

Deployment Steps

Setup Postgres DB Service

- 1. Login into [Render Dashboard](#) . Select **PostgreSQL** from **New** drop down menu.

Render

DashboardBlueprintsEnv Groups

+ New

My Workspace

Introducing workspaces: Your account is now a Hobby workspace. There are no changes to your billing or services. [Learn more](#)

Overview

Search services

Projects

Get organized with Projects

An easier way to organize your resources and collaborate with team members.

Create your first project

Learn more

Ungrouped Services

Active (4)Suspended (2)All (6)

| Service name | Status | Type | Runtime | Region | Last deployed |
|---------------------|-----------|-------------|---------------|-----------|---------------|
| climatechic-project | Deployed | Web Service | Python 3 | Frankfurt | 3 hours ago |
| Climate-Chic-DB | Available | PostgreSQL | PostgreSQL 16 | Frankfurt | 5 hours ago |

2. Specify
- a. Name of PostgresSQL instance - “Climate-Chic-DB”

b. Database – climatechicdb

c. User – vn1

d. Select nearest region – Frankfurt(EU Central)

e. Make sure Postgres version is 16

f. Select Instance Type as Free

g. Click on “Create Database” button

Render

Dashboard

Blueprints

Env Groups

+ New

irene.martin.campo91@gmail.com

▼

New PostgreSQL

View docs

Name

A unique name for your PostgreSQL instance.

Database

Optional

The PostgreSQL dbname

User

Optional

Region

Your services in the same region can communicate over a private network. You currently have services running in Frankfurt.

PostgreSQL Version

Datadog API Key

Optional

The API key to use for sending metrics to Datadog. Setting this will enable Datadog monitoring.

example-postgresql-name

randomly generated unless specified

randomly generated unless specified

Frankfurt (EU Central)

5 existing services

Deploy in a new region

16

Instance Type

For hobby projects

Free

\$0 / month

256 MB (RAM)

0.1 CPU

1 GB (Storage)

For professional use

Select an instance type for your PostgreSQL instance. Currently, we don't support downgrading PostgreSQL instances. Make sure to pick the instance type that works for you.

Starter

\$7 / month

256 MB (RAM)

0.1 CPU

1 GB (Storage)

Pro

\$95 / month

4 GB (RAM)

2 CPU

96 GB (Storage)

Upgrade to enable more features

Render will delete your free database after 30 days unless you upgrade it to a paid instance. Render also suspends free databases after 72 hours of inactivity. [Learn more](#).

Standard

\$20 / month

1 GB (RAM)

1 CPU

16 GB (Storage)

Pro Plus

\$185 / month

8 GB (RAM)

4 CPU

256 GB (Storage)

Need a custom instance type? We support up to 1024 GB RAM, 128 CPUs, and 5 TB storage.

Access more features like [Point-in-Time Recovery](#) and [High Availability](#) by upgrading to a team plan.

+ Create a team

Create Database

- PostgreSQL service will start in some time. Click on **Connect** drop down menu and from **Internal** tab copy **Internal Database URL**

The screenshot shows the Render dashboard with the 'Dashboard' tab selected. The service 'Climate-Chic-DB' is highlighted, with a 'Basic-256mb' plan. A modal window is open showing the 'Internal Database URL' for the service. The URL is: `postgresql://vn1:CzqKSBfhob9JNXa5GEr89H...`. The modal also includes a 'Connect' button and a 'View docs' link.

Setup Django Web Service

- Requirement.txt** in github repo (<https://github.com/Irenetitor/climatechic-project/blob/main/requirements.txt>) mentions package dependencies needed by render.com and build.sh (<https://github.com/Irenetitor/climatechic-project/blob/main/build.sh>) script mentions build commands before starting server


[climatechic-project / requirements.txt](#)

```
1 asgiref==3.8.1
2 certifi==2024.8.30
3 charset-normalizer==3.4.0
4 click==8.1.7
5 colorama==0.4.6
6 crispy-bootstrap5==2024.10
7 dj-database-url==2.2.0
8 Django==5.1.1
9 django-crispy-forms==2.3
10 gunicorn==23.0.0
11 h11==0.14.0
12 idna==3.10
13 packaging==24.1
14 pillow==10.4.0
15 pycogp2-binary==2.9.9
16 python-decouple==3.8
17 requests==2.32.3
18 sqlparse==0.5.1
19 typing_extensions==4.12.2
20 tzdata==2024.2
21 urllib3==2.2.3
22 uvicorn==0.31.1
23 whitenoise==6.7.0
```

[climatechic-project / build.sh](#)

```
1 #!/usr/bin/env bash
2 # Exit on error
3 set -o errexit
4
5 pip install -r requirements.txt
6
7 # Convert static asset files
8 python manage.py collectstatic --no-input
9
10 # Apply any outstanding database migrations
11 python manage.py makemigrations
12 python manage.py migrate
13
14 # Load initial data
15 python manage.py fetch_products
```

- In Render.com, after login into dashboard select **Web service** from **New** drop down menu.

 **Render**

Dashboard

Blueprints

Env Groups

+ New

My Workspace

Introducing workspaces: Your account is now a Hobby workspace. There are no changes to your account. [Learn more.](#)

Overview

Search services

Projects

+ Create new project

Project

Static Site

Web Service

Private Service

Background Worker

Cron Job

PostgreSQL



Redis

Blueprint


Invite your team

Ungrouped Services

Active (4)Suspended (2)All (6)

| <input type="checkbox"/> | Service name | Status | Type | Runtime | Region | Last deployed ↑ | |
|--------------------------|--|-------------|-------------|---------------|-----------|------------------------------|-----|
| <input type="checkbox"/> |  climatechic... | ✓ Deployed | Web Service | Python 3 | Frankfurt | 3 hours ago | ... |
| <input type="checkbox"/> |  Climate-Ch... | ✓ Available | PostgreSQL | PostgreSQL 16 | Frankfurt | 5 hours ago | ... |

3. Select option – Build and deploy from git repository and click next

 **Render**

Dashboard

Blueprints

Env Groups

Docs

Community

Help

New +

irene.martin.campo@ig...

Create a new Web Service

Connect a Git repository, or use an existing image.

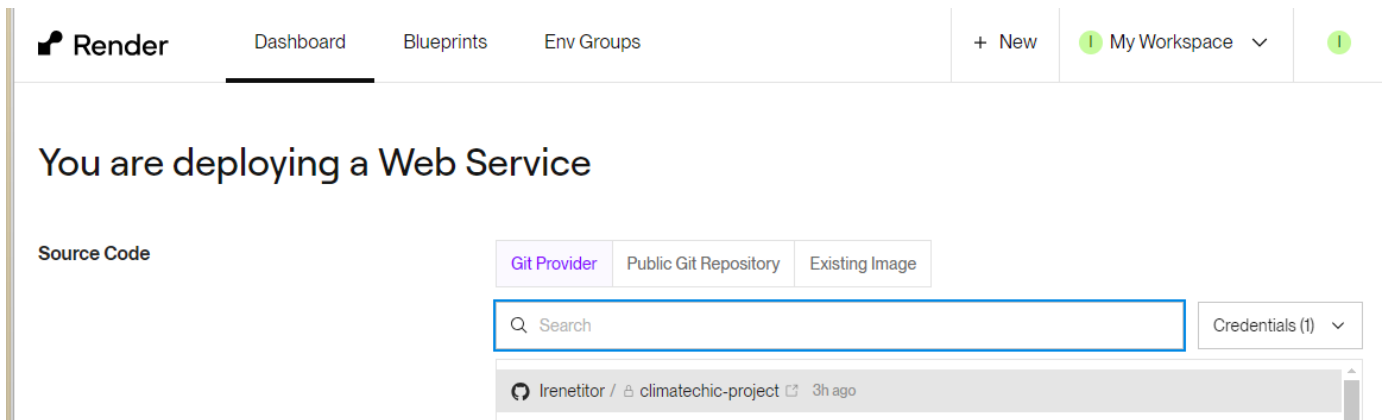
How would you like to deploy your web service?

☒ Build and deploy from a Git repository
Connect a GitHub or GitLab repository.

☐ Deploy an existing image from a registry ADVANCED
Pull a public image from any registry or a private image from Docker Hub, GitHub, or GitLab.

Next

4. Connect with gitrepo <https://github.com/Irenetitor/climatechic-project>



Render Dashboard

You are deploying a Web Service

Source Code

Git Provider Public Git Repository Existing Image

Search

Credentials (1)

Irenetitor / climatechic-project 3h ago

5. Specify

- Name for web service instance – **climatechic-project**
- Select nearest region – **Frankfurt(EU Central)**
- Branch for repo as **main**
- Keep Runtime as **Python 3**
- Build command as **\$./build.sh**
- Start command as **\$ gunicorn climatechic.asgi:application -k uvicorn.workers.UvicornWorker**
- Select Instance Type as **Free**
- IMPORTANT – Mention Environment variable DATABASE_URL as copied Internal Database URL for deployed PostgreSQL service before while setup of PostgreSQL database.**

Set Environment Variables:

```
SECRET_KEY=<Secret Keys>
WEATHER_API_KEY=<Secret API Keys>
PRODUCT_API_KEY=<Secret product API Keys>
DEBUG=True
ALLOWED_HOSTS=127.0.0.1, climatechic-project.onrender.com
DATABASE_URL=<Copied Internal DB url>
EMAIL_HOST_USER=<Admin account gmail email id>
EMAIL_HOST_PASSWORD = <Gmail App password>
```

Events

Logs

Disks

Environment

Shell

Previews

Jobs

Metrics

Scaling









Settings

Environment

Environment Variables

[↶](#) Create environment group

Set environment-specific config and secrets (such as API keys), then read those values from your code. [Learn more](#).

| Key | Value | |
|---------------------|-------|---|
| ALLOWED_HOSTS | |  |
| DATABASE_URL | |  |
| DEBUG | |  |
| EMAIL_HOST_PASSWORD | |  |
| EMAIL_HOST_USER | |  |
| PRODUCT_API_KEY | |  |
| SECRET_KEY | |  |
| WEATHER_API_KEY | |  |

- i. Click on **Create Web Service** button