

a MPb7f e

O z x z v z M©z MUx OÙv y z 92  
 Ml g Nzy xò2M v MdUS v z v z  
 Ml g Yv wyv z è©ùè v z ùx z  
 z z èz xz

77gz z wz 868: 3A è 4 zvy

Pb	P fl Q	P	W fl Pb	m
----	--------	---	---------	---

## Introduzione

bzù v v v zx ù©è yè 2è x èùx z z z èz xz èz z è v y è 2  
 ùU zùèz v M èèèùz z z è v© è v4

h v ùz z è èèz v ùèv è èùx zv è z yèM©z "è z ùèz è2xv vxèyèè z v©èz x  
 è z èMdUw zy yèz v z zx ©è z è xèv©v yz è z z z èv zùN8N2xàz zù  
 N8O4

U z v è ù3 èùz yèv x z èù v z M©z MUx OÙv yz 9 èè v y  
 Ml g Nzy xò4Y z z v èz M v MdUSv z v z Ml g Yv wyv z ù  
 è z©z z x x vò ùzy©z ww z dèzx z z v z ù è è v è z  
 yzùv v v z fz èvùM © z zy Sz z v è .f MS/4

U ù z2 zy z x z èù è z v èè v z Ml g Yv wyv z ©èv©z èM v Nzy xò  
 z zx z v z è v è èyz©è z èyèz v z z yv yv vw z zù è vùz z  
 è vùè v ùz è v xàv 4

dz èù èùxè z èz yè z z è v è E M©z yèv è z v xùz èè èè 2  
 xv vxz yè©z èz èàèz z z èèzùz èèèv f v4Mù z è z yzùv è ù 2v z  
 z M©z MU è v z2è ©vy yèvè v z èxùz èx è z èèzx z v z  
 è v è è ùz v z z z è è èv z z vèx vùz©v è4

hè z è ù v x xz v yèw z yzè z è èM g z -yèz z è v yè  
© v v è z4

O è xà !

d è v yèz v z zù è yzù è ù z z v è z2yèx è w z z z z ù z x ù © è xà è z  
xàz è z z E

74 **Amazon Bedrock e Claude 3** EM v Nzy xò z è è x ù v z z © z è  
xàz z vxxz z ù z vèR yv è a yzù .Ra / à v è yvù z è xè vù è  
v è yz yèU zù © z v M è xà z4

Où yz 92 à v yv M à è2 yzè yzù è yè è © v © è v v v è yè è v è è  
v è z M v Nzy xò 4 Qxxz ù z z v x z è z z © z z v è z yzù è © v © è  
v vù z z yz y ù è yz vù z ù x z v è z yè M © z M Ux z v è v è z4

Yz W ù y © z Nv z yè M v Nzy xò è z z yèx ù z © v z © è M © z M Uv z è  
yè yv è z è z è x z z y ù yè vxxzyz z z è v z è v è è è z v è z  
è è v z z è z4

Qxx vù è è xà è z ù z W ù y © z Nv z z © è M © z è M v Nzy xò E

- Yz W ù y © z Nv z è v v z x è v xà è v v **Retrieval-Augmented Generation**  
. f MS/ z è x v z è v è è yvù z è yè yv è yz è x ù z è zù z è z © z z v z  
yv è yv è vù yzù xà z v è z v © è M © z 4
- dz x ù z © v z v W ù y © z Nv z v è è yv è yz è z x è v z w xò z M v g9  
x z z z è y x z è è è z yè yv è è z v è x z z4
- Yz W ù y © z Nv z è x è v è è x z v è v è xà è z v © è M © z 2  
z è è v y ù è à © v è z v v ù è z f MS x ù v z z © z è v 4
- e z v v v è z vxxz ù z v è è z3 3 v ò z è y xz y ù z xz v è z è z © v z  
è è yv è zù v è v è yzù v © z z è è v è x è z z è v y ù z xz è yè  
x è vyyz v z yzù yzù z ù è è yè yv è è v è 4
- Sù è v © z è yè M v Nzy xò è z z v è z v è è x z ù xà z v è z yè  
x è è v v x ù v è z v è v yè yv è z z z x è z yè v è z z è z © v è z yè è z è z  
x ù v è z yè yv è z v è è z yzù z è v y z W ù y © z Nv z x ù z © v z4
- Sù è v © z zù w v z è è è © v © è v vù z yv © è z è z z zù w v è z  
è z v © è z x z W ù y © z Nv z2 xà z v z v è è z è z è z z zù w v z4

84 **Amazon API Gateway e AWS Lambda** EM v MdUS v z v z è è  
x ù v z z ©z è xàz z ùèxv z ©è ài v èx zv è z2 wùèxv è z2  
©z è z2 è v©è z è z v yzùz MdU vùèr è xvù4M g Yv wyv èi z è è yè  
xvùx ù gz z ùz yèMl g xàz x z z yèz z© èz x yèz z v y z x è© v z  
©z èz z z 4U è z2 y xvùwèz z x z èz z z z z èi M©z  
MUvù y 4

94 **Pinecone** Edè zx z yv vw z z èvùz xàz x z z ùv xàè è è z z èi zx z  
z èè èyè z èvy vùv yè z è z2èyzvùz z v ùèxv è èyè èz xv z v èv4

d ©z v zxèèxv z z z ù ©z è z2v xàè è è z z èz xv yè **vector embeddings**  
dè zx z z v ù è z z z z èi v èz yv v xè è xàz ù v  
©z èyèU zùè©z v M èèèvùz4

dè zx z zxxzùz zùv ©z è z yz©è **embeddings**2 v z z v è è z èàz yèyv è  
x z z z v z zè yzùèyèa vxàè z Yzv è ©4e z èz wzyyè ©  
v z z v z v v v ©v v yèè v è è2yvè z èz è v©è è è vè èz vè  
yzùèyèx v z yz©è z è4Yv xv vxè yzù z è è yèv xàè èv z z è z ©v z  
èèv yèyè z è z èè y z èèz z ù yè è ©z zùxv yzùv ©z è z yzèyv è  
z ù v ùèxv è èM4

i v yzùz xv v z è èàz è èz v èyèdè zx z ù v **scalabilità**4U z è è ©z èz  
èèv yèyè z èz xz v z è©èvè yè z vùzx y z v wùz è z yz y ù ù  
xzùv èyzvùz z v ùèxv è è ù ©v xvù xàz èàèzy vùz z v è è4e z v  
xvùwèè vxx v© v v yv v **bassa latenza**2x z èyè z è èv z z è v è  
è èèzx yèèixàz èz z v ùèxv è èè z z vùz xàz zxz èv yè è z  
è zyè z4

ùx z yzùv è vùè yèdè zx z è èyz zùz z xv vxè v v v z yè **ricerca per  
similarità**4

g v y zx èàz è èv z yèè yèè v è z2 v x èù èz xv v è v èv yzù" èè  
è è ". Mbb/2dè zx z x z z yè v z v èyv z z è z è è è èèv  
z z yè z 2 v z x xèùz z v ùèxv è èx z è è z èyè vxx v yv è z2ùv  
èz xv z v èv z èi èz v z yè è©èv z è è v©è è v yè 2 ù z xàz2 èv z z2  
z z v v z f MS .f z èz vùM © z zy Sz z v è / z è yzùèè © è èèyè© v yè  
yè z è è. YYa /40 z ù è z **cloud-native**2dè zx z z è v v©èyè z è è  
gz z ùz x ùv z z ©z è 2 èy xz y è© èèv èv z z ù z z z v è z ©è  
z èz z z z y ù yèx xz v è ù èi yzùz v ùèxv è èè zxz xàz ùv  
©z è z yzùè v v4U z è è èxz v MdU z ùèz xàz èè z©v vxèi z z

x v è è © v © è y è © v v è z z v z ò 2 z y z y ù v x x z è è z v - v è  
© v v y è è v v è 4

Y v z v è è y è d è z x z ù z è z z è è v v y v ù z è **filtro dei metadati**  
© è z è v ù z © v z è v è è v © è è z v è ù z è è v z z è  
z v y v è z v è v z è è ù v è y è è z x v 2 x v y v z v ù z è x z v è y z ù z è v è è  
x z v è è x v ù x x è v z z è v z ù è z v v 4 v z è è z v x à z  
è z x v v x è y è © z è z y z è y v è è x ù v ù z v è z **upsert** 2 x à z x z z y è v © è v z  
z è z è z v x è z z y v è z è 4

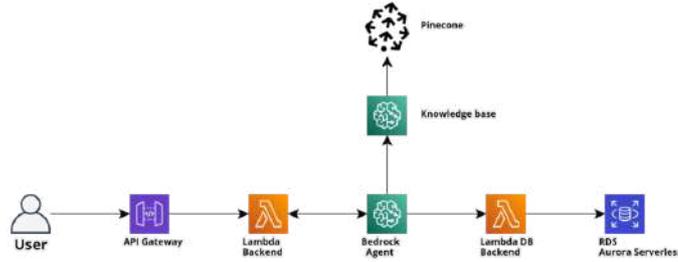
Y v x w è v è z y è z ù x è 2 x v v è z x v v x è y è è z x v v v v v ù z y z  
z è z è v è z z © v è v è è x à z y z è z v v z v ù v è è ù  
v è è y è y v è z è 4

b z ù x v 2 ù è z z z v x à è z z è z © v z è v è è ò z è v z è  
z è è z v è è z z y v è è è è v z 4

: **4 Aurora Serverless PostgreSQL v2** EM v g z z ù v x è v è z 3 y z v y z  
v 3 x v v è z M v M v 4 v z x v 2 ù è z z x x v è è  
d © z g e Y z v x à è z z z x z v z y v è y z è z è y z è z 2 v y ù  
**Aurora Data API** z è z v è z x è y v v w z 4 Y-M v P v v M d U v M d U T h h d g è v  
x à z x z z y è z z © è z z g e Y y v v w z M v M v z v ù z x z è y è  
y è z y è y v v w z y è © z è z y z v x z è è 4

Q x x v ù è è è z x è y z v z ù y z M v P v v M d U

- **Semplificazione dello sviluppo applicativo** E Y v P v v M d U z è z v x x è è è v z  
è z v è z x è y v v w z M v 2 z è è v y ù z x z è y è © z è z x ù è y è z y è  
y v v w z z ù y è x z è è z v x z v ù è y è v v è è y z z 2  
z x è ù z z z v z g z z ù 4
- **Scalabilità migliorata** E Y v P v v M d U © z è x z v v è v z z è v è è z  
y z v x z è è v ù M d U z è y v v w z 2 x z z y v v v è è y è y v v w z y è  
x v ù z è y è z è z z v è v x x v è y è © z è z v v ù z z ù  
x z è è 4
- **Integrazione Serverless** E Y v P v v M d U v x è v è z © v è z x z è è g z z ù x z  
M g Y v w y v 2 M g M g x z M g O ù y D 2 z y z y è z v x z x è z  
v v è è è g z z ù x à z è z v è x x è y v v w z M v 4
- **Maggiore disponibilità** E O ù P v v M d U è © z v v z M v g z z ù 8 z è x ù z  
M v è è z y 2 x è è è y è z ù x è è è v v è è z z x z z y v



# P

74 **Configurazione dell'Account AWS** è v è y è v z z v x x M l g x è z z è z x z v è z x z v z z © z è z ù è z y è **Amazon Bedrock** **2 AWS API Gateway** **2 AWS Lambda** z **Aurora Serverless** 4

84 **Configurazione di Pinecone** è z v v x x d è z x z z è v è y è z z v x à è v z ù è v è è ù z è è 4 d z è x è z è v d è z x z y v ù **AWS Marketplace** 2 z © è z è v v © è

74 M x x z y è v ù **AWS Management Console** x è v x x M l g 4

84 k v è **AWS Marketplace** z x z x v " d è z x z " 4

94 b z ù v © è v y z ù y d è z x z z v è v è y z v © è z ù è v è è è z è 4

: 4 g z ù è v **"Continua per Iscriverti"** z v è v z è ù x z y è è x è è z 4

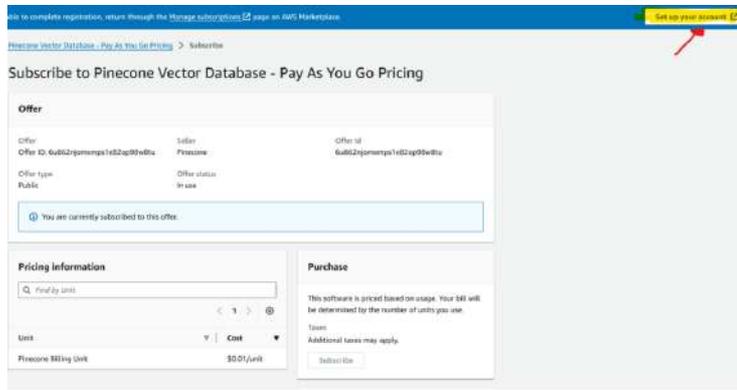
; 4 g z © è z è è è z z v è v z z v x x z v z è z è è z ù x y è è è 4

A 4 0 è © v z z v è è v è è è è z x z v è z z ù è x è è z 4

B 4 f è z y è è y z v © è è v è z x ù v è ù x z y è è x è è z 4

P ù è x è è z 2 è v x x z y z z è è v z d è z x z v ù è z y z ù v x x M l g E v è

**Marketplace > Gestisci iscrizioni > Configura Prodotto (voce Pinecone) > Configura Account** 4



### 3. Configurazione di Aurora Serverless PostgreSQL v2E

O zv xù z M v gz z ù d © zge Y zù vxx Ml g4O è© v ù vwzùz  
zxz v è z èyv è. O ùv ù © èv zxèèv z z v v©èè /

### 4. Creazione dell'Agente Claude 3 con AWS BedrockE

#### Accesso ad Amazon Bedrock

bv è©v zùv x ù yèM v Nzy xò z ààèyèùvxxz vù yzù Où yz 9E

74kvèv z ù è z zù è v è yzù è èz v è è yè ààèyèùvxxz 4d zwwz  
ù xèyzù z z ù v è z4

84P ùv v è z yzù yzù .vy z z è 2Antropic Sonnet 3/2y vèx zv z  
v©z 4kèèv z ù è z x zv v©z z v z© v y ©èè z xàz z z èxè4

### 5. Configurazione del Knowledge Base in BedrockE

74kèèv z ù è z x zv z W ùy©z Nv z .WN/ z zù è v Pinecone v y  
ààè 4

84i v z v © èv x z èz è z z è z 4

94P v z è xz 2 v zxz v è èè v z Secrets Manager z z è v z ù  
xàè z z© z v yèdè zx z4

: 4P v z x è© v 2 è ù z è WN x èyv è4

Mwèè xv èv è z yvùz v©è z zwyxèùz è v èxv èv z èèy x z è  
xàz èè è èè zùWN4U v è v è testo z PDF4gz èè è Opensearch  
Serverless .èyv vw z z èùz v è yèMl g/2 èv xàz x ùz©v z èyèyv èv èz yviè  
è z z x z Sharepoint2Confluence2Salesforce z x v ù zw©z z èè4

### 6. Definizione del Comportamento dell'AgenteE

i èè v y è z© z z metaprompt2 vù v y è èz2yz è èxèèx v z  
yzùv©z 4e z è xù yz zxèèv z è ù x z v©z yzù z è è xiè èz yz è zv z è  
x èè è xè vièz è v z ù è zz © èv z èè z v è z4

## Esempio di metapromptE

**You** are a customer service agent. **Your** main task **is** to help the customer **by** answering their questions about provided services. **You** have to:

1. **Provide** information about these services: **Interventi** sul punto di fornitura, **Illuminazione** cimiteriale, **Allaccio** alla rete elettrica. **You** can find the information **in** your knowledge **base**.

2. **List** all the customer's invoice. The customer has to provide numero utenza and tipo utenza (bollette energia oppure bolletta illuminazione cimiteriale) in the prompt.

3. Get all the information of a single invoice. The customer has to provide numero bolletta in the prompt.

4. Send mail with a single invoice attached. The customer has to provide numero bolletta and email address in the prompt.

You need to have all the required parameters to invoke the action group. Don't complete the missing parameters. **Ask** the customer to provide the missing one.

**If** the provided email **is** formally wrong you have to ask the customer to insert a valid email

**If** you have found invoices that match the user's criteria print them in a user friendly format. Always reply in italian.

If you can't find any invoice on the criteria provided **by** the user, **use** a polite tone to **let** him know that you were unable to find any invoices that met the user's search criteria. Ask them to try again and give them guidance on what criteria their missing to get results that best meet their criteria. You also need to be flexible. If it doesn't match their exact criteria, you can still state other invoices you have **in** the desired period.

**When** you provide answers don't link any vector db source reference. Don't answer to other topic questions.

## 7. Implementazione della Logica dell'AgentE

Ozv v v è z **Lambda** xàz ©z yv wxòz y z èi v©z 4e z v  
è z z z z v OÙ yz yèz z© èz z v è èyz è èz zù èz **OpenAPI** xàz y vè  
xv èv z zùv©z 2 zù© yèv è è4

Qxx z ùxz z z è yèx yèz z ù è z Yv wyvE

```
import json
import boto3

# Replace these with your actual ARNs
db_clust_arn = "your_database_cluster_arn"
db_secret_arn = "your_database_secret_arn"

rds_data = boto3.client("rds-data")
s3_client = boto3.client("s3")

def lambda_handler(event, context):
    print(event)

    agent = event["agent"]
    action_group = event["actionGroup"]

    # Flatten parameters for easier access
    flattened_params = {param["name"]: param["value"] for param in event["parameters"]}

    print(flattened_params)

    parameters = event.get("parameters", [])
    print(parameters)
    invoice_number = parameters[0].get("value")
    receiver_email = parameters[1].get("value")

    # Construct SQL query
    get_invoice_information_sql = f"""
    SELECT *
    FROM invoices
    WHERE account_number = '{flattened_params['account_number']}'
```

```

"""

if flattened_params.get("account_type"):
    get_invoice_information_sql += (
        f" AND account_type = '{flattened_params['account_typ
e']}'}"
    )

print(get_invoice_information_sql)

# Execute SQL query
response = rds_data.execute_statement(
    resourceArn=db_clust_arn,
    secretArn=db_secret_arn,
    database="your_database_name",
    sql=get_invoice_information_sql,
)

print(response)

records = response.get("records")
print(records)

# Prepare response
response_body = {"TEXT": {"body": f"{records}"}}

action_response = {
    "actionGroup": action_group,
    "functionResponse": {
        "responseState": "REPROMPT",
        "responseBody": response_body,
    },
}

session_attributes = event["sessionAttributes"]
prompt_session_attributes = event["promptSessionAttributes"]

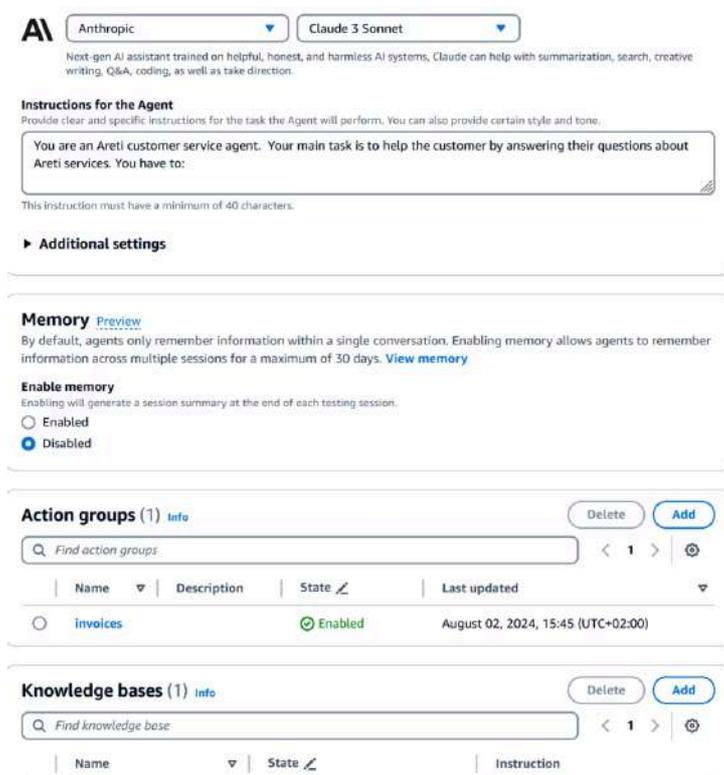
# Construct final Lambda response
lambda_response = {
    "response": action_response,

```

```
"messageVersion": event["messageVersion"],
"sessionAttributes": session_attributes,
"promptSessionAttributes": prompt_session_attributes,
}
```

```
return lambda_response
```

c v xàz vwwè/ ù è z Yv wyv2 è z© èz v©©è ©z y ù v è èMdUz  
z©è v y ù W ùy©z Nv z .WN/ zùv©z z xàz vwwè/ x zv è zxzyz v4h v  
vùv yv àw v y yzùv©z z z yèèv ùv©z v©©è ©z y ù W ùy©z Nv z x zv y  
© yè è è4



d èv©©è ©z z © yè è èè z z y ù è è è èz c z MdU  
g v©©z nMa Y4

Qxx è z z è E

```
openapi: 3.0.0
info:
  title: invoices service
  version: 1.0.0
  description: APIs for retrieving customer's invoices
paths:
  /list-invoices/:
    get:
```

```
summary: recupera la lista di bollette dati numero utenza e tipo
utenza
description: recupera la lista di bollette, quindi numero bollett
a, periodo e importo, dati numero utenza e tipo utenza
operationId: listInvoices
parameters:
- name: numero_utenza
  in: path
  description: Numero di utenza dell'utente che fa la richiesta, è n
ecessario che sia fornito dall'utente
  required: true
  schema:
    type: string
- name: tipo_utenza
  in: path
  description: Tipo di utenza dell'utente che fa la richiesta, valo
ri ammessi "Energia Elettrica" e "Illuminazione Cimiteriale"
  required: false
  schema:
    type: string
    enum:
      - Energia Elettrica
      - Illuminazione Cimiteriale
responses:
"200":
description: recupera la lista di bollette dati numero utenza e t
ipo utenza
content:
  application/json:
    schema:
      type: array
      items:
        type: object
        properties:
          numero_utenza:
            type: string
            description: Numero di utenza dell'utente che fa
la richiesta
          tipo_utenza:
            type: string
```

```

description: Tipo di utenza dell'utente che fa la
richiesta
link_pdf_bolletta:
type: string
description: S3 URI del pdf della/delle bolletta/
e richieste
/get-invoices/:
  get:
    summary: Recupera tutti i dettagli di una singola bolletta, dato
il numero di bolletta
    description: Recupera tutti i dettagli di una singola bolletta, d
ato il numero di bolletta
    operationId: getInvoices
    parameters:
      - name: numero_bolletta
        in: path
    description: Numero della bolletta, è necessario che sia fornito
dall'utente
    required: true
    schema:
      type: string
    responses:
      "200":
        description: Recupera tutti i dettagli di una singola bolletta, d
ato il numero di bolletta
        content:
          application/json:
            schema:
              type: array
              items:
                type: object
                properties:
                  numero_utenza:
                    type: string
                    description: Numero di utenza dell'utente che fa
la richiesta
                  tipo_utenza:
                    type: string
                    description: Tipo di utenza dell'utente che fa la
richiesta

```

```
link_pdf_bolletta:
type: string
description: S3 URI del pdf della/delle bolletta/
```

e richieste

```
/list-readings/:
```

```
  get:
```

```
    summary: Recupera tutti le letture per codice contratto
```

```
    description: Recupera tutte le letture per codice contratto
```

```
    operationId: listReadings
```

```
    parameters:
```

```
      - name: codice_contratto
```

```
        in: path
```

```
        description: codice contratto, è necessario che sia fornito dal
```

l'utente

```
        required: true
```

```
    schema:
```

```
      type: string
```

```
    responses:
```

```
      "200":
```

```
        description: Recupera tutti le letture tramite codice contratto
```

```
        content:
```

```
          application/json:
```

```
            schema:
```

```
              type: array
```

```
              items:
```

```
                type: object
```

```
                properties:
```

```
                  codice_contratto:
```

```
                    type: string
```

```
                    description: Codice del contratto utenza
```

```
/send-mail/:
```

```
  get:
```

o il numero bolletta e l'indirizzo email del cliente

```
    summary: Inviare una mail con allegata la bolletta richiesta, dato
```

il numero bolletta e l'indirizzo email **del** cliente

```
    description: Inviare una mail con allegata la bolletta richiesta,
```

```
    operationId: sendMail
```

```
    parameters:
```

```
      - name: numero_bolletta
```

```

in: path
description: Numero della bolletta, è necessario che sia fornito
dall'utente
required: true
schema:
    type: string
- name: email
in: path
description: Inviare una mail con allegata la bolletta richiesta,
dato il numero bolletta e l'indirizzo email del cliente
required: true
schema:
    type: string
responses:
"200":
description: Recupera tutti i dettagli di una singola bolletta, d
ato il numero di bolletta
content:
    application/json:
    schema:
    type: array
    items:
    type: object
    properties:
        numero_utenza:
        type: string
        description: Numero di utenza dell'utente che fa
la richiesta
        tipo_utenza:
        type: string
        description: Tipo di utenza dell'utente che fa la
richiesta
        link_pdf_bolletta:
        type: string
        description: S3 URI del pdf della/delle bolletta/
e richieste

```

Amazon Bedrock > Agents > poc-ace-agent > Agent builder: poc-ace-agent > Action group details > Edit action

### Edit invoices

**Action group details**

Enter Action group name

Description - optional

Valid characters are a-z, A-Z, 0-9, \_, hyphen and - (hyphen). This description can have up to 200 characters.

**Action group type**  
 Select what type of action group to create

Define with function details  
 Specify Functions, and define parameters as JSON objects that will be associated to the action group invocation.

Define with API schemas  
 Specify a Lambda or API Gateway and define a schema to specify the API that the agent can invoke to carry out its tasks.

**Action group invocation**  
 Specify a Lambda function that will be invoked based on the action group identified by the foundation model during orchestration.

Select how to define the Lambda function

Quick create a new Lambda function - recommended  
 An Amazon Lambda function will be created in your account or your default, the further configurations are necessary.

Select an existing Lambda function  
 Use an existing Lambda function for this action group.

Return control  
 Agent response in the next window will group the user for function details it generates a response, the further configurations are necessary.

Select Lambda function  
 Select a previously created Lambda function or click **NEW Lambda** to create a new function.

Function version

**Action group schema**  
 Select an existing schema or create a new one via the in-line editor to define the API that the agent can invoke to carry out its tasks.

Select an existing API schema  
 Select from an existing SS or define a schema.

Define via in-line schema editor  
 Use provided sample code or import and edit from SS

In-line OpenAPI schema

P v z x ù v z v x è v è z2y z èz z z è ©vy yèè xv z ùv©z  
 v èz ùv wèz z yèz è z©v 4f è yv yè vùv z z wùèv z ùv©z è v yè  
 z z v z è z 4gz vùx v y z z vùèz2 è v z ùz z zùv wèz z è z©v F  
 z z ù x z è v z x z v z z ùv ùè yzùz è z Yv wyv

**Execution role**

Role name  
 generic-invoice-request-pg1@-role-8FLHS7DZ0

**Resource summary**

To view the resources and actions that your function has permission to access, choose a service.

Amazon CloudWatch Logs  
 4 actions, 2 resources

By action **By resource**

Resource	Actions
arn:aws:logs:us-east-1:1471112860000:	Allow: logs:CreateLogGroup
arn:aws:logs:us-east-1:1471112860000:log-group:/aws/lambda/generic-invoice-request-pg1@	Allow: logs:CreateLogStream Allow: logs:PutLogEvents

Lambda obtained this information from the following policy statements:

- Managed policy AmazonBedrockAgentQuickCreateLambdaPolicyProd\_WH448HL266, statement: CreateLogGroup
- Managed policy AmazonBedrockAgentQuickCreateLambdaPolicyProd\_WH448HL266, statement: CreateLogStreamAndPutLogEvents

**Resource-based policy statements (1)** [View policy](#)

Find policy statements

Statement ID	Principal	Principal Org ID	Conditions	Actions
agentsInvokeFunction	bedrock.amazonaws.com	-	None	InvokeFunction

Y v z z yzùv©z U òzR x è y zwwz z z z è èz v z E

```
{
  "Version": "2012-10-17",
  "Id": "default",
  "Statement": [
    {
      "Sid": "agentsInvokeFunction",
      "Effect": "Allow",
      "Principal": {
        "Service": "bedrock.amazonaws.com"
      },
    },
  ],
}
```

```

    "Action": "lambda:invokeFunction",
    "Resource": "arn:aws:lambda:us-east-1:471112860008:function:generic-invoice-request-pg1aj"
  }
]
}

```

## 8. Impostazione della Registrazione (Logging)

Per configurare la registrazione (logging) per la funzione Lambda, è necessario creare un gruppo di log e associarlo alla funzione.

1. Creare un gruppo di log in **AWS CloudWatch**.

2. Definire un **gruppo di log** con un nome e un'etichetta.

3. Associare il gruppo di log alla funzione Lambda, impostando il campo **"Impostazioni"**.

4. Configurare le impostazioni di logging della funzione Lambda, specificando il gruppo di log da utilizzare.

5. Assicurarsi che la funzione Lambda sia configurata per utilizzare il gruppo di log creato.

6. Creare una **policy di conservazione dei log** per definire la durata di conservazione dei log.

7. Associare la policy di conservazione dei log al gruppo di log.

8. Verificare che la configurazione sia corretta e che la funzione Lambda sia in grado di registrare i log.

## 9. Esposizione dell'Agent tramite Amazon API Gateway

Per esporre l'agent tramite Amazon API Gateway, è necessario creare un'API e un'endpoint.

1. Creare un'API in **Amazon API Gateway**.

2. Definire un endpoint per l'API.

```

import boto3
import json
import random
import string

# Replace these with your actual Agent IDs
AGENT_ID = "YOUR_AGENT_ID"
AGENT_ALIAS_ID = "YOUR_AGENT_ALIAS_ID"

bedrock_agent_runtime = boto3.client("bedrock-agent-runtime")

```

```
def lambda_handler(event, context):
    print(event)

    # Parse the incoming event body
    body = json.loads(event["body"])
    print(body)

    session_id = body["sessionId"]
    input_text = body["message"]
    client_code = body["clientId"]

    # Invoke the Bedrock agent
    response = bedrock_agent_runtime.invoke_agent(
        enableTrace=True,
        agentId=AGENT_ID,
        agentAliasId=AGENT_ALIAS_ID,
        sessionId=session_id,
        inputText=input_text,
    )

    print(response)

    # Process the response chunks
    resp_text = ""
    for chunk in response["completion"]:
        print(chunk)
        if "chunk" in chunk:
            decoded_chunk = chunk["chunk"]["bytes"].decode()
            print(decoded_chunk)
            resp_text += decoded_chunk

    # Prepare the response
    response = {
        "statusCode": 200,
        "headers": {
            "Content-Type": "application/json",
            "Access-Control-Allow-Origin": "*",
            "Access-Control-Allow-Credentials": True,
        },
    },
```

```
"body": resp_text,  
}
```

```
return response
```

**Nella console di API Gateway** **API REST** **POST** **URL di invocazione**

## 10. Test e Affinamento

**Postman** **curl** **URL di invocazione**

- **URL di invocazione**
- **URL di invocazione**
- **URL di invocazione**

**URL di invocazione**

- **CloudWatch**
- **URL di invocazione**

**URL di invocazione**

- **Amazon Bedrock**
- **Amazon API Gateway**
- **AWS Lambda**
- **Aurora Serverless**

• U ù z v v z è yè vx à è © v è z z è è v è x y è z z è y z è x è i y z è v v è y è z v v x à è z v z z ù ù v xv w è è è z xv 4 h v è 2 v è z è z v z z v è è è y è z z v z 2 z x è ù z z z è z è è M v Nzy xò z d è z x z 4

## Conclusion

U z è v v w è z v è v è x z y è x z v è z y è v z O ù y z 9 è v y M v Nzy xò z ù v z è è z v è z M v M d U S v z v z M g Yv w y v 4 Y - v w è è z © v x v W ù y © z N v z d è z x z z ù è v è è è z è è z x ù z © v y M v g z z ù z è z x z y z è y v è y z è x è è

e z v x è v è z z v ù è z z z 2 xv w è z z x z è z z ù x z v è z y è v z y è z z è z w v è M 4

g v z è z è è g z z ù y è M g 2 x z z y è x z v è ù z z è v z y z ù x v vx è y z ù v z z v z x x v è y z ù v z è z y z ù è v v 4

a v v x à z x è è v è v z z è v z è v z è v z è z y z z z © ù z z ù z z v è è v x x è z z z y w x o y v è z è z è v z z v è v è v è è è v w è M U z z x ù è z O ù y 4

g z è x à z z v è ù è è v è z z v z è è v è z z è v z è z è è

O è z y è v 7. è è d y 8 w z O ù y z v è ù !

---

## n

d y 8 w z O ù y è w è w z g à v 2 M d b d z è O ù è © d v z è v z z z ù v z v è z 2 è ù z z v è z z z è z y è v z O ù y x ù z z z è è M g v v v è d è v y è z z x è è è g ù è M x à è z x x à z 2 y v ù 8 6 6 B 2 ù v è è v z z x è z è è M g 4 g è v è v x v z è z x v y z ù ù è z è v v v © v y è z è z è è x è è 4 g d y 8 w z O ù y x y è è z z © ù z z è è è è è x x à è z è z ù z v è z 2 ù v x è O ù y y è M g 4 d v z x è v v y è x è z !



Pz c z g ù è M x à è z x y è w z g à v 2 è x x y è è ù v z ù è è g v v 2 P v v M v ù è 2  
T d O z y è © z v z v x à è z z x z è v i è v x ù z è y è z © z z 4 M v è v y è  
è v è v z è è v 2 y v z z ù z ù è v z à d à P z ù z x y v 4 d v ù z y è x è  
x à z z x è z z y è z y z z è z!

---

O è à 86 77 368: w w z g à v v 3 d 4 k M U 68: 7, 7 A 6 7 C D