

Despliegue de Aplicaciones

Índice

- 1. Introducción**
- 2. Despliegue de la BBDD en AWS**
- 3. Despliegue de la API REST en Vercel**
- 4. Despliegue de la Aplicación Angular en Vercel**

| Introducción

INTRODUCCIÓN

Despliegue de Aplicaciones

- El objetivo de este manual es detallar los diferentes pasos a seguir para desplegar una aplicación desarrollada en **Angular** que utiliza como base de datos **mysql**.
- Además el método que se va a aplicar para realizar este despliegue va a permitir también la **integración continua**. Como integración continua se entiende que cuando subimos algún cambio importante en la rama máster de nuestro repositorio automáticamente se cambie en la aplicación ya desplegada.
- Para realizar todo este proceso será necesario darse de alta en las siguientes plataformas:



INTRODUCCIÓN



Amazon Web Service

Amazon web service (aws) es una plataforma de Amazon que te permite realizar despliegue de aplicaciones, crear bases de datos y otros muchos servicios.

En este manual se utilizará **aws** para crear un servidor de bases de datos **mysql** y tener acceso desde cualquier lugar a una base de datos **MySQL**.

Podrás acceder a aws a través de este enlace: <https://aws.amazon.com/es>

INTRODUCCIÓN



GitHub

Github es un contenedor de proyectos y repositorios **GIT**.

En este manual se utilizará para contendr los repositorios de la aplicación en **Angular** y de la **API REST** que se comunica con la base de datos de **mysql**.

Podrás acceder a GitHub a través de este enlace: <https://github.com/>

INTRODUCCIÓN



Vercel

Vercel es una plataforma que funciona como hosting para el despliegue de aplicaciones.

En este manual va a ser utilizado para desplegar tanto la aplicación de **Angular** como la aplicación que implementa la **API REST** que se comunica con la base de datos MySQL.

Además se va a integrar con **GitHub** permitiendo de esta manera poder realizar integración continua.

Podrás acceder a **Vercel** a través de este enlace: <https://vercel.com/>



INTRODUCCIÓN

Fases de Despliegue

Para realizar el despliegue de nuestra aplicación completa en Angular vamos a dividir cada proceso en fases, en total tendremos 3 fases diferentes:

- Creación y Despliegue de una BBDD [MySQL](#). En esta fase usaremos la plataforma [aws](#) para crear la BBDD y desplegarla, y la aplicación [MySQLWorkbench](#) como cliente de la BBDD.
- Despliegue de la aplicación que implementa la [API REST](#) que se comunicará con la base de datos. Para esta fase utilizaremos [GitHub](#) y [Vercel](#) como plataformas y [Postman](#) para probar su funcionamiento.
- Despliegue de la aplicación [Angular](#). Para esta última fase utilizaremos [GitHub](#) y [Vercel](#), como en la fase anterior.

| Fase 1: MySQLDB

CONFIGURACIÓN

Amazon Web Service

En esta sección se va a proceder a realizar la primera fase para el despliegue de nuestra aplicación completa.

Para ello lo primero es crear una base de datos [MySQL](#) que además pueda ser accesible desde cualquier dirección IP.

Para poder realizar esta fase vamos a utilizar [aws](#) que nos proporciona servicio gratuito durante 12 meses.

En primer lugar nos debemos dar de alta en [aws](#), lo más seguro es que solicite una tarjeta de crédito, pero no va a cobrar nada porque el servicio que vamos a utilizar se encuentra dentro de la capa gratuita de [aws](#) y por lo tanto es gratuito como mínimo los primeros 12 meses.

Después de este periodo empezarán a cobrar por el trafico de las solicitudes.

CONFIGURACIÓN

Amazon Web Service

Recomendamos consultar la página de [administración de costes de aws.](#)

The screenshot shows the AWS Cost Explorer dashboard. The top navigation bar includes the AWS logo, a search bar, and account information for 'codenotch'. The left sidebar has a tree view with 'Administración de costos de AWS' expanded, showing 'Inicio' selected. Under 'Inicio', there are links for 'Cost Explorer', 'Informes', 'Budgets', 'Detección de anomalías de costos', 'Recomendaciones de redimensionamiento', 'Savings Plans' (with 'Información general', 'Inventory', 'Recomendaciones', 'Comprar Savings Plans', 'Informe de la utilización', 'Informe de la cobertura'), 'Carro' (with a count of 0), and 'Reservas' (with 'Información general'). The main content area displays 'Costos del mes actual' (0,00 US\$) and 'Costos de fin de mes previstos' (--) with a note about trends for March. A chart titled 'Costos sin combinar diarios (\$)' shows a grid from feb.-01* to mar.-09*. At the bottom, it says 'Informes a los que accedió recientemente' and 'Ver todos los informes'.

Fase 1.1 Creacion de la BBDD MySQL

FASE 1.1

Creación de la BBDD

Una vez se esté registrado y logueado aparecerás en la consola de administración de aws

The screenshot shows the AWS Management Console homepage. At the top, there's a dark header bar with the AWS logo, a search bar containing placeholder text, and user account information (codenotch, París, Soporte). Below the header, the main title "Consola de administración de AWS" is displayed. On the left, a sidebar titled "Servicios de AWS" lists recently visited services like AWS Cost Explorer, EC2, RDS, and VPC, along with a link to "Todos los servicios". To the right of the sidebar, there's a section titled "Manténgase conectado a sus recursos de AWS en cualquier lugar" which includes a mobile phone icon and a link to download the AWS mobile app. Further down, there are sections for "Crear una solución", "Explorar AWS" (with links to Amazon Redshift and AWS Fargate), and "Ejecutar una máquina virtual", "Diseñar una aplicación web", and "Diseñar utilizando servidores virtuales". Each of these sections provides a brief description and a time estimate for completion.

Servicios de AWS

▼ Servicios visitados recientemente

- AWS Cost Explorer
- EC2
- RDS
- VPC

▶ Todos los servicios

Crear una solución

Comience a usar asistentes sencillos y flujos de trabajo automatizados.

Ejecute una máquina virtual

Con EC2
De 2 a 3 minutos

Diseñe una aplicación web

Con Elastic Beanstalk
6 minutos

Diseñe utilizando servidores virtuales

Con Lightsail
De 1 a 2 minutos

Manténgase conectado a sus recursos de AWS en cualquier lugar

Descargue la aplicación móvil de la consola de AWS en su dispositivo móvil iOS o Android. [Más información](#)

Explorar AWS

Amazon Redshift

Almacén de datos rápido, sencillo y rentable que permite ampliar las consultas a su lago de datos. [Más información](#)

Ejecute contenedores sin servidor con AWS Fargate

FASE 1.1

Creación de la BBDD

Aquí hay que desplegar la pestañas servicios arriba a la izquierda y pinchar en la sección **Bases de Datos, RDS (Relational Database Service)**

The screenshot shows the AWS Management Console navigation bar. On the left, there's a sidebar with 'Favoritos' (Resource Groups & Tag Editor) and 'Visitados recientemente' (Página de inicio de la consola, AWS Cost Explorer, RDS, EC2, VPC). The main area is a dropdown menu titled 'Todos los servicios'. It is organized into several sections: 'Informática' (EC2, Lightsail, Lambda, Batch, Elastic Beanstalk, Serverless Application Repos...), 'Almacenamiento' (S3, EFS, FSx, S3 Glacier, Storage Gateway, AWS Backup), 'Base de datos' (RDS, ElastiCache), 'Habilitación para clientes' (AWS IQ, Support, Managed Services, Activate for Startups), 'Cadena de bloques' (Amazon Managed Blockchain), 'Satélite' (Ground Station), 'Quantum Technologies' (Amazon Braket), 'Administración y gobierno' (AWS Organizations, CloudWatch, AWS Auto Scaling, CloudFormation), 'Machine Learning' (Amazon SageMaker, Amazon Augmented AI, Amazon CodeGuru, Amazon DevOps Guru, Amazon Comprehend, Amazon Forecast, Amazon Fraud Detector, Amazon Kendra, Amazon Lex, Amazon Personalize, Amazon Polly, Amazon Rekognition, Amazon Textract, Amazon Transcribe, Amazon Translate, AWS DeepComposer, AWS DeepLens, AWS DeepRacer, AWS Panorama, AWS Panorama, Amazon Monitron, Amazon HealthLake), 'Móvil' (AWS Amplify, Mobile Hub, AWS AppSync, Device Farm, Amazon Location Service), 'RA y RV' (Amazon Sumerian), 'Integración de aplicaciones' (Step Functions, Amazon AppFlow, Amazon EventBridge, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF, Apache Airflow administrado), and 'Administración de costos de AWS' (AWS Cost Explorer). A search bar at the top right says 'Buscar servicios, características, productos del Marketplace y docum [Opción+S]'. The top right also shows the codenotch logo, Paris, and Soporte.

FASE 1.1

Creación de la BBDD

Pasareis a la siguiente página.

The screenshot shows the AWS RDS (Amazon Relational Database Service) dashboard for the EU (Paris) region. On the left, a sidebar menu lists various RDS resources: Dashboard, Databases, Query Editor, Performance Insights, Snapshots, Automated backups, Reserved instances, Subnet groups, Parameter groups, Option groups, Events, Event subscriptions, Recommendations (1), and Certificate update. A modal window titled "Amazon Aurora" is open, providing information about the service and a "Create database" button. The main content area displays "Resources" usage statistics, including DB Instances (1/40), DB Clusters (0/40), Reserved instances (0/40), Snapshots (1), and Event subscriptions (0/20). It also shows Parameter groups (1), Option groups (1), Subnet groups (1/50), and Supported platforms (VPC). A "Default network" entry is listed as vpc-52fb073a. To the right, a "Recommended for you" section lists "RDS Read replicas", "RDS Multi-AZ", and "Build a Serverless Database". The top navigation bar includes the AWS logo, a search bar, and user account information for codenotch, París, and Soporte.

FASE 1.1

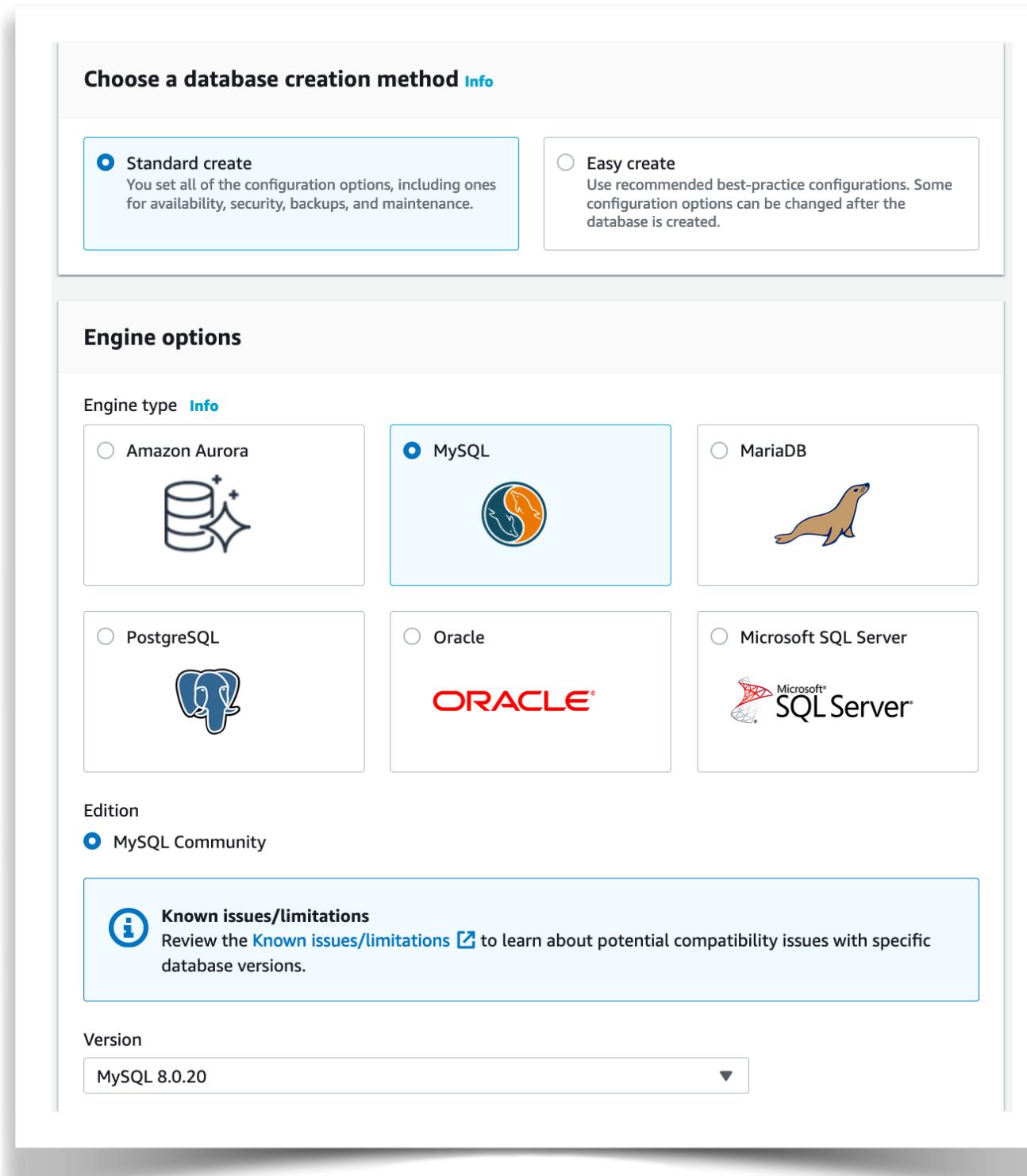
Creación de la BBDD

Hacer scroll hasta llegar a la sección de crear base de datos. No a la de Amazon Aurora que es de pago. Si no a la que aparece en la siguiente imagen.

The screenshot shows the AWS RDS (Amazon Relational Database Service) interface. On the left, there's a sidebar with various navigation options like Dashboard, Databases, Query Editor, etc. The main content area has a heading 'Create database' with a sub-section 'Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database in the cloud.' It features two buttons: 'Restore from S3' (gray) and 'Create database' (orange). Below this, a note says 'Note: your DB instances will launch in the EU (Paris) region'. To the right, there's a 'Service health' section showing 'Current status' for 'Amazon Relational Database Service (Paris)' which is 'Service is operating normally'. There's also a 'View service health dashboard' button. On the far right, there's a sidebar with links like 'Learn more', 'Database Performance Tuning', 'Additional information' (with links to 'Getting started with RDS', 'Overview and features', 'Documentation', etc.), and 'Database Preview Environment'.

FASE 1.1

Creación de la BBDD



Pinchar en “**Create database**” y elegir
“**Standard Create**” y MySQL de opciones

FASE 1.1

Creación de la BBDD

Templates
Choose a sample template to meet your use case.

Production
Use defaults for high availability and fast, consistent performance.

Dev/Test
This instance is intended for development use outside of a production environment.

Free tier
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.
[Info](#)

Settings

DB instance identifier [Info](#)
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
angular

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Credentials Settings

Master username [Info](#)
Type a login ID for the master user of your DB instance.
admin

1 to 16 alphanumeric characters. First character must be a letter

Auto generate a password
Amazon RDS can generate a password for you, or you can specify your own password

Master password [Info](#)
.....

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm password [Info](#)
.....

Elegir el **template gratuito**, un nombre para la base de datos, el nombre de **usuario** y **password**

FASE 1.1

Creación de la BBDD

DB instance class

DB instance class [Info](#)
Choose a DB instance class that meets your processing power and memory requirements. The DB instance class options below are limited to those supported by the engine you selected above.

Standard classes (includes m classes)
 Memory optimized classes (includes r and x classes)
 Burstable classes (includes t classes)

db.t2.micro
1 vCPUs 1 GiB RAM Not EBS Optimized

Include previous generation classes

Storage

Storage type [Info](#)
General Purpose (SSD)

Allocated storage
20 GiB
(Minimum: 20 GiB, Maximum: 16.384 GiB) Higher allocated storage [may improve](#) IOPS performance.

Storage autoscaling [Info](#)
Provides dynamic scaling support for your database's storage based on your application's needs.

Enable storage autoscaling
Enabling this feature will allow the storage to increase once the specified threshold is exceeded.

Maximum storage threshold [Info](#)
Charges will apply when your database autoscales to the specified threshold
1000 GiB
Minimum: 21 GiB, Maximum: 16.384 GiB

La dos siguientes secciones hay que dejar los valores por defecto

FASE 1.1

Creación de la BBDD

Connectivity

Virtual private cloud (VPC) [Info](#)
VPC that defines the virtual networking environment for this DB instance.

Default VPC (vpc-52fb073a)

Only VPCs with a corresponding DB subnet group are listed.

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

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

default-vpc-52fb073a

Public access [Info](#)

Yes
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

No
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group
Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

Choose existing
Choose existing VPC security groups

Create new
Create new VPC security group

Existing VPC security groups

Choose VPC security groups

default X

Availability Zone [Info](#)

No preference

▶ Additional configuration

En la sección de conectividad marcar Yes en “Public access”

FASE 1.1

Creación de la BBDD

Es importante marcar esta opción porque es la que va permitir que la conexión con la base de datos sea pública y por tanto el acceso a ella desde fuera de la [aws](#).

El resto de opciones hay que dejarlas según están y por lo tanto solo queda pulsar el botón [Create Database](#).

Una vez pulsado dicho botón aparecerá una pantalla como la siguiente:

The screenshot shows the AWS RDS Databases console. The top navigation bar has 'RDS' and 'Databases'. Below it, there's a toolbar with 'Databases' (selected), 'Group resources' (switched off), 'Modify', 'Actions', 'Restore from S3', and a prominent orange 'Create database' button. There's also a search bar labeled 'Filter databases' and some pagination controls. The main table lists one database: 'angular'. The columns include 'DB identifier', 'Role', 'Engine', 'Region & AZ', 'Size', 'Status', 'CPU', 'Current activity', 'Maintenance', and 'VP'. The 'Status' column shows 'Starting'.

DB identifier	Role	Engine	Region & AZ	Size	Status	CPU	Current activity	Maintenance	VP
angular	Instance	MySQL Community	eu-west-3c	db.t2.micro	Starting	-		none	vp

FASE 1.1

Creación de la BBDD

A través de la pantalla anterior se podrá acceder a toda la información de configuración de la base de datos creada pulsando sobre su nombre.

Vamos a necesitar acceder a esta página para obtener la dirección pública desde donde se accede a la base de datos y también para realizar algunos cambios mínimos de configuración de seguridad.

Esta página también se utiliza para realizar diferentes acciones sobre nuestra instancia.

FASE 1.1

Creación de la BBDD

Una vez esté creada nuestra base de datos pulsamos sobre su nombre y aparece la siguiente pantalla:

The screenshot shows the AWS RDS 'Summary' page for a MySQL Community database named 'angular'. The instance is currently 'Backing-up'. It has a CPU usage of 2.79% and 0 connections. The 'Actions' dropdown menu is open, showing options like Stop, Reboot, Delete, Create read replica, Promote, Take snapshot, Restore to point in time, and Start database activity stream.

Summary

DB identifier angular	CPU 2.79%	Status Backing-up	Class db.t2.micro
Role Instance	Current activity 0 Connections	Engine MySQL Community	Region & AZ eu-west-3c

Actions ▲

- Stop
- Reboot
- Delete
- Create read replica
- Promote
- Take snapshot
- Restore to point in time
- Start database activity stream

Connectivity & security

Endpoint & port Endpoint angular.cymgr1wrxrnd.eu-west-3.rds.amazonaws.com Port 3306	Networking Availability zone eu-west-3c VPC vpc-52fb073a Subnet group default-vpc-52fb073a Subnets subnet-abd3c8c2 subnet-34602779 subnet-93c6f4e8	Security VPC security groups default (sg-87fcdee2) (active) Public accessibility Yes Certificate authority rds-ca-2019 Certificate authority date Aug 22nd, 2024
--	---	--

FASE 1.1

Creación de la BBDD

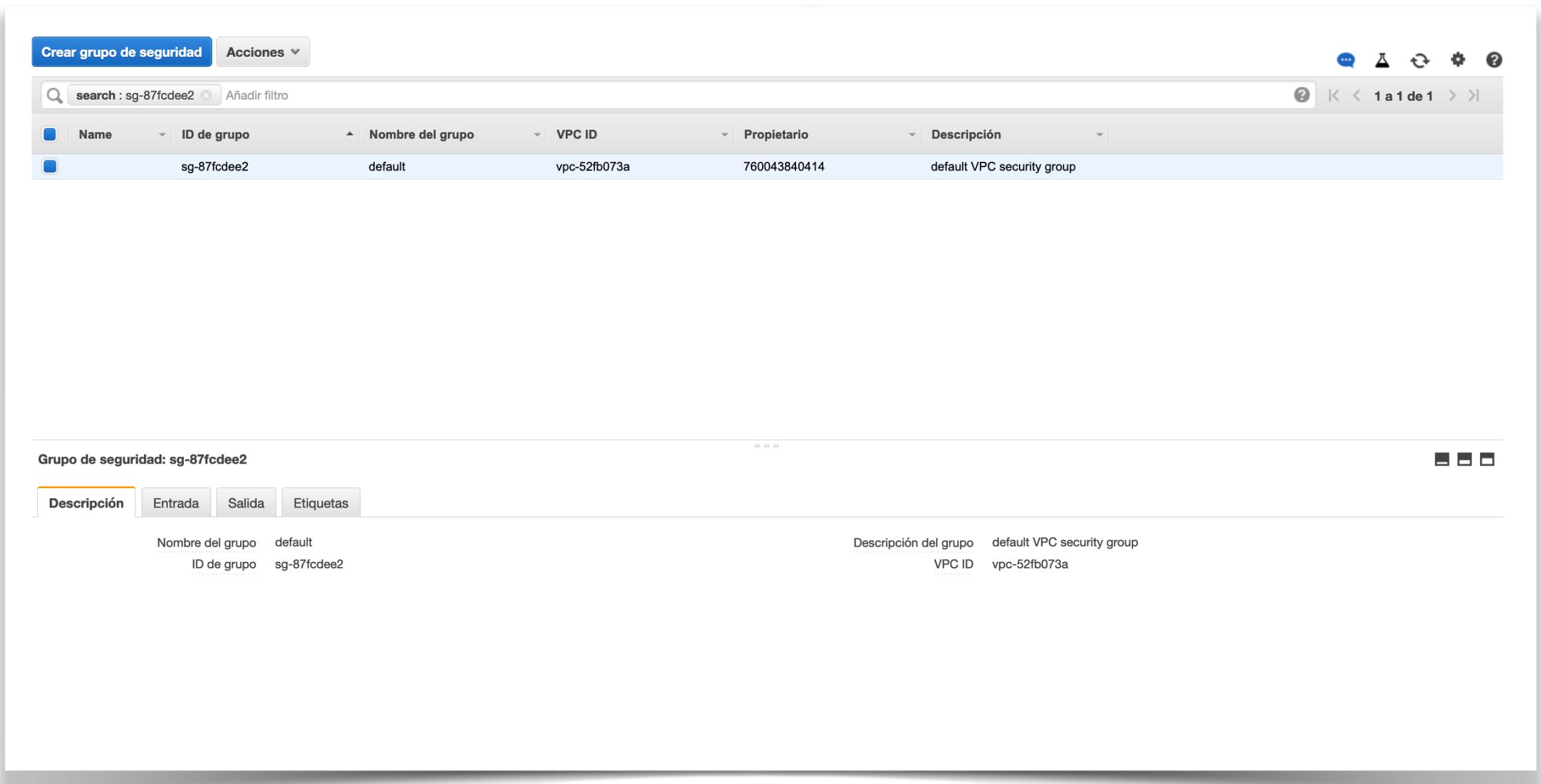
El botón superior derecho denominado **Actions** nos va a servir para realizar diferentes acciones sobre nuestra instancia de base de datos como, borrarla, pararla, reiniciarla etc. La tabla de **Connectivity & security** nos indica el endpoint donde va a estar alojada nuestra bbdd y el puerto. Dichos datos junto con el usuario y clave nos permitirán acceder a través de un cliente de bbdd a nuestra base de datos.

The screenshot shows the AWS RDS console for the 'angular' database instance. The top navigation bar includes 'Actions' (highlighted in blue), 'Stop', 'Reboot', 'Delete', 'Create read replica', 'Promote', 'Take snapshot', 'Restore to point in time', and 'Start database activity stream'. Below this, the 'Summary' section provides basic information: DB identifier (angular), CPU usage (2.79%), Status (Backing-up), Class (db.t2.micro), Role (Instance), Current activity (0 Connections), Engine (MySQL Community), Region & AZ (eu-west-3c). The 'Connectivity & security' tab is selected, showing the Endpoint & port (Endpoint: angular.cymgr1wrxnd.eu-west-3.rds.amazonaws.com, Port: 3306) and Networking (Availability zone: eu-west-3c, VPC: vpc-52fb073a, Subnet group: default-vpc-52fb073a, Subnets: subnet-abd3c8c2, subnet-34602779, subnet-93c6f4e8). The 'Security' section is highlighted with a green box and shows VPC security groups (default (sg-87fcdee2) (active)). Other tabs include 'Monitoring', 'Logs & events', 'Configuration', 'Maintenance & backups', and 'Tags'.

FASE 1.1

Creación de la BBDD

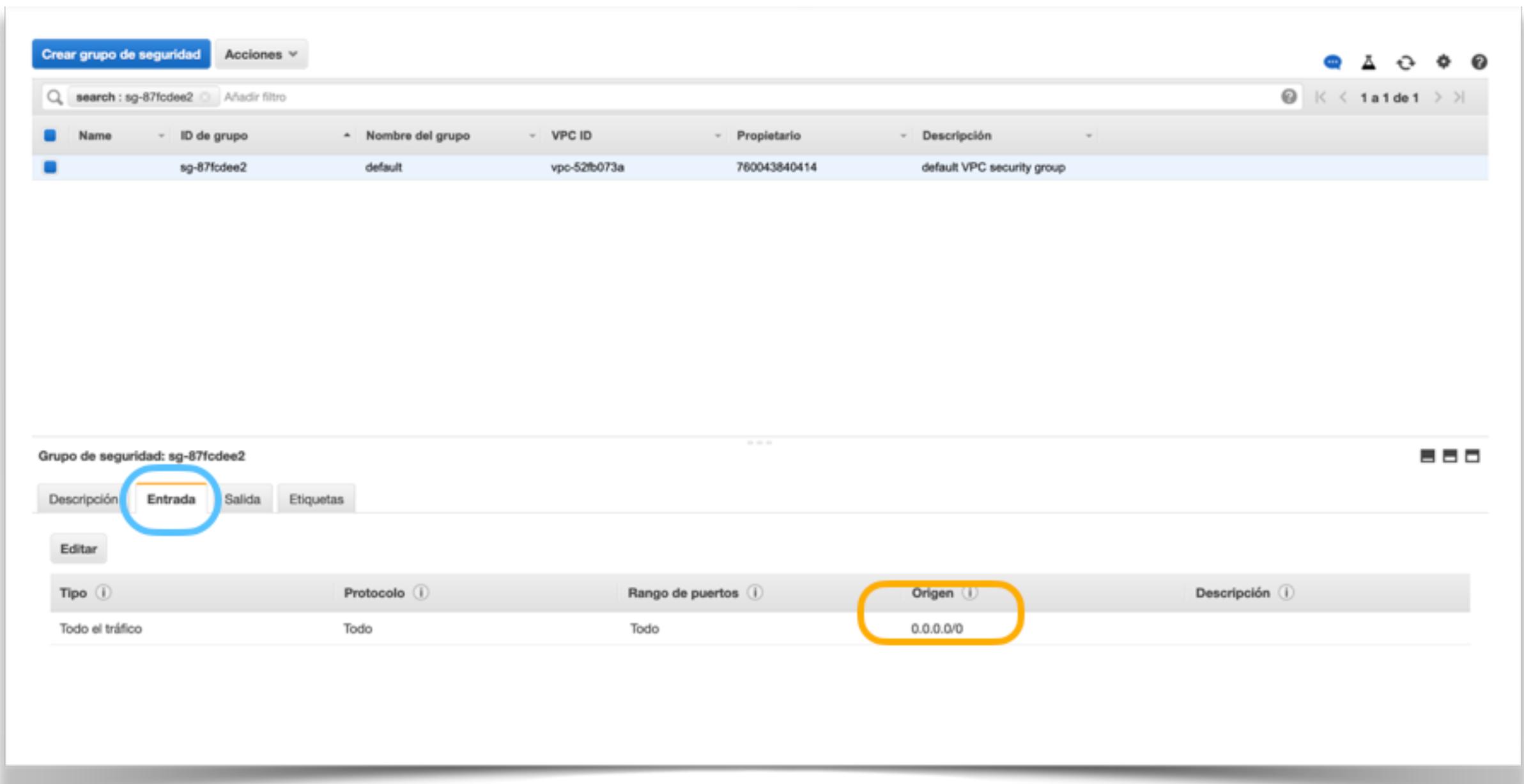
Accede a la sección de **grupos VPC (recuadro verde)** para cambiar los ajustes de seguridad y permitir el tráfico entrante desde cualquier dirección IP a nuestra BBDD. Al pinchar sobre el enlace aparece la siguiente pantalla:



FASE 1.1

Creación de la BBDD

Pinchamos sobre la pestaña de Entrada (**recuadro azul**) y cambiamos la dirección Origen a 0.0.0.0/0 (**recuadro naranja**). Para permitir todas las conexiones entrantes desde cualquier dirección IP origen.



Fase 1.2 Acceso a la BBDD

FASE 1.2

Acceso la BBDD

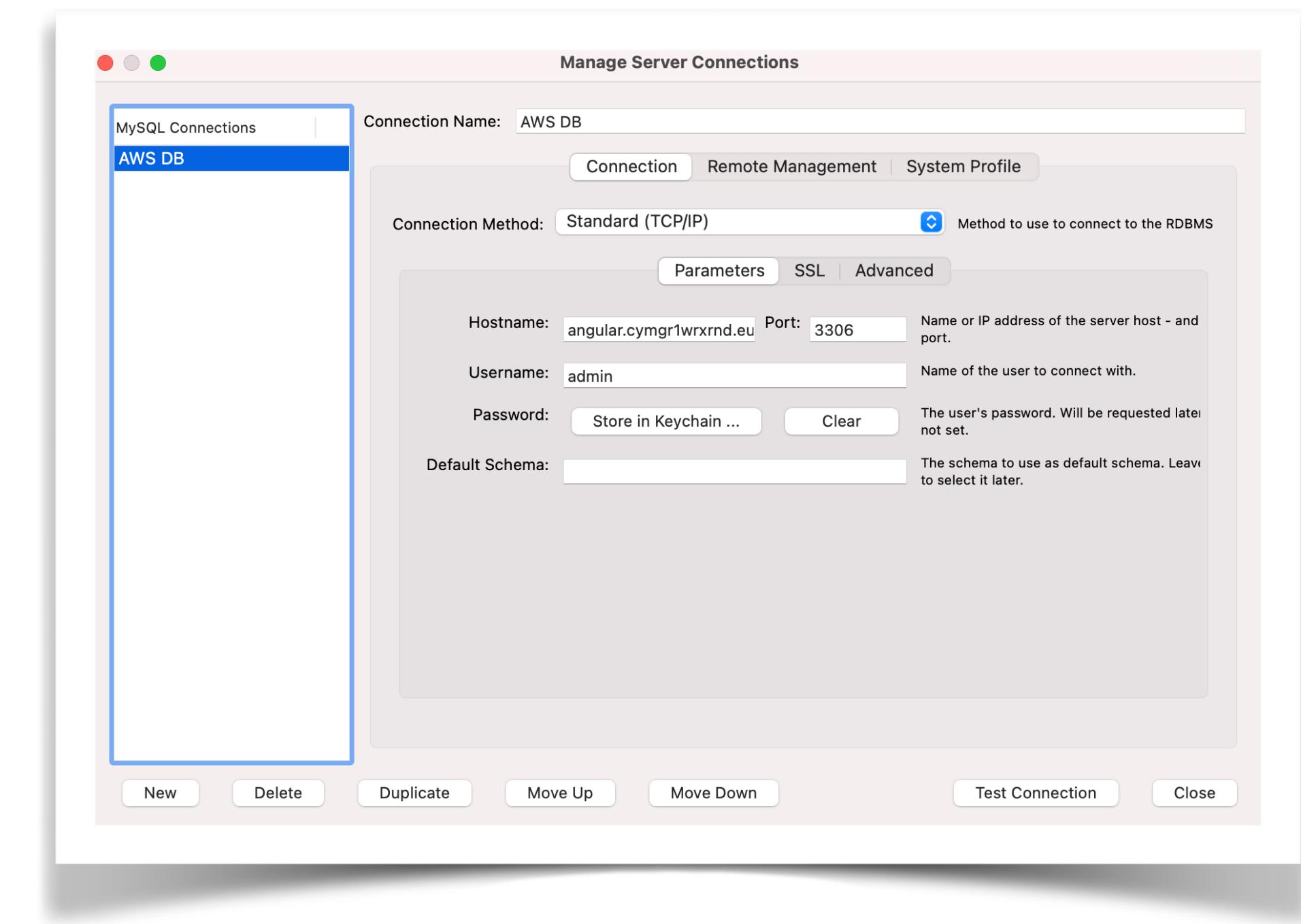
Para realizar el acceso a nuestra bbdd vamos a utilizar como cliente [MySQL Workbench](#)

Debes pulsar sobre el símbolo + en [MySQL Connections](#) para añadir una nueva conexión y entonces podrás meter los datos de la conexión.

Es importante cómo Hostname poner el [endpoint](#) que os indica [aws](#) en el apartado conexión y como [Username admin](#).

Desde aquí podéis hacer un test de conexión o conectaros directamente.

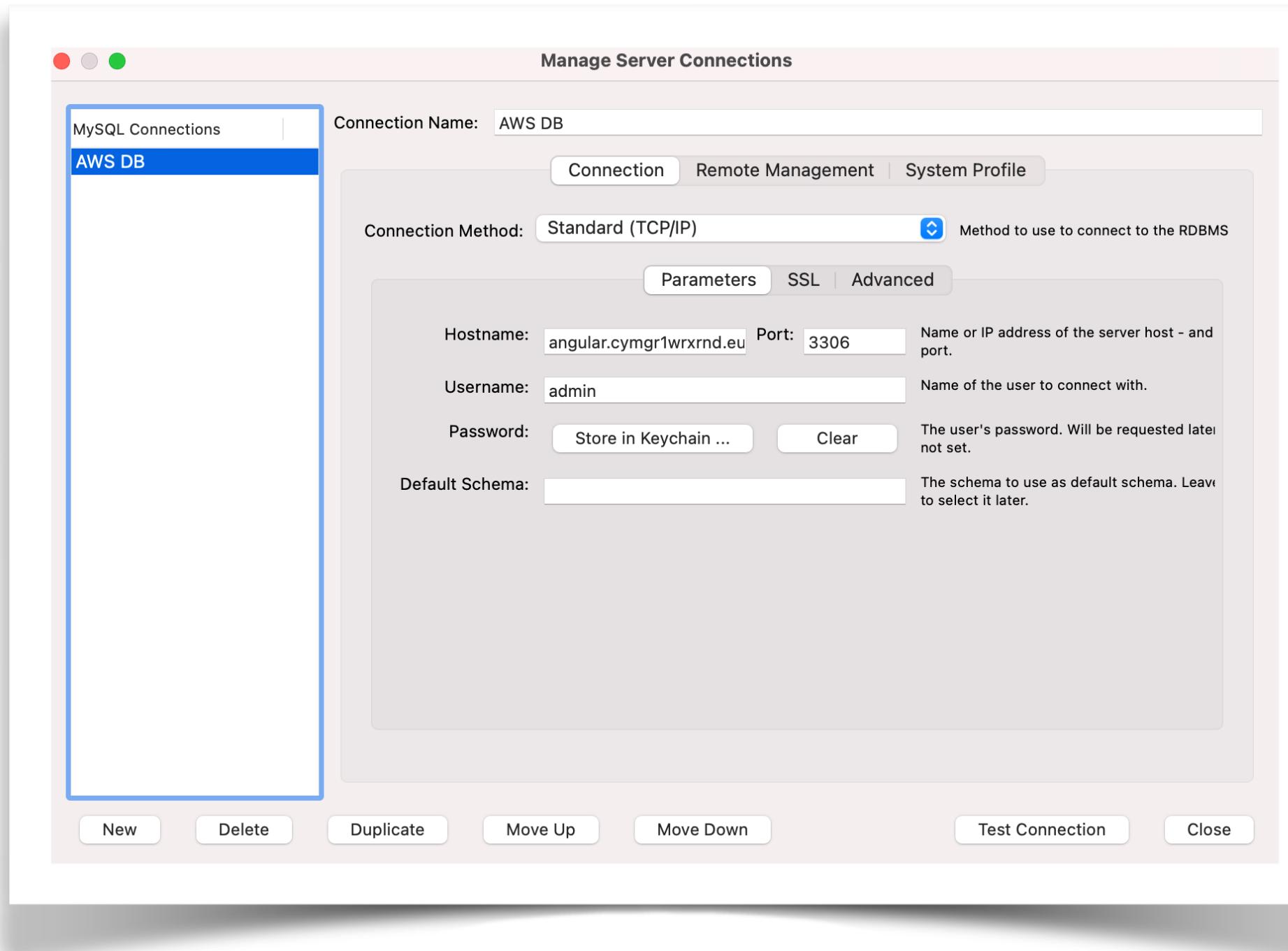
En cualquiera de los dos casos os pedirá la clave de acceso a la base de datos.



FASE 1.2

Creación de la BBDD

Si todo ha ido correctamente aparecerá una pantalla como la siguiente:



Llegados a este punto, ya tendrás la BBDD desplegada, y podrás manipularla desde [Workbench](#).

IMPORTANTE: Los datos introducidos en el punto anterior serán los valores de las **variables de entorno** que usarás en el despliegue de la [API REST](#).

| Fase 2: API REST

INTRODUCCIÓN

Despliegue de la API REST

Una vez realizado el primer paso, consistente en crear una **bbdd MySQL** plenamente funcional y **accesible desde cualquier punto**, el siguiente paso consiste en desplegar una **API REST** para que interactúe con esta bbdd.

Dicha API REST deberá por tanto ser accesible también desde cualquier sitio y plenamente pública.

Para realizar dicho despliegue vamos a hacer uso de la plataforma Vercel.

Por lo que es importante darse de alta en dicha plataforma antes de continuar.

Además el uso de esta plataforma junto con GitHub nos va a permitir realizar la **integración continua**. Como se ha comentado previamente la **integración continua** nos permite que nuestra aplicación desplegada se actualice automáticamente cuando se suba cambios en la rama principal de nuestro repositorio de GitHub.

INTRODUCCIÓN

Depliegue de la API REST

Por lo tanto los pasos que vamos a seguir en esta fase son los siguientes:

- Preparación de la [API REST](#) para subirla a [Vercel](#)
- Crear una aplicación en [Vercel](#) y asignarle [variables de entorno](#) que se enlace con nuestra API REST a través de [GitHub](#).
- Comprobar el dominio.
- Probarla con [Postman](#).

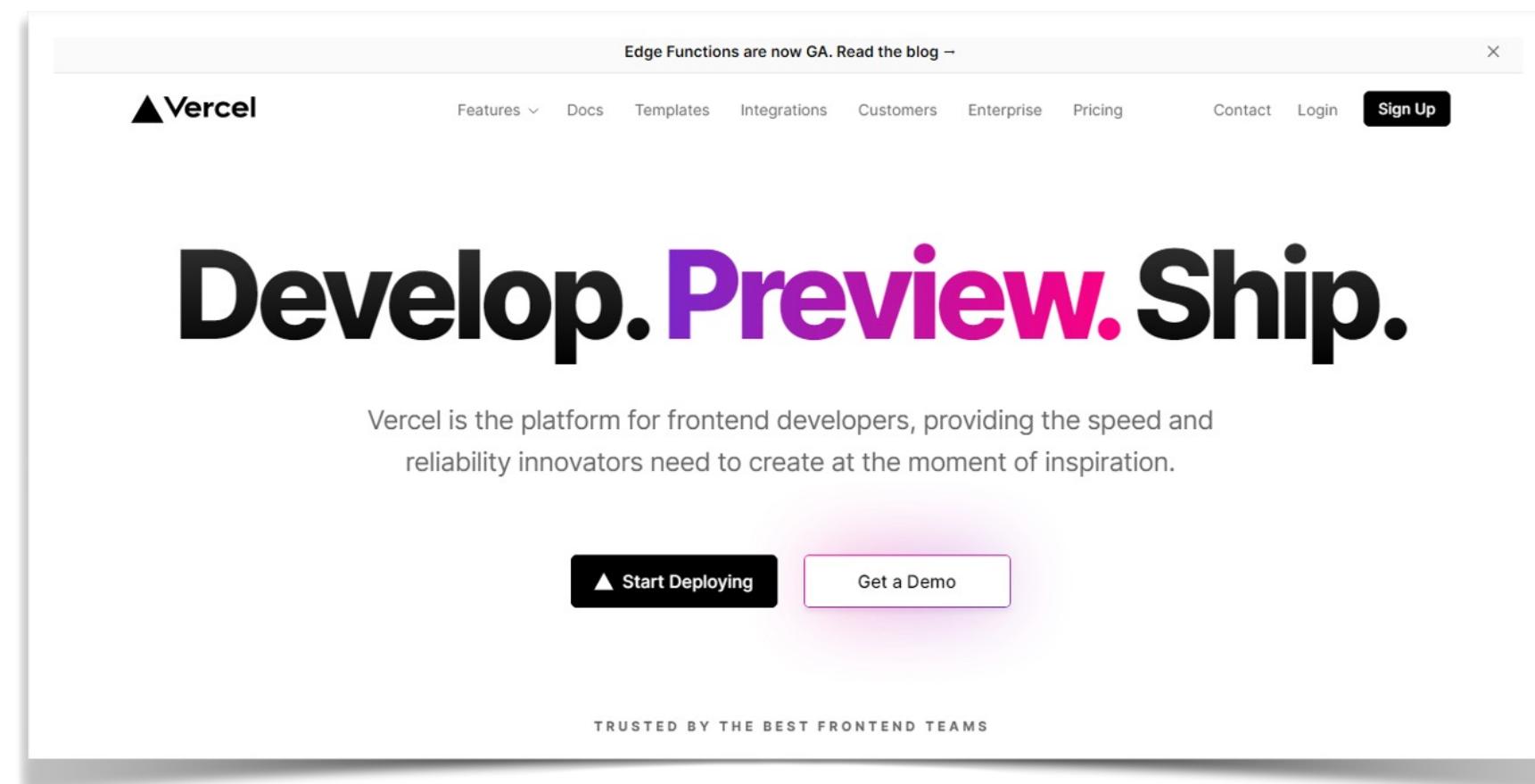
INTRODUCCIÓN

Vercel

Vercel es una plataforma que pone a nuestra disposición alojamiento gratuito para nuestra **API REST** y para nuestro proyecto de Angular con una integración continua con GitHub.

Podéis acceder a Vercel a través de este [enlace](#)

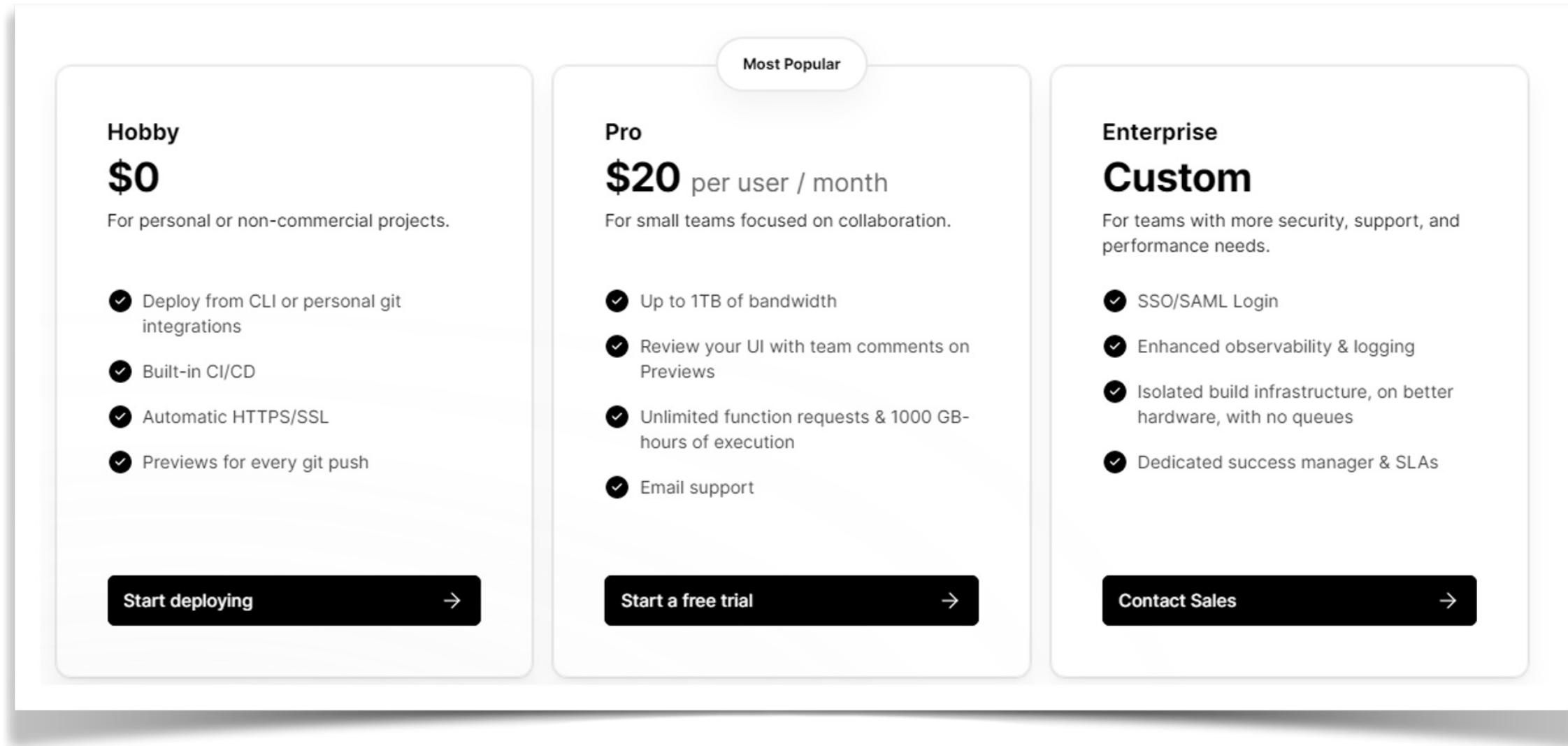
No os olvidéis de iniciar sesión con vuestra cuenta de GitHub ya que vamos a subir los repositorios directamente.



INTRODUCCIÓN

Vercel

Vercel cuenta con varios niveles de cuenta para desplegar proyectos.



Si quieres saber más sobre los límites de las diferentes versiones de Vercel puedes consultar en [este](#) enlace, pero para desplegar nuestra API REST y nuestro proyecto Angular, será suficiente con la versión gratuita.

Fase 2.1 Preparación de la API REST

FASE 2.1

Preparación de la API REST

Antes de nada, debemos realizar unos cambios en nuestra API REST para que podemos utilizarla en el servidor de Vercel correctamente.

Debemos modificar un poco la estructura de las carpetas, ya que Vercel buscará en la carpeta raíz el archivo llamado **index.js** para ejecutarlo, de modo que tendremos que poner en la carpeta raíz nuestro archivo **apiRest.js** y cambiar el nombre a **index.js**.

De modo que tendremos la siguiente estructura:

```
> node_modules  
> src  
  ◆ .gitignore  
  JS index.js  
  {} package-lock.json  
  {} package.json
```

FASE 2.1

Preparación de la API REST

Debemos ir al archivo **package.json** y añadir en “scripts” lo siguiente:

```
"start": "node index.js"
```

De este modo le indicaremos al servidor el comando que necesita para arrancar nuestra API

```
{} package.json > {} dependencies
1  {
2    "name": "back",
3    "version": "1.0.0",
4    "main": "index.js",
5    ▷ Debug
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1",
8      "start": "node index.js"
9    },
10 }
```

FASE 2.1

Preparación de la API REST

Además, debemos crear el archivo **vercel.json**, a la misma altura que el resto de los archivos **json** con la siguiente configuración:

```
{} vercel.json X  
{} vercel.json > ...  
1  {  
2    "version": 2,  
3    "name": "VERCEL-EXPRESS",  
4    "builds": [  
5      { "src": "./index.js", "use": "@vercel/node" }  
6    ],  
7    "routes": [  
8      { "src": "/(.*)", "dest": "/" }  
9    ]  
10 }
```

Nota: El campo **"name"** es opcional, puedes ponerle el nombre que deseas.

FASE 2.1

Preparación de la API REST

Ahora toca configurar las **variables de entorno**.

Las **variables de entorno** son variables que se asignan de forma dinámica a nuestro código. En este caso serán variables como el nombre o la dirección de la base de datos, la cual ahora mismo no podemos saber.

Para prepararlas, debes ir primero a tu archivo **index.js** y cambiar el código para que quede de este modo:

```
1 const app = require("./app");
2
3 app.listen(process.env.PORT || 3000)
```

Así, nuestra API escuchará el puerto que le indique el servidor en el que esté desplegado.

FASE 2.1

Preparación de la API REST

Ahora, debes ir al archivo [database.js](#) de tu API.

Podemos crear variables de entorno a través de `process.env.NOMBRE_DE_LA_VARIABLE`, y debemos asignar una a cada campo relacionado con la base de datos.

Una vez realizados estos cambios, súbelos a [GitHub](#).

```
1 const mysql = require("mysql2")
2
3 const connection = mysql.createConnection({
4     host:      process.env.DB_HOST      || "localhost",
5     user:      process.env.DB_USER      || "root",
6     password: process.env.DB_PASSWORD || "",
7     database: process.env.DB_NAME      || "appbooks",
8     port:      process.env.DB_PORT      || 3306
9 })
```

Fase 2.2 Despliegue en Vercel

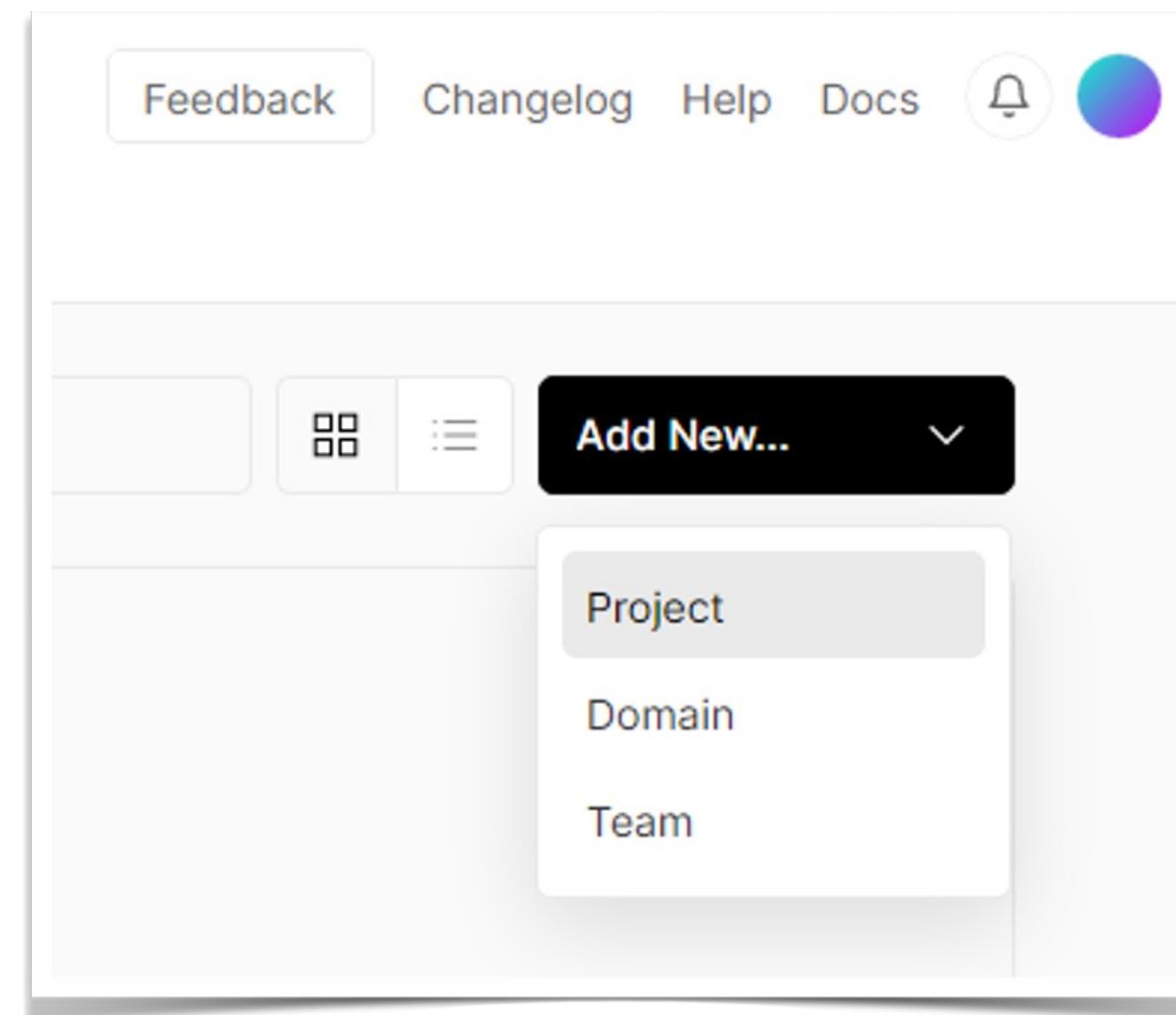
FASE 2.2

Creación del Proyecto en Vercel

Una vez subidos todos los cambios a [GitHub](#), vamos a realizar la subida a [Vercel](#)

Para ello, debes de haber accedido antes con tu cuenta de [GitHub](#).

En la esquina superior derecha, podemos ver el botón de “[Add New...](#)” y elegiremos “[Project](#)”



FASE 2.2

Despliegue

The screenshot shows a user interface for importing GitHub repositories. At the top, there's a dropdown menu set to 'CodenotchSchool' and a search bar with placeholder text 'Search...'. Below this, a list of repositories is displayed:

Repository	Last Updated	Action
Angular-Api	2d ago	Import
Angular	35d ago	Import
Mongo	41d ago	Import
SQL	71d ago	Import
pruebasssss	54d ago	Import

At the bottom of the list, there's a link 'Import Third-Party Git Repository →'.

A continuación, nos pedirá permisos de **Github**.

Nos pedirá elegir los repositorios que queremos

Podemos elegir que reconozca todos, o solo los que le digamos.

Simplemente debemos importar el repositorio en el que esta la API.

FASE 2.2

Configuración

Configure Project

Project Name
api-rest-user

Framework Preset
A Other

Root Directory
. / Edit

> Build and Output Settings

Environment Variables

Name	Value (Will Be Encrypted)
process.env.DB_USER	root

TIP: Paste a .env above to populate the form [Learn more about Environment Variables](#)

Deploy

Ahora, toca configurar los ajustes del proyecto.

En la pestaña de Framework Present elegiremos “other”

En **Build and Output Settings** dejaremos los que vienen por **defecto**

Y en la pestaña de **Enviroment Variables** podremos añadir las **variables de entorno** que necesitamos para conectarnos con la **BBDD**.

No olvidéis que los nombres deben coincidir con los que habéis puesto en la **API REST**.

Por último, clicamos en **Deploy**. Si todo ha ido correctamente ya estaría desplegada tu API

Fase 2.3 Dominio y Prueba

FASE 2.3

Generar dominio

Tras la subida, **Vercel** automáticamente generará un **dominio** para tu **API REST**.

En este caso, hicimos un endpoint "/" que devolvía el mensaje "**"API DESPLEGADA"**".

Production Deployment

The deployment that is available to your visitors.

API DESPLEGADA

DEPLOYMENT
vercel-express-test-oddjxv66x-danielveramontelongo.vercel.app

DOMAINS
vercel-express-test-dusky.vercel.app +2

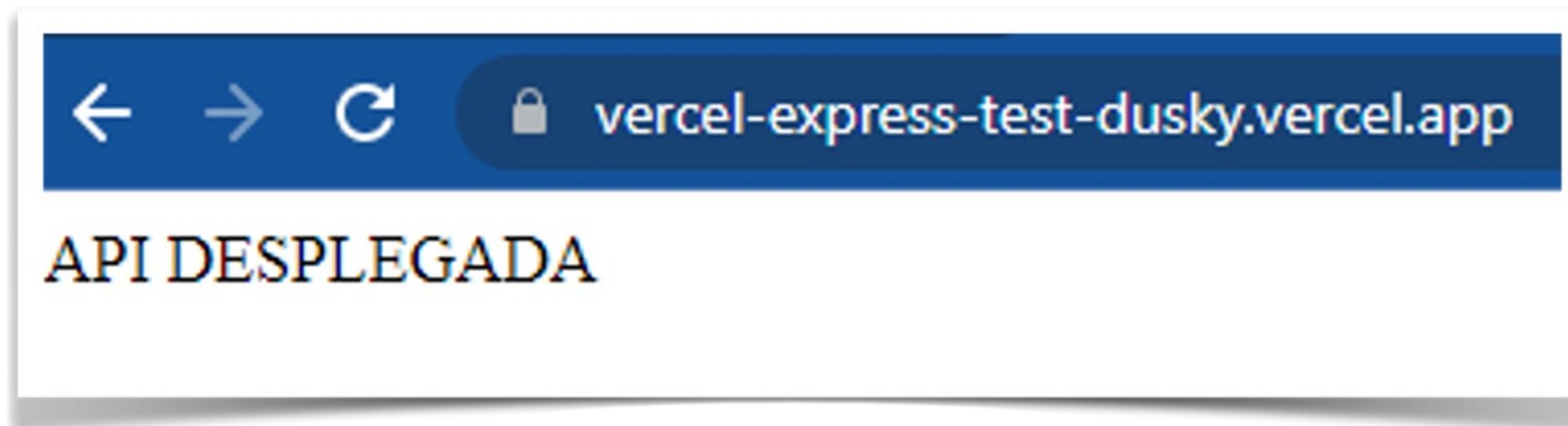
STATUS CREATED
● Ready 2h ago by DanielVeraMontelongo

BRANCH
 main
cambio al index

FASE 2.3

Comprobar funcionamiento

Si clicamos en la URL que nos da [Vercel](#) en la imagen anterior, veremos como se ejecutará el endpoint de la ruta “/”



En este caso la API devuelve el mensaje de API DESPLEGADA cuando se ejecuta el endpoint “/”.

Una vez hecho esto ya podréis probar la [API REST](#) en [Postman](#) sustituyendo “localhost: 3000” por el [endpoint](#) que Vercel os ha generado.

| Fase 3: APP Angular

INTRODUCCIÓN

Depliegue de la APP Angular

En este punto del manual tenemos ya desplegados tanto la bbdd en [MySQL](#) como el API REST para poder acceder a ella.

Además gracias a la conexión de [Vercel](#) con [GitHub](#), el [API REST](#) dispone también de integración continua.

De tal manera que cada cambio que se realice en el API REST y se suba a la rama principal del [Git](#) se desplegará de [manera automática](#).

En esta fase se desplegará una aplicación [Angular](#), se supone que nuestra aplicación Angular ya está creada y conectada con un repositorio de GitHub.

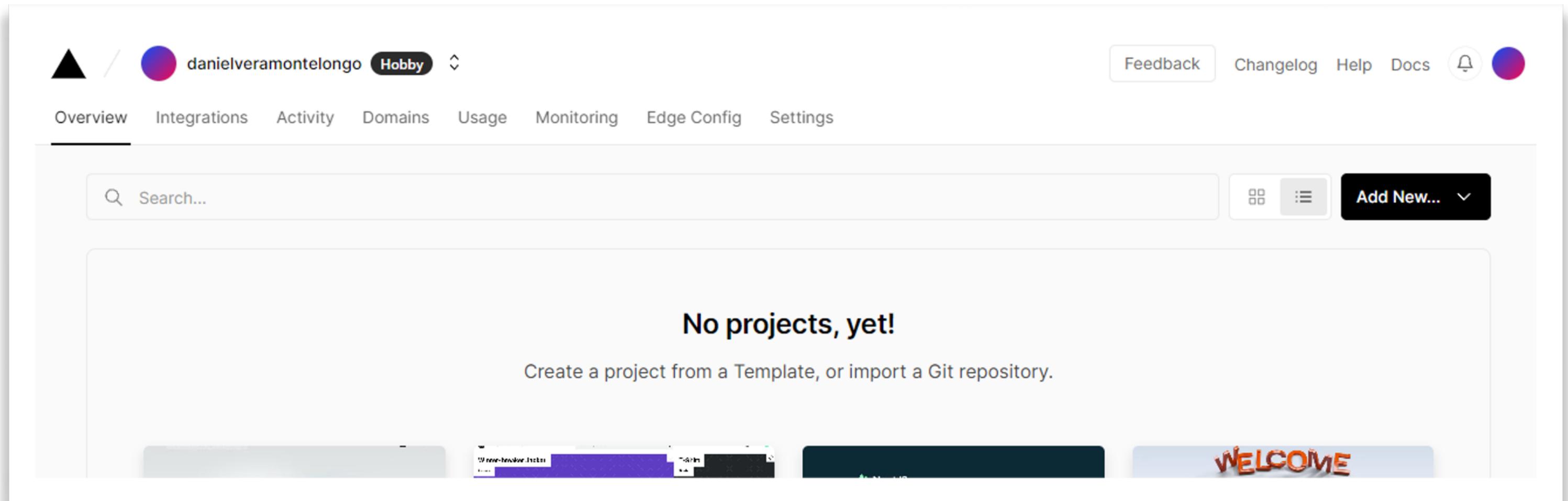
Este manual especificará qué pasos hay que realizar una vez que ya se tenga la aplicación y su conexión con [GitHub](#).

ANGULAR

Despliegue de la APP Angular

Para desplegar en [Vercel](#) una aplicación de Angular, debemos de [iniciar sesión con nuestra cuenta de GitHub en Vercel](#). Nos aparecerán los proyectos que tenemos desplegados en caso de tenerlos, en caso contrario, nos avisará de que no hay proyectos todavía.

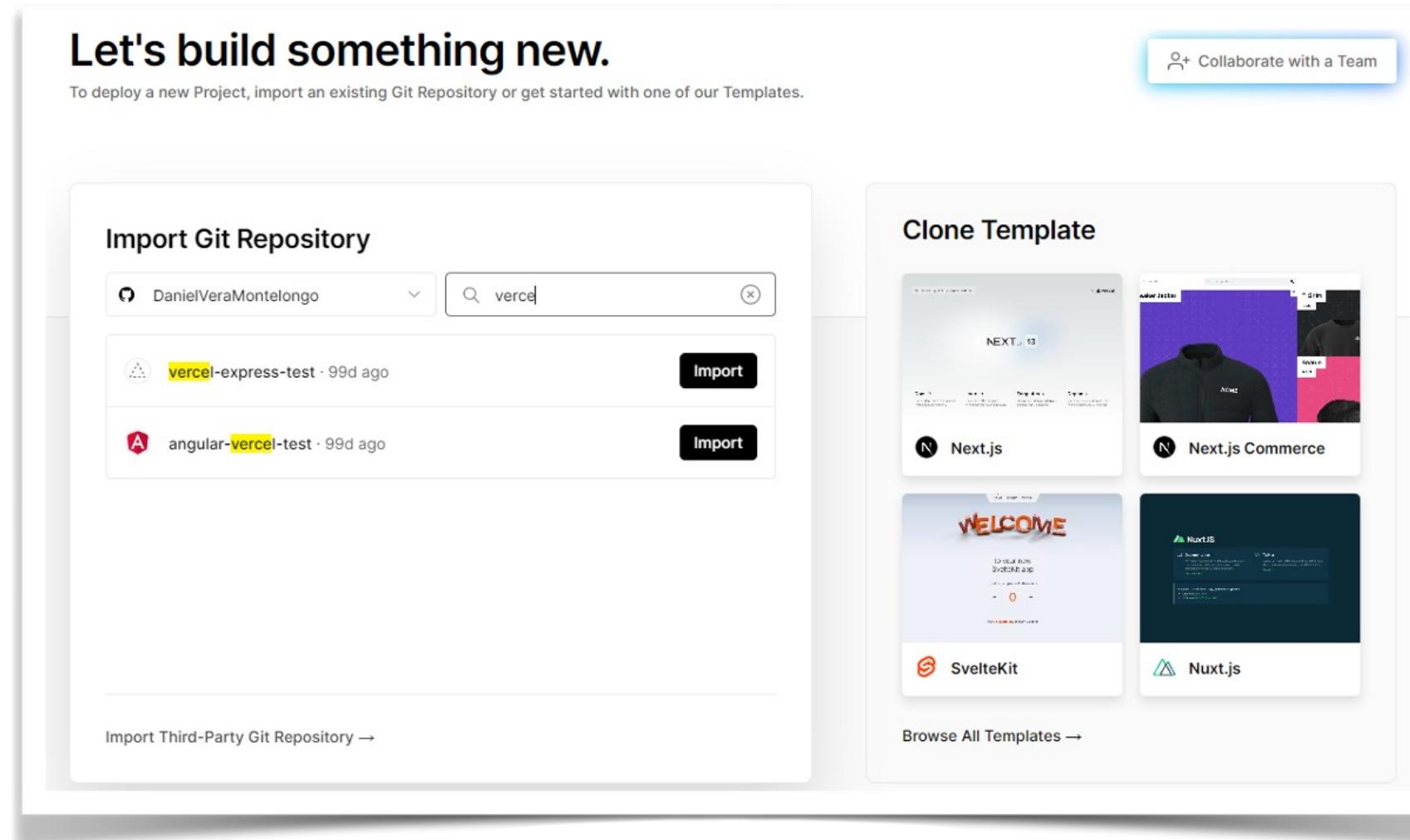
A la derecha veremos un botón “[Add new](#)”. Si le clicamos, nos permitirá desplegar algún proyecto que tenemos en nuestro repositorio.



ANGULAR

Despliegue de la App Angular

Cuando clicamos en “[Add new](#)” nos sacará un desplegable, y clicamos en “[project](#)”. Haciendo esto nos permitirá elegir el proyecto de nuestro GitHub para desplegar.



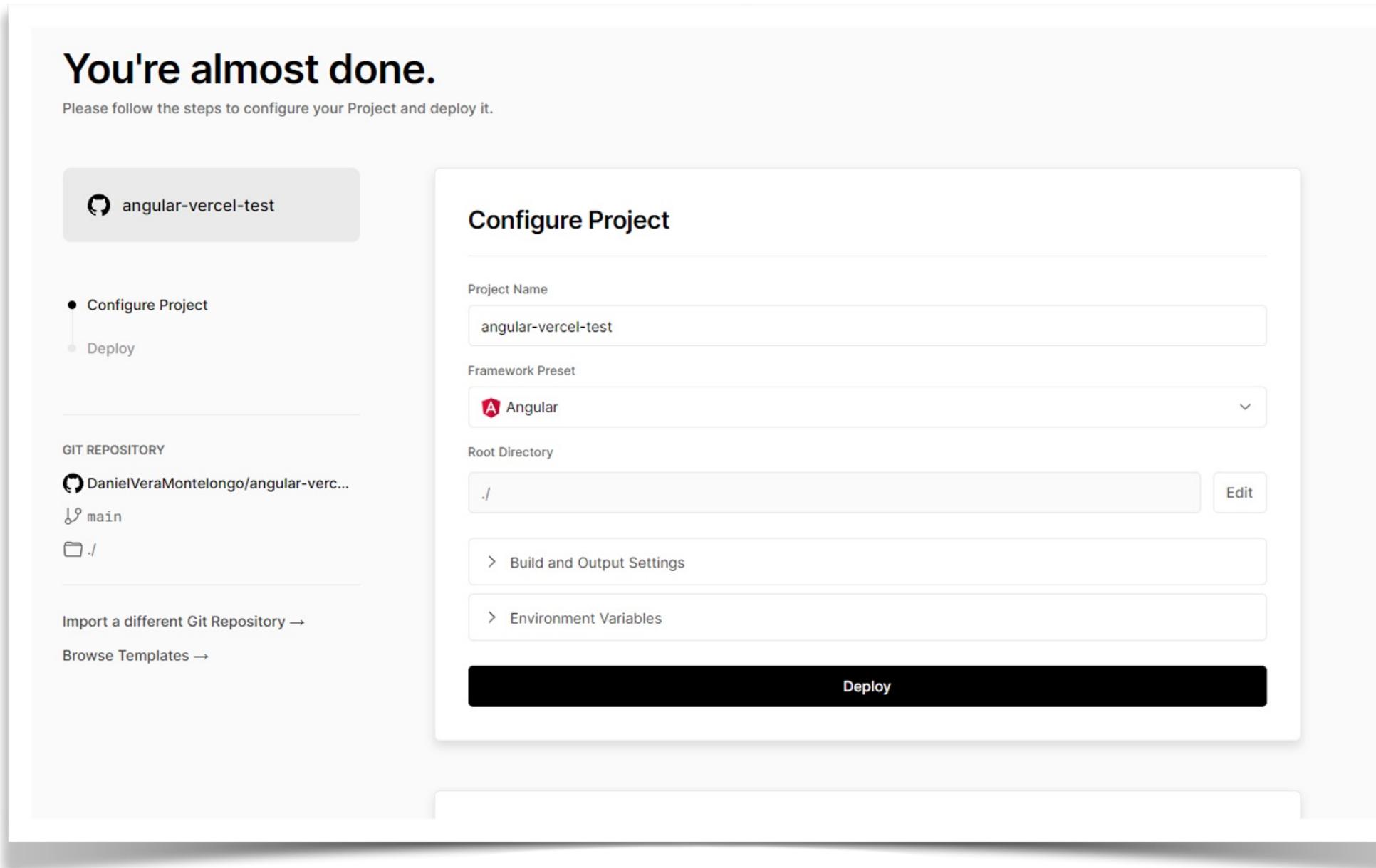
[Vercel](#) cuenta con un buscador para buscar entre los proyecto que tengamos subidos.

Además es capaz de detectar cuál es la tecnología usada en cada proyecto, y en este caso ve como uno de los proyectos es de [Angular](#).

Clicamos en el botón de la derecha “[Import](#)”, para continuar.

ANGULAR

Despliegue de app Angular



En este paso nos pedirá que configuremos el proyecto.

Debemos elegir la **tecnología** con la que está hecho el proyecto, y el directorio raíz.

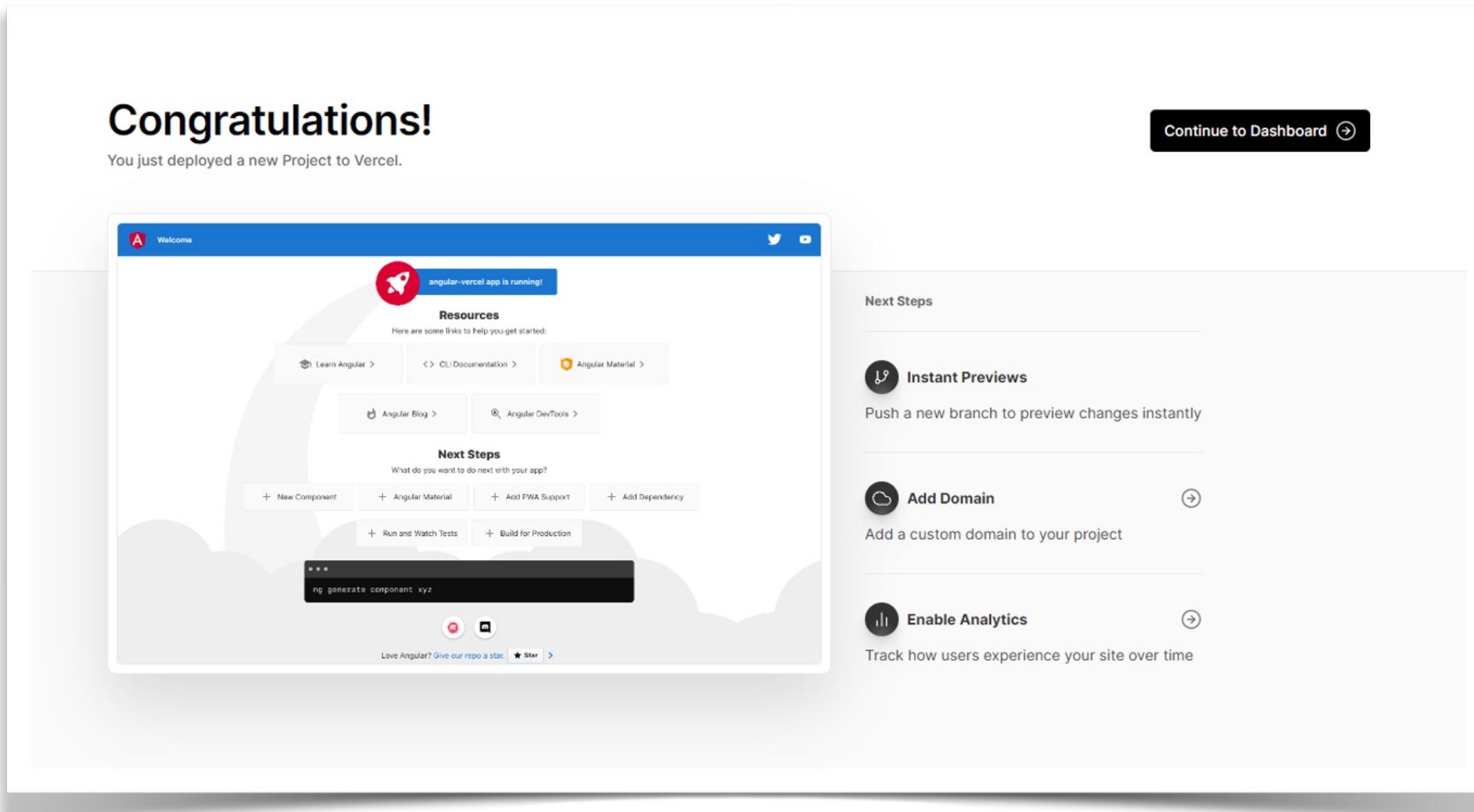
Normalmente no hace falta configurarlo ya que **Vercel** detecta cuál es la tecnología y el directorio.

Hecha la configuración, hacemos clic en "**Deploy**"

ANGULAR

Despliegue de app Angular

EL proceso de despliegue puede **tardar unos minutos**. Si todo ha ido correctamente nos avisará de que el **despliegue ha sido efectivo**. Al hacer clic en la imagen te redirigirá a la url de **Vercel** donde está **desplegado el proyecto**.



codenotch