

# **Documento Despliegue proyecto fondo**

**Fondo-AWS**

Para:  
xxxxxx

Por:  
Javier Orlando Mantilla Portilla  
"jmantillap"

## Tabla de contenido

---

<b>1</b>	<b>ASPECTOS TECNICOS .....</b>	<b>3</b>
1.1	HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO BACKEND .....	3
1.1.1	JDK17 .....	3
1.1.2	Spring Tool Suite.....	3
1.1.3	GIT y AWS .....	3
1.2	ASPECTO TÉCNICO DE DESARROLLO DEL SISTEMA.....	4
1.2.1	Configuración del proyecto .....	4
1.2.2	Estructura del proyecto.....	5
1.2.3	Pruebas unitarias de cobertura.....	5
<b>2</b>	<b>RECURSOS NECESARIOS PARA DESPLIEGUE AWS.....</b>	<b>6</b>
2.1	Archivos .....	6
2.1.1	Plantillas de Cloudformation requeridas para este proyecto: .....	6
2.1.2	Datos para cargar en dynamodb requeridas en las plantillas.....	6
<b>3</b>	<b>EJECUCION DEL PROYECTO Y POSTERIOR DESPLIEGUE AWS .....</b>	<b>7</b>
3.1	Pasos para el correcto funcionamiento.....	7
3.1.1	Configurar credenciales de aws en la CLI .....	7
3.1.2	Creación del bucket s3 para subir los datos que se necesitaran para dynamodb .....	7
3.1.3	Copiar los archivos csv que se van a importar a dynamodb .....	8
3.1.4	Desplegar las tablas de dynamodb necesarias para el proyecto. ....	8
3.1.5	Configuración del SES para el envío de correo. ....	9
3.1.6	Creación de la imagen del contenedor que se subirá a AWS ECR.....	11
3.1.7	Crear el repositorio privado en AWS ECR. ....	12
3.1.8	Subir la imagen al repositorio AWS ECR.....	13
3.1.8.1	Paso 1: Autenticarse. ....	13
3.1.8.2	Paso 2: Crear el tag. ....	13
3.1.8.3	Paso 3: Subir la imagen. ....	13
3.1.9	Desplegar ECS Fargate. ....	13
3.1.10	Prueba de funcionamiento de la infraestructura.....	14
3.1.11	Desplegar bucket para web site (En progreso).....	14

# 1 ASPECTOS TECNICOS

---

## 1.1 HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO BACKEND

### 1.1.1 JDK17

Java™ Development Kit (JDK) es un software para los desarrolladores de Java. Incluye el intérprete Java, clases Java y herramientas de desarrollo Java (JDT): compilador, depurador, desensamblador, visor de applets, generador de archivos de apéndice y generador de documentación.

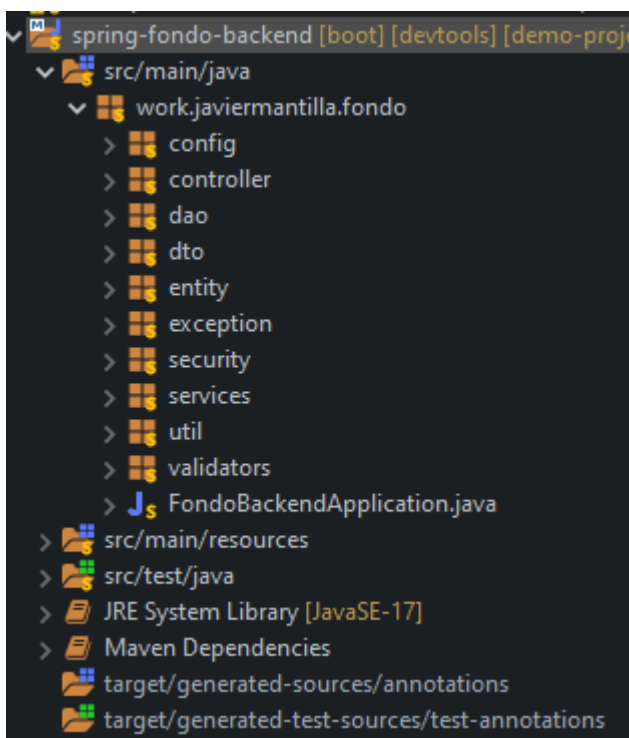
<https://jdk.java.net/java-se-ri/17>

### 1.1.2 Spring Tool Suite

Spring Tool Suite es un IDE para desarrollar aplicaciones Spring. Es un entorno de desarrollo basado en Eclipse. Proporciona un entorno listo para usar para implementar, ejecutar, implementar y depurar la aplicación. Valida nuestra aplicación y proporciona soluciones rápidas para las aplicaciones.

<https://www.javatpoint.com/spring-boot-download-and-install-sts-ide>

<https://www.springla.io/spring/spring-tool-suite>

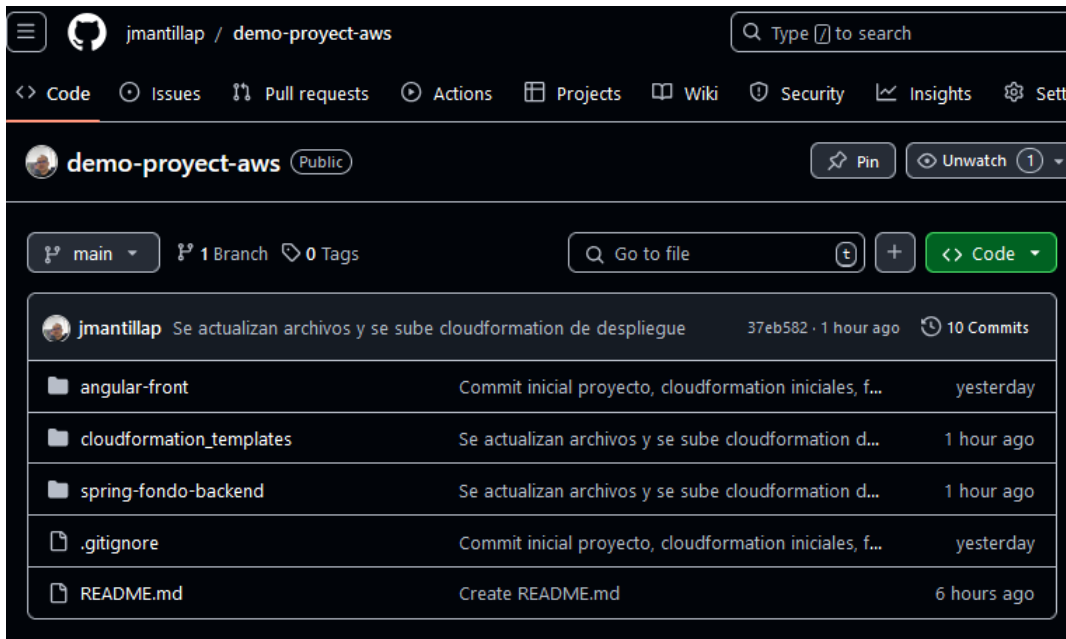


### 1.1.3 GIT y AWS

Git es un sistema de control de versiones distribuido: seguimiento de cambios en cualquier conjunto de archivos, generalmente utilizado para coordinar el trabajo entre programadores que desarrollan en colaboración el código fuente durante el desarrollo del software

La ubicación del proyecto está en la ruta.

<https://github.com/jmantillap/demo-proyect-aws>



Amazon Web Services, Inc. es una subsidiaria de Amazon que proporciona API y plataformas de computación en la nube bajo demanda a personas, empresas y gobiernos, en forma de pago por uso medido. Muchas veces, los clientes usarán esto en combinación con el escalado automático.

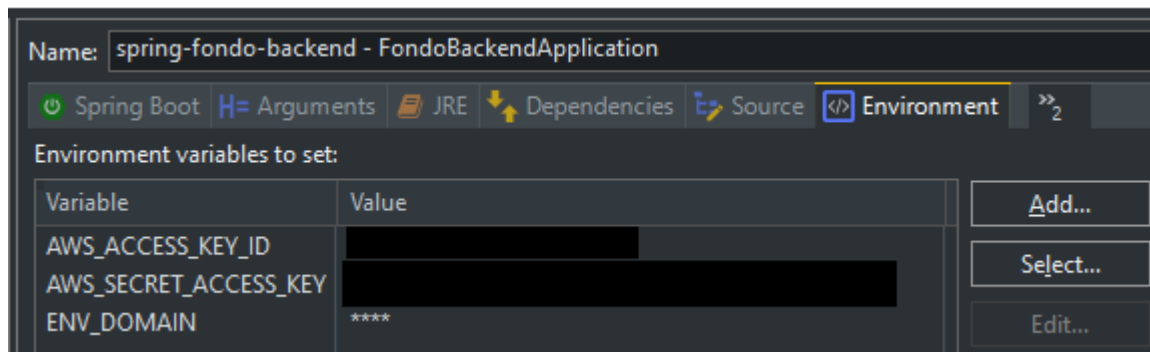


## 1.2 ASPECTO TÉCNICO DE DESARROLLO DEL SISTEMA

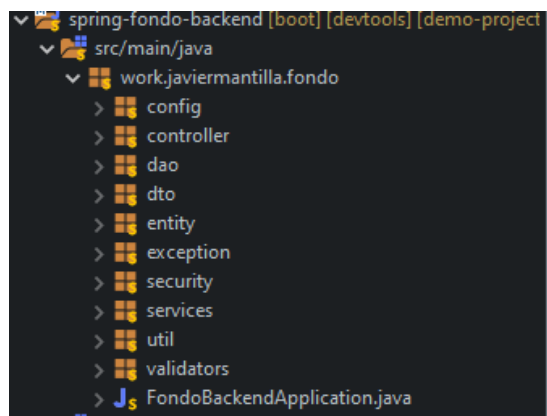
### 1.2.1 Configuración del proyecto

Para que funcione el proyecto localmente y se pueda conectar a aws para dynamodb y ses se deben colocar las variables de entorno:

- Valores:
  - AWS\_ACCESS\_KEY\_ID=<\*\*\*\*>
  - AWS\_SECRET\_ACCESS\_KEY=<\*\*\*>
  - ENV\_DOMAIN=\*\*\*\*



### 1.2.2 Estructura del proyecto



### 1.2.3 Pruebas unitarias de cobertura

\$ mvn clean test

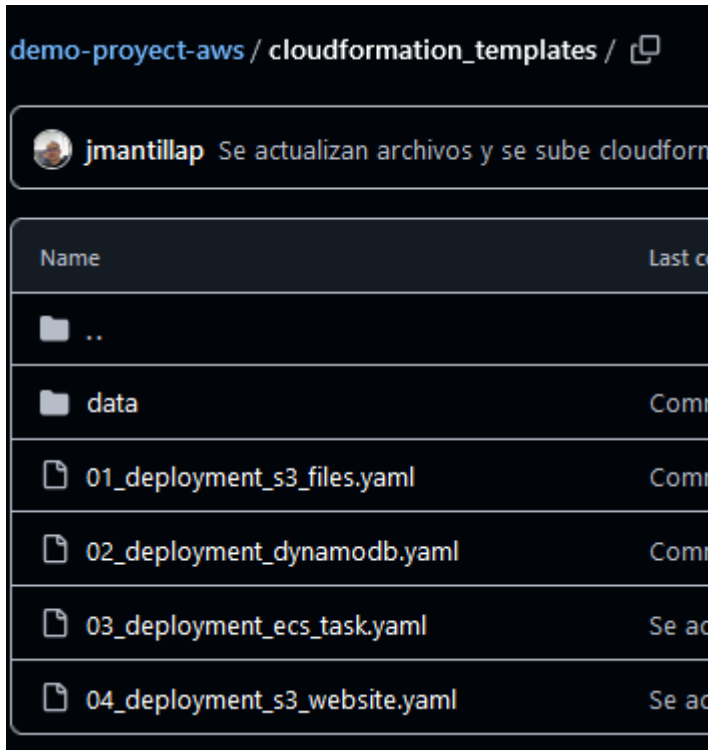
fondo-bakend					
fondo-bakend					
Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missi
work.javiermantilla.fondo.services.impl	<div><div></div></div>	81 %	<div><div></div></div>	65 %	
work.javiermantilla.fondo.dao.impl	<div><div></div></div>	100 %	<div><div></div></div>	100 %	
work.javiermantilla.fondo.controller	<div><div></div></div>	100 %		n/a	
Total	160 of 1.381	88 %	14 of 50	72 %	

## 2 RECURSOS NECESARIOS PARA DESPLIEGUE AWS

---

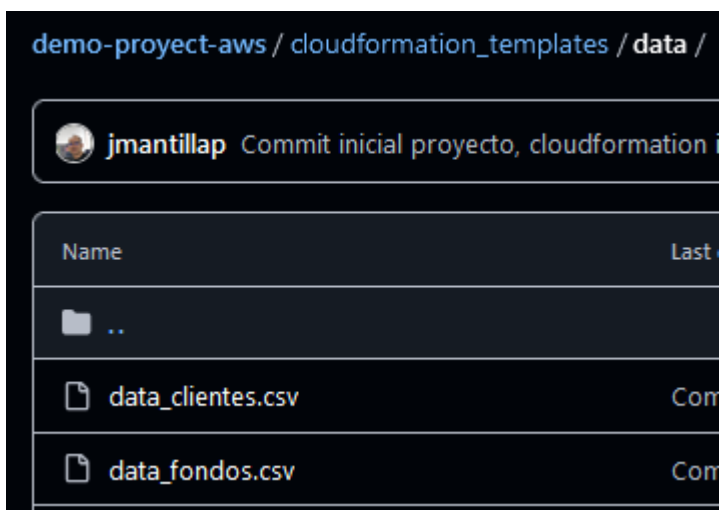
### 2.1 Archivos

#### 2.1.1 Plantillas de Cloudformation requeridas para este proyecto:



demo-proyect-aws / cloudformation_templates /	
jmantillap Se actualizan archivos y se sube cloudform	
Name	Last c
..	
data	Comi
01_deployment_s3_files.yaml	Comi
02_deployment_dynamodb.yaml	Comi
03_deployment_ecs_task.yaml	Se ac
04_deployment_s3_website.yaml	Se ac

#### 2.1.2 Datos para cargar en dynamodb requeridas en las plantillas.



demo-proyect-aws / cloudformation_templates / data /	
jmantillap Commit inicial proyecto, cloudformation i	
Name	Last
..	
data_clientes.csv	Com
data_fondos.csv	Com

## 3 EJECUCION DEL PROYECTO Y POSTERIOR DESPLIEGUE AWS

### 3.1 Pasos para el correcto funcionamiento

#### 3.1.1 Configurar credenciales de aws en la CLI

```
C:\Users\JAVIER>aws configure
AWS Access Key ID [None]: 
AWS Secret Access Key [None]: 
Default region name [None]: us-east-1
Default output format [None]:
```

#### 3.1.2 Creación del bucket s3 para subir los datos que se necesitaran para dynamodb

\$ aws cloudformation validate-template --template-body [file:///01\\_deployment\\_s3\\_files.yaml](#)

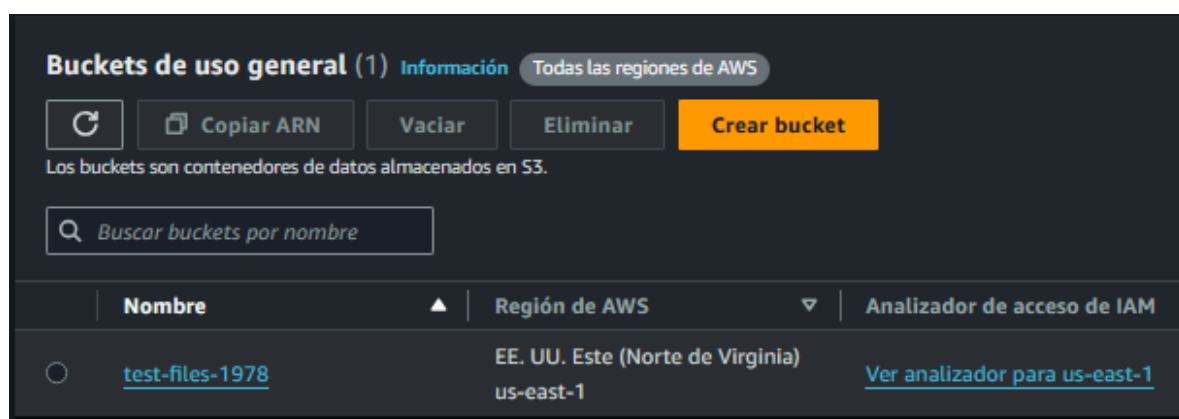
```
JAVIER@DEV MINGW64 /d/demo-project-pactual/cloudformation_templates
$ aws cloudformation validate-template --template-body file:///deployment_s3_file
s.yaml
{
  "Parameters": [
    {
      "ParameterKey": "BucketName",
      "DefaultValue": "test-files-1978",
      "NoEcho": false
    }
  ],
  "Description": "CloudFormation template for s3 bucket"
}
```

\$ aws cloudformation deploy --stack-name stack-test-files-1978 --template-file  
01\_deployment\_s3\_files.yaml --capabilities CAPABILITY\_IAM CAPABILITY\_NAMED\_IAM

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/cloudformation_templates
$ aws cloudformation deploy --stack-name stack-test-files-1978 --template-file deployment_s3_files.yaml --capabilities CAPABI
TY_IAM CAPABILITY_NAMED_IAM

Waiting for changeset to be created..
Waiting for stack create/update to complete
Successfully created/updated stack - stack-test-files-1978
```

Imagen del bucket creado.



### 3.1.3 Copiar los archivos csv que se van a importar a dynamodb

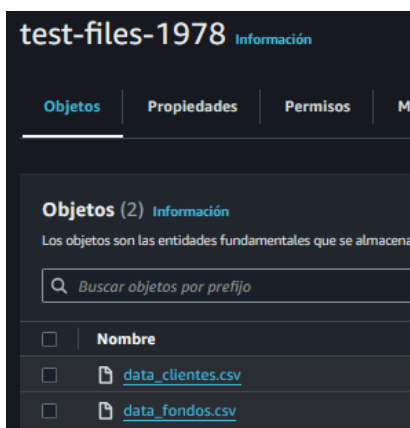


```
$aws s3 cp data_clientes.csv s3://test-files-1978/data_clientes.csv
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/cloudformation_templates
$ aws s3 cp data_clientes.csv s3://test-files-1978/data_clientes.csv
upload: .\data_clientes.csv to s3://test-files-1978/data_clientes.csv
```

```
$ aws s3 cp data_fondos.csv s3://test-files-1978/data_fondos.csv
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/cloudformation_templates
$ aws s3 cp data_fondos.csv s3://test-files-1978/data_fondos.csv
upload: .\data_fondos.csv to s3://test-files-1978/data_fondos.csv
```



### 3.1.4 Desplegar las tablas de dynamodb necesarias para el proyecto.

```
aws cloudformation validate-template --template-body
file://02_deployment_dynamodb.yaml
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/cloudformation_templates
$ aws cloudformation validate-template --template-body file://02_deployment_dynamodb.yaml
{
  "Parameters": [
    {
      "ParameterKey": "BucketName",
      "DefaultValue": "test-files-1978",
      "NoEcho": false
    }
  ],
  "Description": "CloudFormation template para creacion de tablas y importacion de datos"
}
```

```
$ aws cloudformation deploy --stack-name stack-dynamo-tables-test --template-file
02_deployment_dynamodb.yaml --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/cloudformation_templates
$ aws cloudformation deploy --stack-name stack-dynamo-tables-test --template-file 02_deployment_dynamodb.yaml --capabilities CAP
ABILITY_IAM CAPABILITY_NAMED_IAM
waiting for changeset to be created..
waiting for stack create/update to complete
Successfully created/updated stack - stack-dynamo-tables-test
```



<input type="checkbox"/>	Nombre ▲	Estado ▼	Clave de partición ▼	Clave de ordenación
<input type="checkbox"/>	<a href="#">CLIENTES</a>	✓ Activo	id (S)	email (S)
<input type="checkbox"/>	<a href="#">FONDOS</a>	✓ Activo	id (S)	codigo (S)
<input type="checkbox"/>	<a href="#">TRANSACCIONES</a>	✓ Activo	id (S)	cliente (S)

Datos de la importacion en la tabla de clientes

Seleccione una tabla o un índice

Tabla - CLIENTES ▼

Seleccione la proyección

Todos los atributos

► Filtros

Ejecutar

Restablecer

✓ Completado. Unidades de capacidad de lectura consumidas: 3.5

Elementos devueltos (2)

<input type="checkbox"/>	id (Codena) ▼	email (Codena) ▼	nombre ▼	saldo
<input type="checkbox"/>	<a href="#">37896334</a>	eliana.bravovesga@g...	Eliana Bravo	500000
<input type="checkbox"/>	<a href="#">13544171</a>	jmantillap@gmail.com	Javier Mant...	500000

```
data > data_clientes.csv
1 id,email,nombre,saldo
2 13544171,jmantillap@gmail.com,Javier Mantilla,500000
3 37896334,eliana.bravovesga@gmail.com,Eliana Bravo,500000
```

### 3.1.5 Configuración del SES para el envío de correo.

Necesario para cuando un cliente se suscribe a un fondo. Se Necesita crear la identidad en aws:

Amazon SES > Configuración: identidades

Identidades

En el panel **Identidades**, se muestran las identidades de los dominios, subdominios y direcciones de correo electrónico. Todas las identidades deben verificarse antes de usarse para enviar correos electrónicos en Amazon SES. [Más información](#). En el panel **Recomendaciones**, se muestran los problemas de autenticación de correo electrónico de gran impacto encontrados en las identidades que selecciona y comprueba si hay recomendaciones. [Más información](#)

Identidades (2) Información

Comprobar si hay recomendaciones

Enviar correo electrónico de prueba

Eliminar

Crear identidad

Buscar todas las identidades

Identidad	Tipo de identidad	Estado de identidad
google.com	Dominio	Verificación pendiente
jmantillap@gmail.com	Dirección de correo electrónico	Verificado

Recomendaciones (0) Información

Para ver recomendaciones de gran impacto, seleccione hasta 10 identidades de la lista anterior y elija **Comprobar si hay recomendaciones**. Si implementa una solución proporcionada, seleccione **Comprobar si hay recomendaciones** para actualizar la tabla y asegurarse de que ya no aparezca en la lista. Si habilita el **Administrador virtual de entrega**, esta tabla se actualizará automáticamente para todas las identidades.

Buscar recomendaciones

Impacto	Nombre de identidad	Antigüedad	Recomendación o descripción	Última comprobación	Resolver el problema
No se encontraron recomendaciones					
Seleccione hasta 10 identidades de dominio verificadas y haga clic en <b>Comprobar si hay recomendaciones</b> en la tabla Identidades anterior.					
Permitir que el Administrador virtual de entrega genere recomendaciones automáticamente					

Posteriormente llega al correo la confirmación desde el correo que se va a realizar el envío.

Amazon Web Services – Email Address Verification Request in region US East (N. Virginia)

(Amazon Web Services: solicitud de verificación de dirección de correo electrónico en la región Este de EE. UU. (Norte de Virginia))

Recibidos

Amazon Web Services

<no-reply-aws@amazon.com>

17:25 (hace 4 horas)

☆

😊

↩

⋮

para mí

inglés → español

Mostrar original

⚙

Estimado cliente de Amazon Web Services:

Hemos recibido una solicitud para autorizar el uso de esta dirección de correo electrónico con Amazon SES y Amazon Pinpoint en la región este de EE. UU. (Norte de Virginia). Si solicitó esta verificación, acceda a la siguiente URL para confirmar que está autorizado a usar esta dirección de correo electrónico:

[https://email-verification-us-east-1.amazonaws.com/?Context=590184132147&X-Amz-Date=20240913T222511Z&Identity.IdentityName=jmantillap%40gmail.com&X-Amz-Algorithm=AWS4-HMAC-SHA256&Identity.IdentityType=EmailAddress&X-Amz-SignedHeaders=host&X-Amz-Credential=AKIAVM67ZIEFRDECB3HF%2F20240913%2Fus-east-1%2Fses%2Faws4\\_request&Operation=ConfirmVerification&Namespace=Bacon&X-Amz-Signature=bb63862c45bd6b9b17d68eac4ce9d49eb267eedbd821fa55e8e73982ce13d0ee](https://email-verification-us-east-1.amazonaws.com/?Context=590184132147&X-Amz-Date=20240913T222511Z&Identity.IdentityName=jmantillap%40gmail.com&X-Amz-Algorithm=AWS4-HMAC-SHA256&Identity.IdentityType=EmailAddress&X-Amz-SignedHeaders=host&X-Amz-Credential=AKIAVM67ZIEFRDECB3HF%2F20240913%2Fus-east-1%2Fses%2Faws4_request&Operation=ConfirmVerification&Namespace=Bacon&X-Amz-Signature=bb63862c45bd6b9b17d68eac4ce9d49eb267eedbd821fa55e8e73982ce13d0ee)

Amazon SES > Configuración: identidades

Identidades

En el panel **Identidades**, se muestran las identidades de los dominios, subdominios y direcciones de correo electrónico. Todas las identidades deben verificarse antes de usarse para enviar correos electrónicos en Amazon SES. [Más información](#). En el panel **Recomendaciones**, se muestran los problemas de autenticación de correo electrónico de gran impacto encontrados en las identidades que selecciona y comprueba si hay recomendaciones. [Más información](#)

Identidades (2) Información

Comprobar si hay recomendaciones

Enviar correo electrónico de prueba

Buscar todas las identidades

Identidad	Tipo de identidad	Estado de identidad
google.com	Dominio	Verificación pendiente
jmantillap@gmail.com	Dirección de correo electrónico	Verificado

Prueba desde el aplicativo cuando se realiza una apertura



### 3.1.6 Creación de la imagen del contenedor que se subirá a AWS ECR.

Se debe tener el plugin en el pom.xml de dockerfile-maven-plugin

```
<plugin>
<groupId>com.spotify</groupId>
<artifactId>dockerfile-maven-plugin</artifactId>
<version>1.4.13</version>
<executions>
<execution>
<id>default</id>
<goals>
<goal>build</goal>
<goal>push</goal>
</goals>
</execution>
</executions>
<configuration>
<repository>jmantillap/fondo</repository>
<tag>${project.version}</tag>
<buildArgs>
<JAR_FILE>${project.build.finalName}.jar</JAR_FILE>
</buildArgs>
</configuration>
</plugin>
```

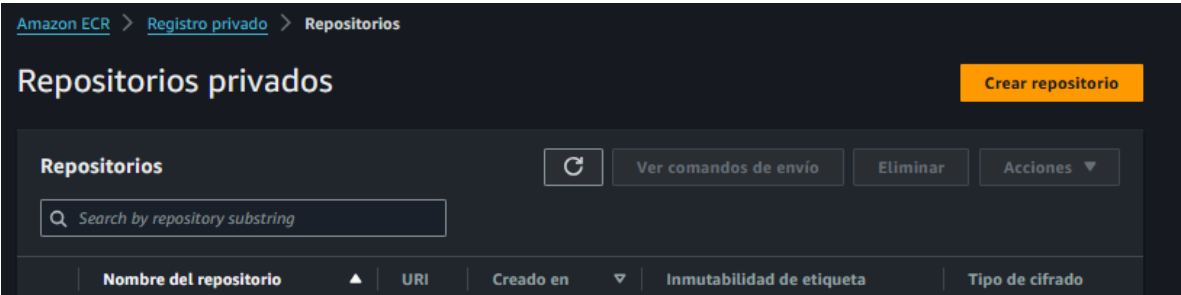
Ejecutar el comando maven que genera el jar y crea la imagen con base en el archivo Dockerfile que tiene el proyecto.

\$ mvn clean verify

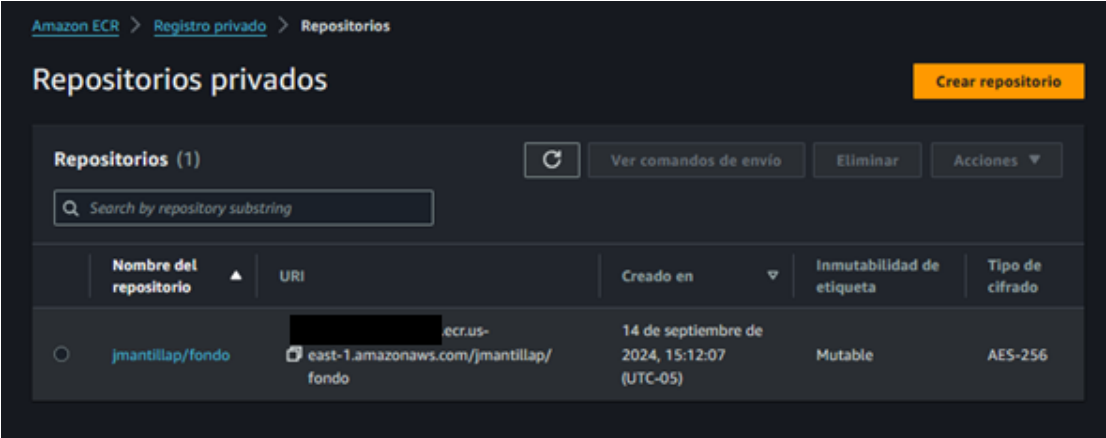
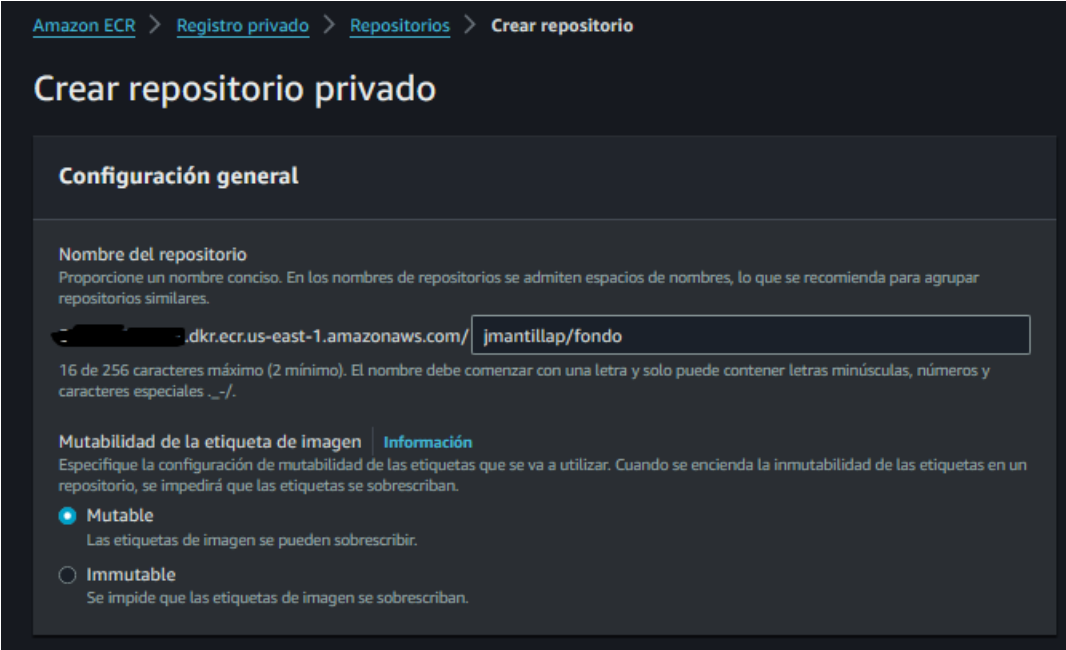
```
INFO] [Warning] One or more build-args [JAR_FILE] were not consumed
INFO] Successfully built 1f8032893556
INFO] Successfully tagged jmantillap/fondo:1.0
INFO]
INFO] Detected build of image with id 1f8032893556
INFO] Building jar: D:\demo-project-pactual\spring-fondo-backend\target\fondo-backend-1.0-docker-info.jar
INFO] Successfully built jmantillap/fondo:1.0
INFO] -----
INFO] BUILD SUCCESS
INFO] -----
INFO] Total time: 03:53 min
INFO] Finished at: 2024-09-14T15:05:51-05:00
INFO] -----
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/spring-fondo-backend (main)
$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
jmantillap/fondo    1.0         1f8032893556     About a minute ago 463MB
openjdk             17-jdk-slim 37cb44321d04     2 years ago       408MB
```

3.1.7 Crear el repositorio privado en AWS ECR.



Se debe crear el repositorio con el mismo nombre de la imagen que se creó previamente en el punto 3.1.6



### 3.1.8 Subir la imagen al repositorio AWS ECR.

CLI de AWS necesaria para montar la infraestructura del fargate.

#### 3.1.8.1 Paso 1: Autenticarse.

```
$ aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin <idCuenta>.dkr.ecr.us-east-1.amazonaws.com
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/spring-fondo-backend (main)
$ aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin .dkr.ecr.us-east-1.amazonaws.com
Login Succeeded
```

#### 3.1.8.2 Paso 2: Crear el tag.

```
$ docker tag jmantillap/fondo:1.0 <idCuenta>.dkr.ecr.us-east-1.amazonaws.com/jmantillap/fondo:1.0
```

#### 3.1.8.3 Paso 3: Subir la imagen.

```
$ docker push <idCuenta>.dkr.ecr.us-east-1.amazonaws.com/jmantillap/fondo:1.0
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/spring-fondo-backend (main)
$ docker push dkr.ecr.us-east-1.amazonaws.com/jmantillap/fondo:1.0
The push refers to repository [dkr.ecr.us-east-1.amazonaws.com/jmantillap/fondo]
e57bd226f0cb: Preparing
6be690267e47: Preparing
13a34b6fff78: Preparing
9c1b6dd6c1e6: Preparing
13a34b6fff78: Pushed
e57bd226f0cb: Pushed
9c1b6dd6c1e6: Pushed
6be690267e47: Pushed
1.0: digest: sha256:8d04d822c79ff79d3a2e81b42902e78837a8cb85e868eaf46b4f0e3acd9c4490 size: 1165
```



Amazon ECR > Registro privado > Repositorios > jmantillap/fondo

## jmantillap/fondo

Ver comandos de envío

Imágenes (1)

Buscar artefactos

	Etiqueta de imagen	Tipo de artefacto	Enviado a	Tamaño (MB)	URI de imagen	Resumir
<input type="checkbox"/>	1.0	Image	14 de septiembre de 2024, 15:23:25 (UTC-05)	270.32	 Copiar URI	 sha256:8d04d822c79ff79...

### 3.1.9 Desplegar ECS Fargate.

Ejecutar el template → 03\_deployment\_ecs\_task.yaml

```
$ aws cloudformation deploy --stack-name stack-ecs-test --template-file 03_deployment_ecs_task.yaml --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM
```

```
JAVIER@DEV MINGW64 /d/demo-project-pactual/cloudformation_templates (main)
$ aws cloudformation deploy --stack-name stack-ecs-test --template-file 03_deployment_ecs_task.yaml --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM
Waiting for changeset to be created..
Waiting for stack create/update to complete
Successfully created/updated stack - stack-ecs-test
```

### 3.1.10 Prueba de funcionamiento de la infraestructura.

### 3.1.11 Desplegar bucket para web site (En progreso).

Ejecutar el template → 04\_deployment\_s3\_website.yaml

```
$ aws cloudformation deploy --stack-name stack-s3-web-site-test --template-file  
04_deployment_s3_website.yaml --capabilities CAPABILITY_IAM CAPABILITY_NAMED_IAM
```

---

# Gracias

**Contacto developer:**

[jmantillap@gmail.com](mailto:jmantillap@gmail.com)

Floridablanca, Colombia

---

**software**one