

# Website Access and UI

URLs:

Production: <https://production.d1rel1z0j4hes1.amplifyapp.com/>

Dev: <https://dev.d1rel1z0j4hes1.amplifyapp.com/>

GitHub repo: <https://github.com/Abdelrahman-Elnagar/GAAS>

We have the app hosted with amplify and have developing and production to have solid business continuity.

The screenshot shows the AWS Amplify console for the 'GAAS\_New' app. On the left, there's a sidebar with options like Overview, Hosting (selected), Access control, Build notifications, Build settings, Custom domains, Custom headers and cache, Environment variables, Firewall (New), Monitoring, Previews, Rewrites and redirects, Secrets, App settings, CloudShell, and Feedback. The main area is titled 'Get to production' and includes three steps: 'Add a custom domain' (status: Enabled), 'Enable firewall protections' (status: Enabled), and 'Connect new branches' (status: Connected). Below this, there's a 'Branches' section with two entries: 'dev' (Deployed) and 'production' (Deployed). Each entry shows the domain URL, last deployment time, and last commit message. At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information and links for Privacy, Terms, and Cookie preferences.

Categories

- > Personal 0
- > Work 0
- > Shopping 0

Tasks Mangement

New task title... Personal

Task description (optional)...

No file selected Attach File Add Task

abdebrahimelnagar123 abdebrahimelnagar123@gmail.com

New Topic

Topic Name

Cancel Create Topic

# The CRUD operations for tasks and the service proof of usage

Categories

- > Personal
- > Work
- > Shopping

Tasks Mangement

this is the test task

Work

this is the description

File selected: ddd.txt Detach File Add Task

Categories

- > Personal
- > Work**
- > Shopping

Tasks Mangement

New task title...

Work

Task description (optional)...

No file selected Attach File Add Task

Work

this is the test task

Task Details

**this is the test task**

Work

Description

this is the description

Attached File

Edit Task Delete Task

abdelrahmanelnager123 abdelrahmanelnager123@gmail.com

Table: Todo-y43asnij53rghpejt66fphumbbe-NONE - Items returned (1/9)

Scan started on May 22, 2025, 18:06:48

C Actions ▾ Create item

◀ 1 ▶ ⚙

id (String)	_type...	content	createdAt	owner	updatedAt
672cc512-f970-4...	Todo	this is the test task :Person...	2025-05-22T15:03:33.7...	c06c293c-7...	2025-05-22T1...
4da8cb6a-c563-4...	Todo	wertewrt :Work :wertwer :	2025-05-22T14:03:48.9...	400ca90c-1...	2025-05-22T1...
41ed29a0-a02b-...	Todo	nergew f e  :Work :fdff :	2025-05-21T22:16:36.3...	909c89dc-4...	2025-05-21T2...
r2hdad20-faff-4...	Todo	dfhd nwrdl :Personall-l ofaf	2025-05-21T22:18:28.8	20hr394r-e	2025-05-22T1...

Amazon S3 > Buckets > amplify-d1rel1zoj4hes1-de-gaasstoragebucket84ca08f-lar3p940ytdu > files/

**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

Block Public Access settings for this account

**Storage Lens**

- Dashboards
- Storage Lens groups
- AWS Organizations settings

Feature spotlight

AWS Marketplace for S3

**files/**

**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 permissions. [Learn more](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	CC-Architecture Diagram.pdf	pdf	May 22, 2025, 17:30:52 (UTC+03:00)	117.6 KB	Standard
<input type="checkbox"/>	ddd.txt	txt	May 22, 2025, 18:03:34 (UTC+03:00)	125.0 B	Standard
<input type="checkbox"/>	MetaDR_Research_paper (1).pdf	pdf	May 22, 2025, 12:12:28 (UTC+03:00)	242.5 KB	Standard

[Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Categories

Personal

Work

Shopping

Tasks Mangement

New Task Work

Task description (optional)...

Work

This is the test task

Add Task

Task Details

This is the test task

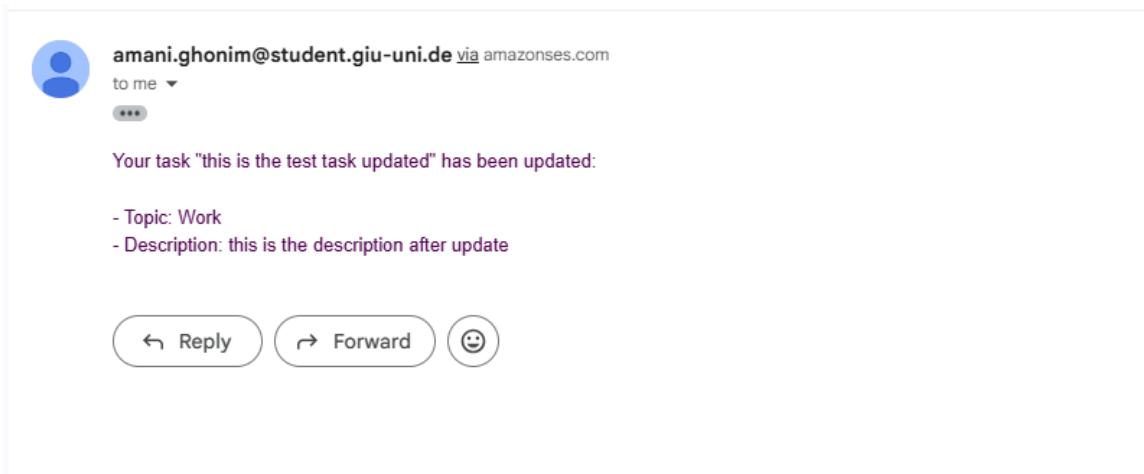
Title: this is the test task updated

Topic: Work

File: Detach Uploaded File

Description: this is the description after update

Cancel Save Changes



Amazon SNS > Topics > TaskUpdateTopic

### TaskUpdateTopic

[Edit](#) [Delete](#) [Publish message](#)

**Details**

Name	TaskUpdateTopic	Display name	-
ARN	arn:aws:sns:eu-north-1:970508865565:TaskUpdateTopic	Topic owner	970508865565
Type	Standard		

[Subscriptions](#) [Access policy](#) [Data protection policy](#) [Delivery policy \(HTTP/S\)](#) [Delivery status logging](#) [Encryption](#) [Tags](#) [Integrations](#)

**Subscriptions (1)**

ID	Endpoint	Status	Protocol
0fb175b2-7d7a-457a-a7cd-5174f1d4d070	arn:aws:sqsc:eu-north-1:970508865565:Task...	Confirmed	SQS

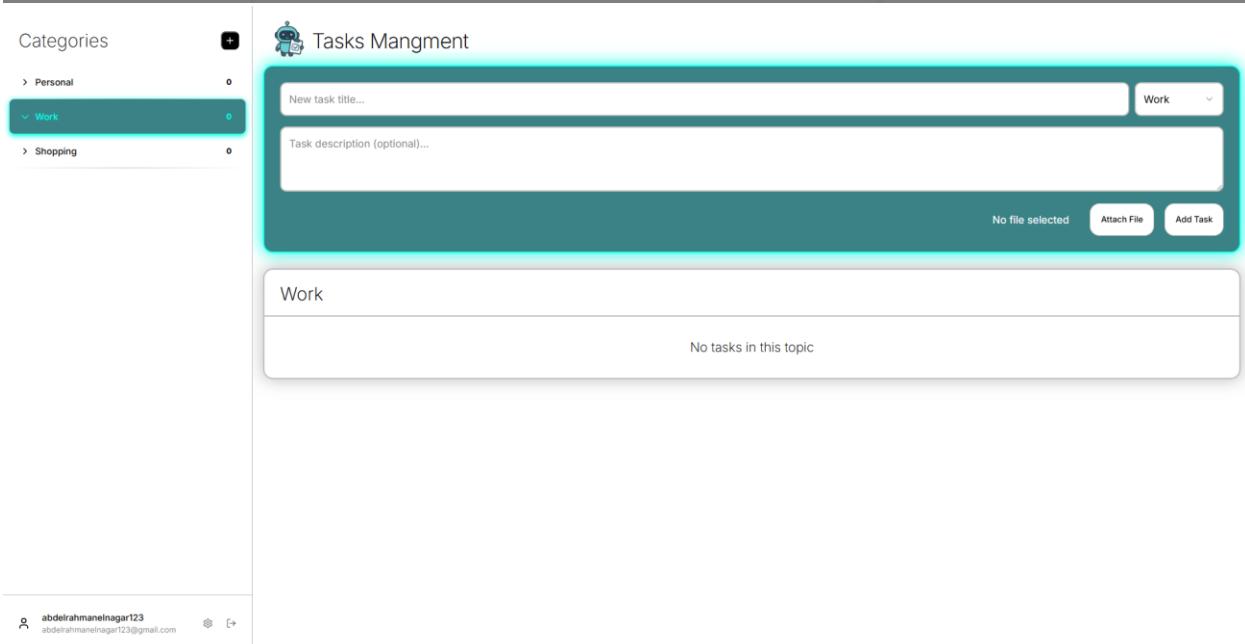
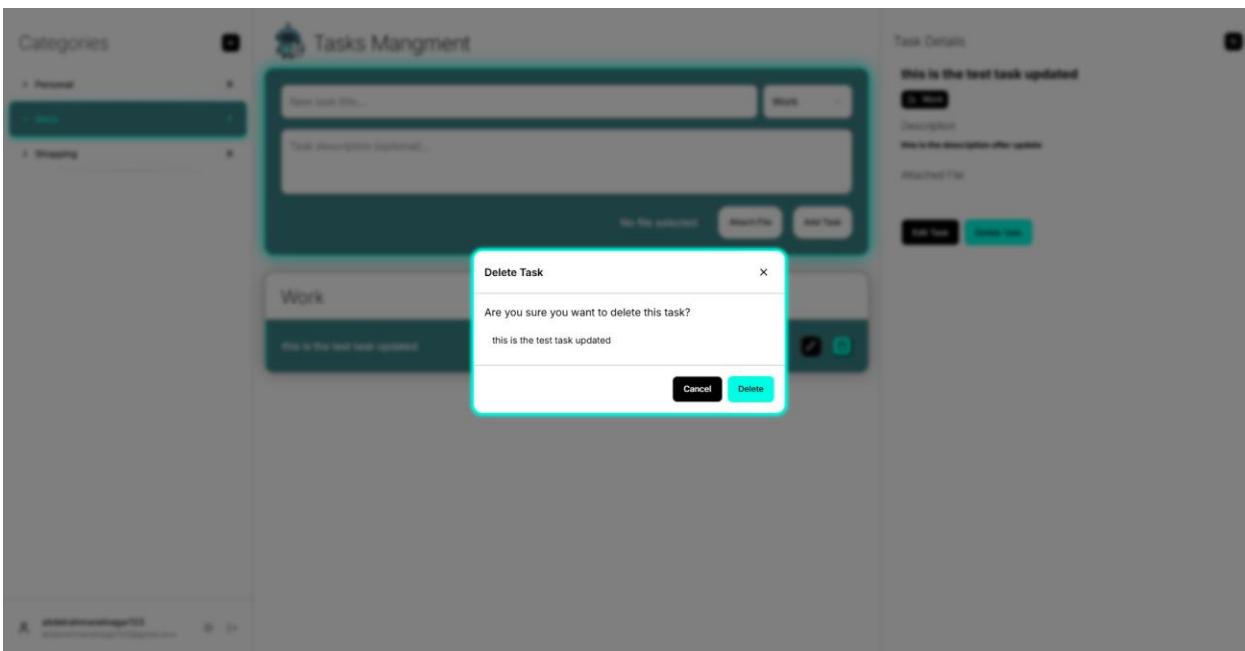
aws | [Search](#) [Alt+S]

☰ Amazon SQS > Queues

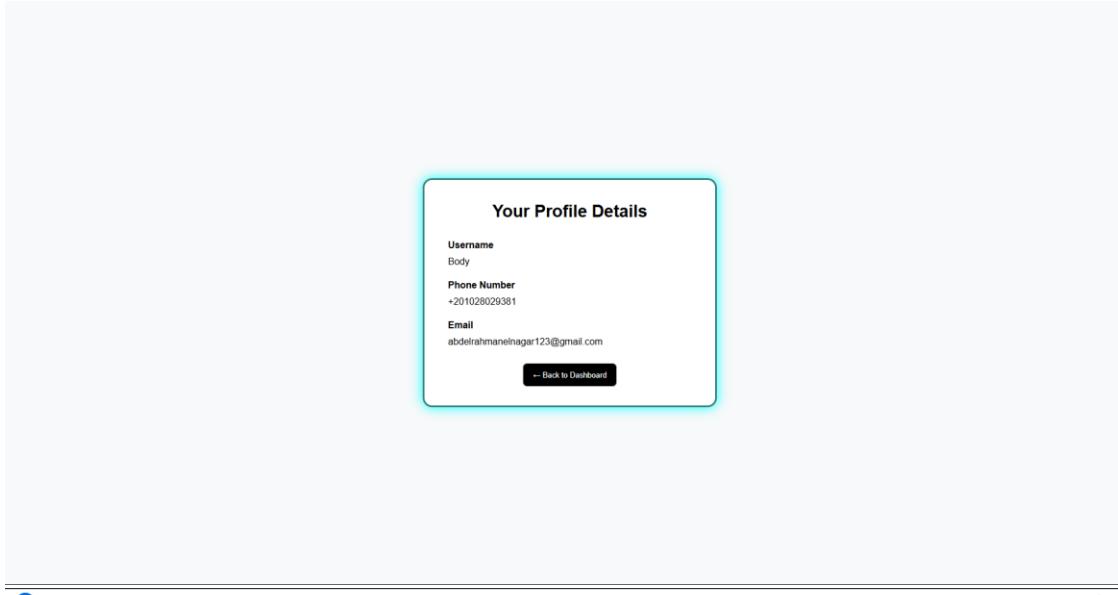
### Queues (2)

[Search queues by prefix](#)

Name	Type	Created	Messages available
<a href="#">DLQ</a>	Standard	2025-05-14T23:21+03:00	0
<a href="#">TaskUpdateQueue</a>	Standard	2025-05-14T22:05+03:00	0



# User Profile and RDS usage



The screenshot shows the 'Users' section of the Amazon Cognito User Pools console. It displays four users with the following details:

User name	Email address	Email verified	Confirmation status	Status
107ce98c-10b1-7056-6b3e-d5e3...	abdelrahman.elnagar@student.giu...	Yes	Confirmed	Enabled
400ca90c-1001-7064-c066-d2d406b55a07	ahmed.essam@student.giu.uni.de	Yes	Confirmed	Enabled
c06c293c-70d1-70c9-a8af-2cb83d95b342	abdelrahmanelnagar123@gmail.com	Yes	Confirmed	Enabled
f07c59ac-30a1-701a-2d11-4250200438b3	aamanighonim@gmail.com	Yes	Confirmed	Enabled

Below the user list is an 'Import users' section with a table showing import job status:

Job name	Status	Imported users	Skipped users	Failed users	CloudWatch logs	Created time
						No user import jobs found

The screenshot shows the AWS Lambda SQL editor interface. The query in the 'Query1' tab is:

```

9   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
10  );
11
12 • SELECT * FROM GAASUsers.users;
13

```

The 'Result Grid' shows the following data:

user_id	email	phone_number	preferredNickname	created_at
107ce98c-10b1-7056-6b3e-d5e319cbc246	abdelrahman.elnagar@student.giu.uni.de	+201028029381	Body	2025-05-22 13:30:54
400ca90c-1001-7064-c066-d2d406b55a07	ahmed.essam@student.giu.uni.de	+201020444348	Negmooo	2025-05-22 13:20:51
c06c293c-70d1-70c9-a8af-2cb83d95b342	abdelrahmanelnagar123@gmail.com	+201028029381	Body	2025-05-22 13:32:50
f07c59ac-30a1-701a-2d11-4250200438b3	aamanighonim@gmail.com	+20104404934	Amani	2025-05-22 14:29:36

The 'Output' section shows the executed query and the message '4 row(s) returned'.

[Sign In](#) [Create Account](#)

Email

Password  
 (@)

Confirm Password  
 (@)

Preferred Username

Phone Number

[Create Account](#)

[Sign In](#)      [Create Account](#)

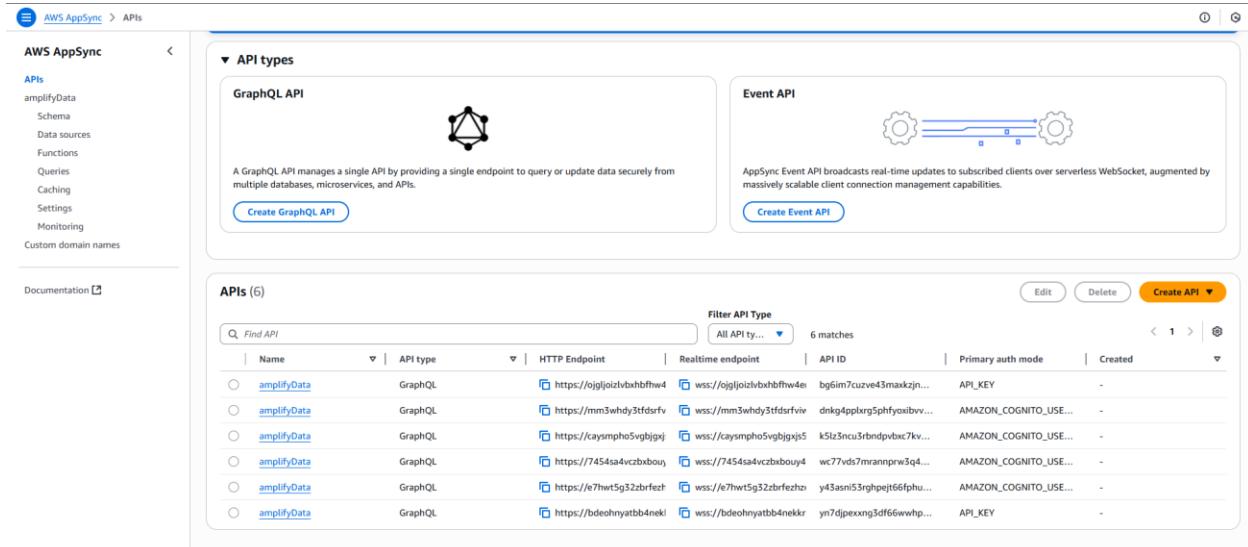
Email

Password  
 [@](#)

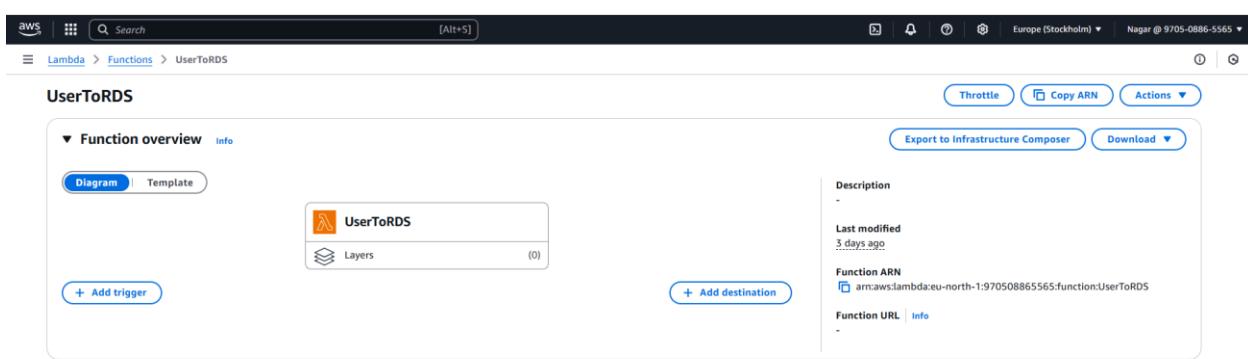
[Sign in](#)

[Forgot your password?](#)

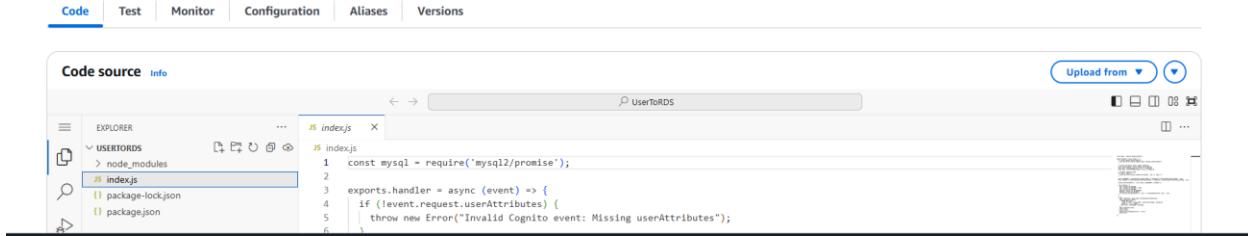
# Amplify and lambda functions:



The screenshot shows the AWS AppSync console. On the left, a sidebar titled "AWS AppSync" contains a "APIs" section with options like "amplifyData", "Schema", "Data sources", "Functions", "Queries", "Caching", "Settings", "Monitoring", and "Custom domain names". Below this is a "Documentation" link. The main area is titled "API types" and shows two options: "GraphQL API" (with a GraphQL icon) and "Event API" (with a gear and signal icon). Below these, a section titled "APIs (6)" lists six entries, each with a small thumbnail, name, type, endpoint, ID, auth mode, and creation date. A search bar at the top of this list says "Find API".

The screenshot shows the AWS Lambda console. The URL is "aws Lambda > Functions > UserToRDS". The main area is titled "UserToRDS" and shows a "Function overview" with tabs for "Diagram" (selected), "Template", "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions". It includes sections for "Description", "Last modified" (3 days ago), "Function ARN" (arn:aws:lambda:eu-north-1:970508865565:function:UserToRDS), and "Function URL". Buttons for "Throttle", "Copy ARN", "Actions", "Export to Infrastructure Composer", and "Download" are also present.

The screenshot shows the AWS Lambda code editor for the "UserToRDS" function. The tab "Code" is selected. The interface includes a file explorer on the left showing "EXPLORER", "USERTORDS", "node\_modules", and files "index.js", "package-lock.json", and "package.json". The main area shows the "index.js" file content:

```
index.js
1 const mysql = require('mysql2/promise');
2
3 exports.handler = async (event) => {
4   if (!event.request.userAttributes) {
5     throw new Error("Invalid Cognito event: Missing userAttributes");
6   }
}
```

Buttons for "Upload from" and "..." are at the top right.