 'Nie teraz' (Not now) button.

Zalogowaliśmy się z powodzeniem, to teraz wyślijmy jakąś wiadomość z załącznikiem do jakiegoś innego użytkownika:

The screenshot displays a mobile application interface for sending messages. On the left, there is a 'Send Message' screen and on the right, a 'Received Messages' screen.

**Send Message Screen (Left):**

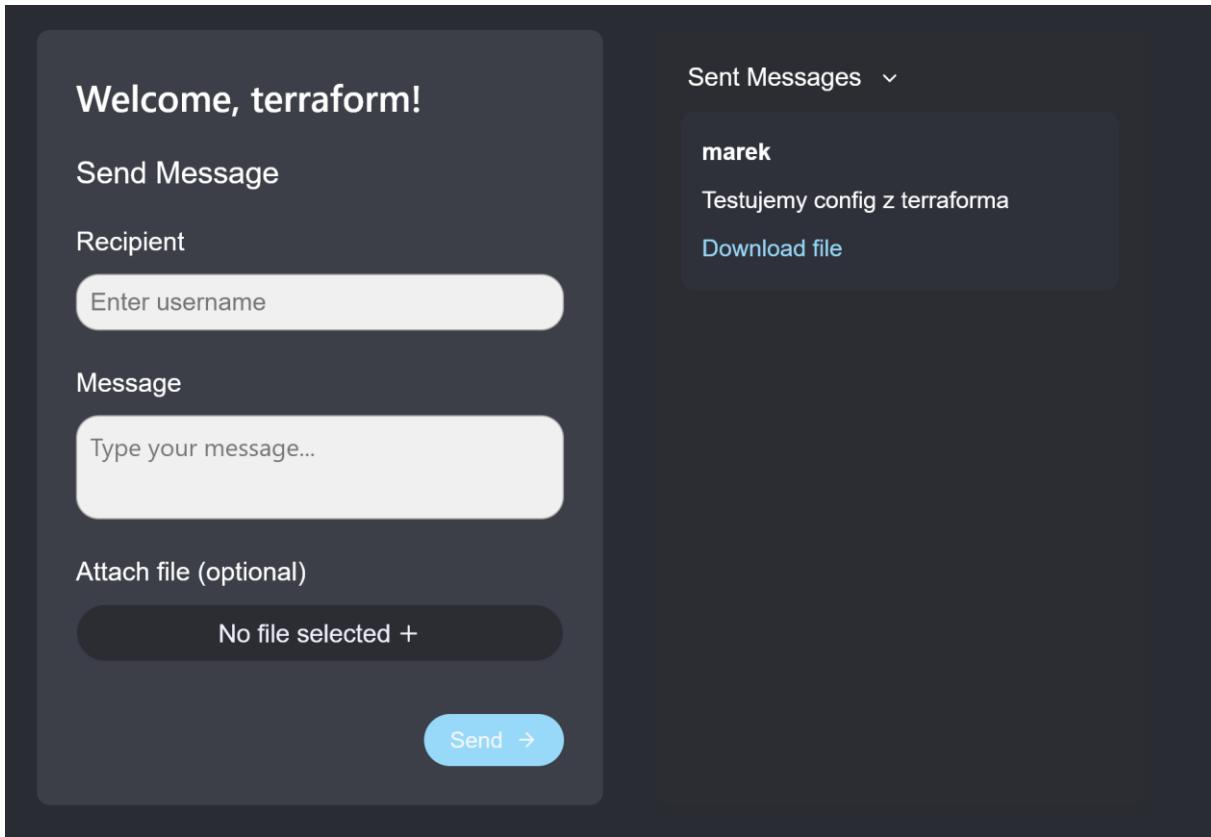
- Welcome, terraform!**
- Send Message**
- Recipient**: Input field containing `marek`.
- Message**: Input field containing `Testujemy config z terraforma`.
- Attach file (optional)**: Input field containing `HELP.md`.
- Send →** button at the bottom.

**Received Messages Screen (Right):**

- Komunikat z witryny terraform-frontend-env-474x.eba-diwcriy5.us-east-1.elasticbeanstalk.com:**
- Message with file sent successfully!**
- OK** button at the bottom right.

**Bottom Buttons:**

- Sent Messages ↗** button on the Received Messages screen.
- Send →** button on the Send Message screen.



No i rzeczywiście w wysłanych mamy tą wiadomość, spróbujmy pobrać plik u siebie:

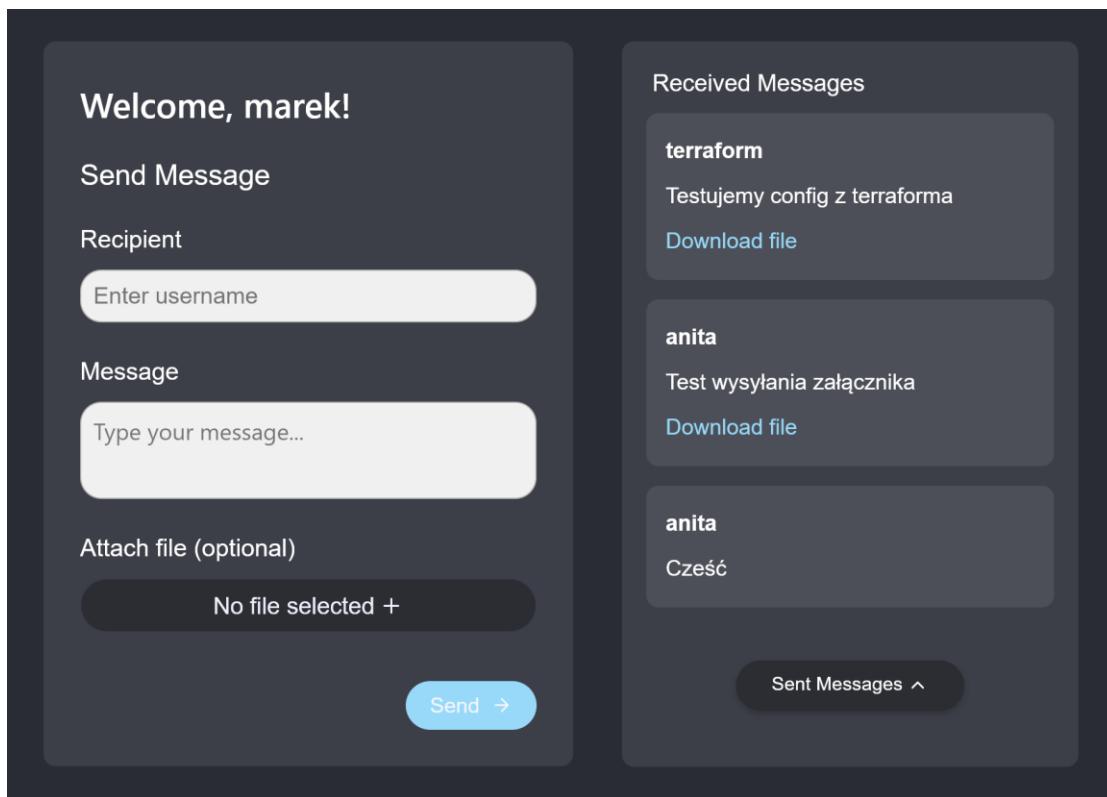
Niezabezpieczona | [terraform-frontend-env-474x.eba-diwcry5.us-east-1.elasticbeanstalk.com](https://terraform-frontend-env-474x.eba-diwcry5.us-east-1.elasticbeanstalk.com)

Pobrane

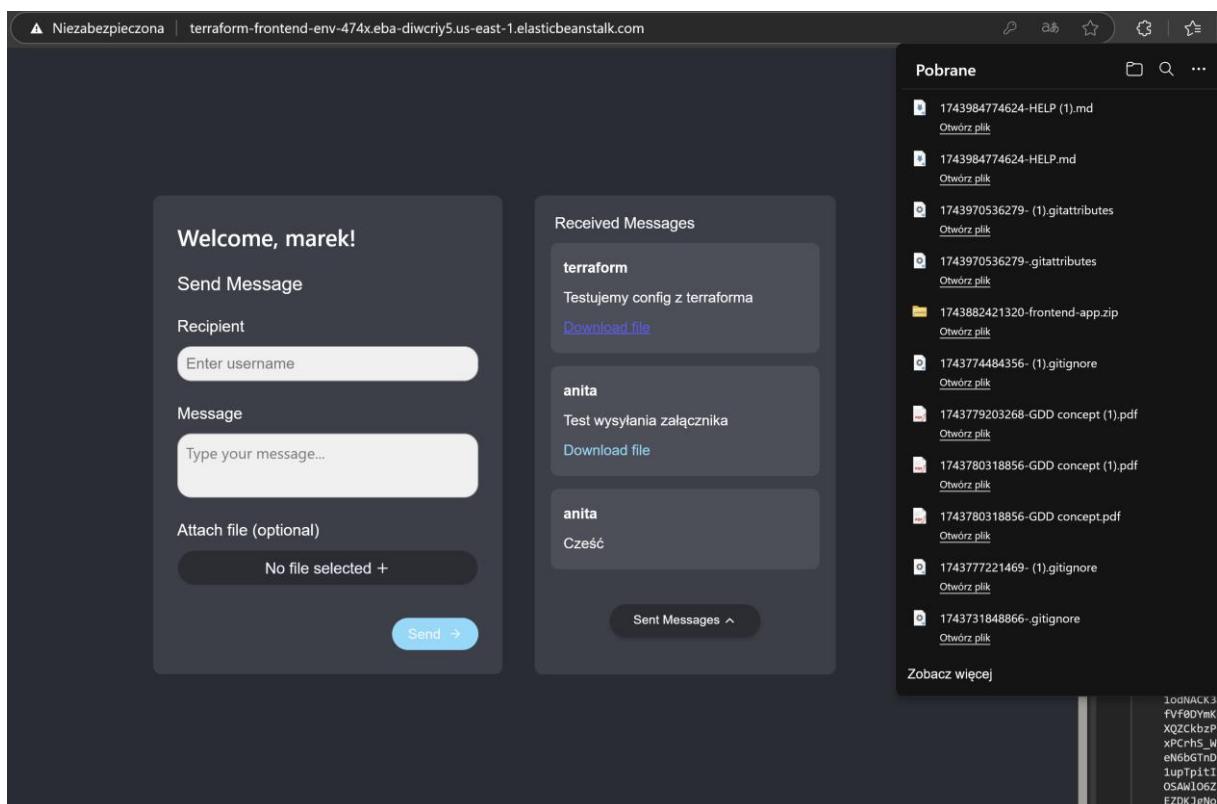
- 1743984774624-HELP.md  
Otwórz plik
- 1743970536279- (1).gitattributes  
Otwórz plik
- 1743970536279-.gitattributes  
Otwórz plik
- 1743882421320-frontend-app.zip  
Otwórz plik
- 1743774484356- (1).gitignore  
Otwórz plik
- 1743779203268-GDD concept (1).pdf  
Otwórz plik
- 1743780318856-GDD concept (1).pdf  
Otwórz plik
- 1743780318856-GDD concept.pdf  
Otwórz plik
- 1743777221469- (1).gitignore  
Otwórz plik
- 1743731848866-.gitignore  
Otwórz plik
- 1743775677111-.gitignore  
Otwórz plik

Zobacz więcej

Plik się pobiera, wszystko działa. Sprawdźmy teraz jak to wygląda u marka:



Mamy wiadomość od terraforma, to sprawdźmy jeszcze pobieranie pliku



Wszystko działa, spełniłem wszystkie założenia projektowe

Jeśli natomiast chodzi o Endpointy to mamy ich 10: 5 POST i 5 GET

Endpoint POST 1:

```
// Endpoint tworzenia wiadomości
@PostMapping(✉️) ✎ stvshy
public Message addMessage(@RequestBody Map<String, String> body) {
    String authorUsername = body.get("author");
    String content = body.get("content");
    String recipientUsername = body.get("recipient");

    logger.debug("Próba zapisu wiadomości. Nadawca: {}, treść: {}, odbiorca: {}",
                 authorUsername, content, recipientUsername);

    Message msg = new Message(authorUsername, content);
    msg.setRecipientUsername(recipientUsername);
    return messageRepository.save(msg);
}
```

Endpoint GET 1:

```
// Endpoint: wiadomości odebrane przez danego użytkownika
@GetMapping(✉️"/received") ✎ stvshy
public List<Message> getReceivedMessages(@RequestParam String username) {
    logger.debug("getReceivedMessages called with username: {}", username);
    return messageRepository.findByRecipientUsername(username);
}
```

Endpoint GET 2:

```
// Endpoint: wiadomości wysłane przez danego użytkownika
@GetMapping(✉️"/sent") ✎ stvshy
public List<Message> getSentMessages(@RequestParam String username) {
    logger.debug("getSentMessages called with username: {}", username);
    return messageRepository.findByAuthorUsername(username);
}
```

Endpoint POST 2:

```
// Endpoint do uploadu pliku pozostaje bez zmian
@PostMapping(value = "/upload", consumes = MediaType.MULTIPART_FORM_DATA_VALUE) ▲ stvshy
public String uploadFile(@RequestParam("file") MultipartFile file) throws IOException {
    File uploadsDir = new File(pathname: "uploads");
    if (!uploadsDir.exists()) {
        uploadsDir.mkdirs();
    }
    File destination = new File(uploadsDir, file.getOriginalFilename());
    file.transferTo(destination);
    return "Plik zapisany: " + destination.getAbsolutePath();
}
```

Endpoint POST 3:

```
@PostMapping(value = "/with-file", consumes = MediaType.MULTIPART_FORM_DATA_VALUE) ▲ stvshy
public Message addMessageWithFile(
    @RequestParam("author") String author,
    @RequestParam("content") String content,
    @RequestParam(value = "recipient", required = false) String recipient,
    @RequestParam("file") MultipartFile file
) throws IOException {
    // Wyślij plik do S3 i pobierz klucz (fileName)
    String s3Key = s3Service.uploadFile(file);
    Message msg = new Message(author, content);
    msg.setRecipientUsername(recipient);
    // Zapisz klucz pliku, nie pełny URL
    msg.setFile(s3Key);
    return messageRepository.save(msg);
}
```

Endpoint POST 4:

```
@PostMapping("/login") ▲ stvshy
public ResponseEntity<?> login(@RequestBody User loginData) {
    try {
        InitiateAuthResponse response = cognitoService.userLogin(loginData.getUsername(), loginData.getPassword());
        Map<String, String> tokens = new HashMap<>();
        tokens.put("idToken", response.authenticationResult().idToken());
        tokens.put("accessToken", response.authenticationResult().accessToken());
        tokens.put("refreshToken", response.authenticationResult().refreshToken());
        tokens.put("tokenType", response.authenticationResult().tokenType());
        return ResponseEntity.ok(tokens);
    } catch (Exception e) {
        e.printStackTrace();
        return ResponseEntity.status(HttpStatus.UNAUTHORIZED)
            .body("Błąd logowania: " + e.getMessage());
    }
}
```

Endpoint POST 5:

```
@PostMapping(@RequestMapping("/register")) stvshy
public ResponseEntity<?> register(@RequestBody User user) {
    try {
        // Rejestrujemy użytkownika w Cognito
        SignUpResponse response = cognitoService.signUp(user.getUsername(), user.getPassword());
        return ResponseEntity.ok().body("Rejestracja pomyślana. UserSub: " + response.getUserSub());
    } catch (Exception e) {
        // Wyświetlamy pełen stack trace w logach
        e.printStackTrace();
    }
    return ResponseEntity.status(HttpStatus.BAD_REQUEST)
        .body("Błąd rejestracji: " + e.getMessage() + "\nStackTrace: " + Arrays.toString(e.getStackTrace()));
}
```

Endpoint GET 3:

```
@GetMapping("/api/users") stvshy
public List<String> getAllUsernames() {
    return userRepository.findAll().stream()
        .map(User::getUsername)
        .collect(Collectors.toList());
}
```

Endpoint GET 4:

```
@GetMapping("/test-db") stvshy
public String testDb() {
    try (Connection conn = DriverManager.getConnection(
        Objects.requireNonNull(environment.getProperty("spring.datasource.url")),
        environment.getProperty("spring.datasource.username"),
        environment.getProperty("spring.datasource.password"))
    ) {
        return "Connected to DB: " + conn.getMetaData().getURL();
    } catch(Exception e) {
        return "Error: " + e.getMessage();
    }
}
```

Endpoint GET 5:

```
// Endpoint do pobierania pliku na podstawie ID wiadomości
@GetMapping("/download/{id}") ✎ stvshy
public ResponseEntity<?> downloadFile(@PathVariable Long id) {
    Message msg = messageRepository.findById(id)
        .orElseThrow(() -> new RuntimeException("Message not found"));

    if (msg.getFile() == null) {
        return ResponseEntity.status(HttpStatus.NOT_FOUND)
            .body("Brak pliku w wiadomości.");
    }

    // msg.getFile() zawiera teraz klucz pliku, np. "1672531234567-nazwa_pliku.jpg"
    String presignedUrl = s3Service.generatePresignedUrl(msg.getFile());

    return ResponseEntity.status(HttpStatus.FOUND)
        .location(URI.create(presignedUrl))
        .build();
}
```