Introducción a Cognito y Step Functions

David Lay,
Nodejs Developer, Globant

Ian Sebastian
Nodejs Developer, Globant

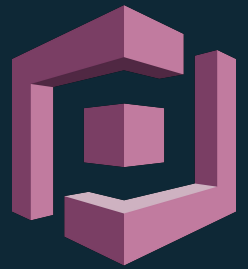
28 Agosto, 2019

Amazon Cognito

Globant
we are ready

aws

Overview



- AWS access control service
- “Easy” implementation of authentication at app level
- Social and enterprise user federation
- End user IAM roles association for protecting resources
- Open standards like Oauth 2.0, OpenID Connect, SAML
- Possibility to add MFA protection (SMS, OTP)
- Support multiple compliance programs (HIPAA, PCI DSS, SOC, ISO 9001)
- Cheap! (First 50.000 Monthly active users are free. After, each $0.00550)

Globant
we are ready

aws

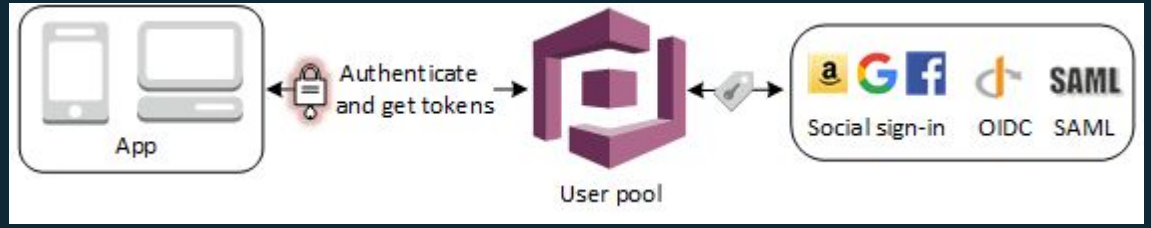
User Pools

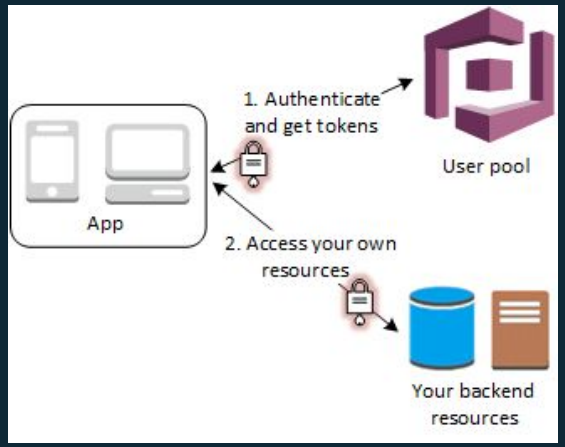
vs

Identity Pools

*“**User pools** es un proveedor de identidad (IdP). Se puede utilizar para **autenticar** usuarios en aplicaciones móviles, sitios webs, y administrar usuarios.*

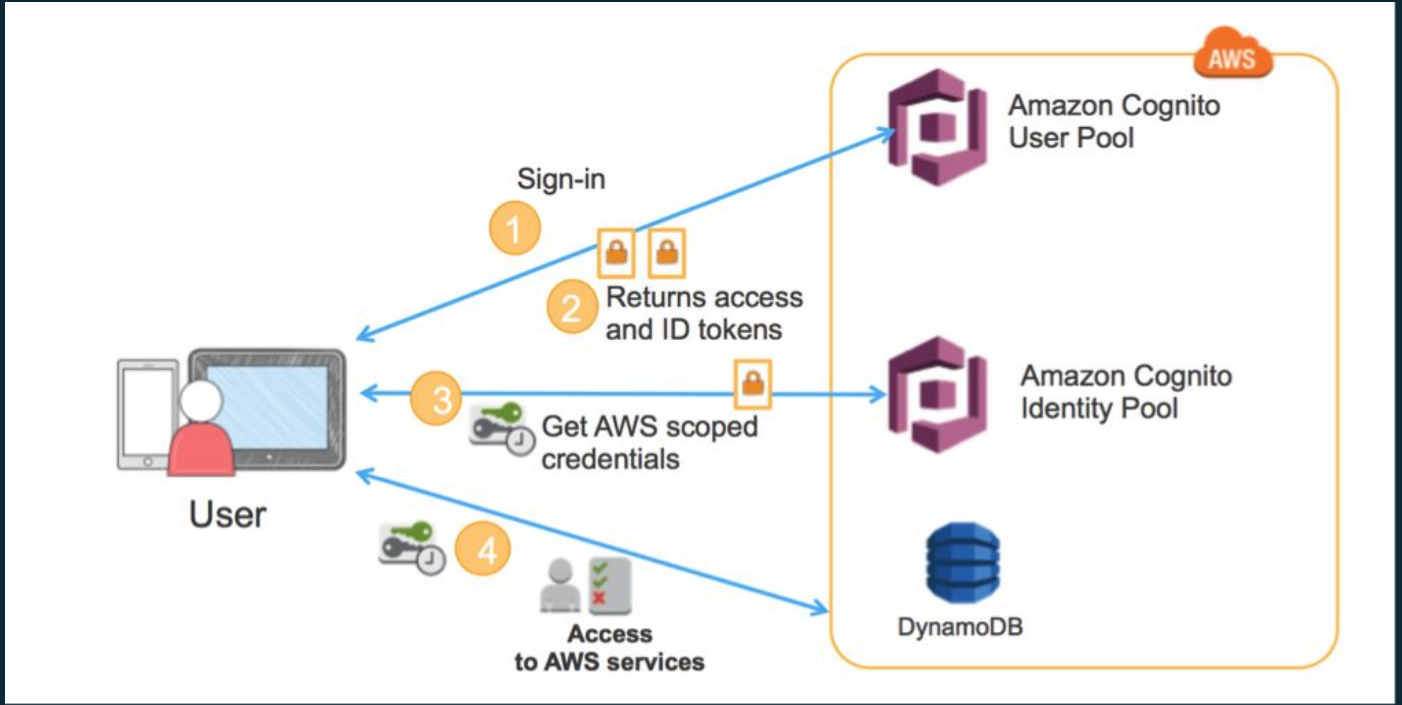
***Identity pools** son utilizadas para **autorizar** accesos a los recursos de AWS, tales como IAM, S3, EC2, EKS, etc. Ésto significa que se puede configurar un identity pool con varios identity providers (facebook, google, cognito) para dar acceso a usuarios externos sin ningún tipo de registro”*





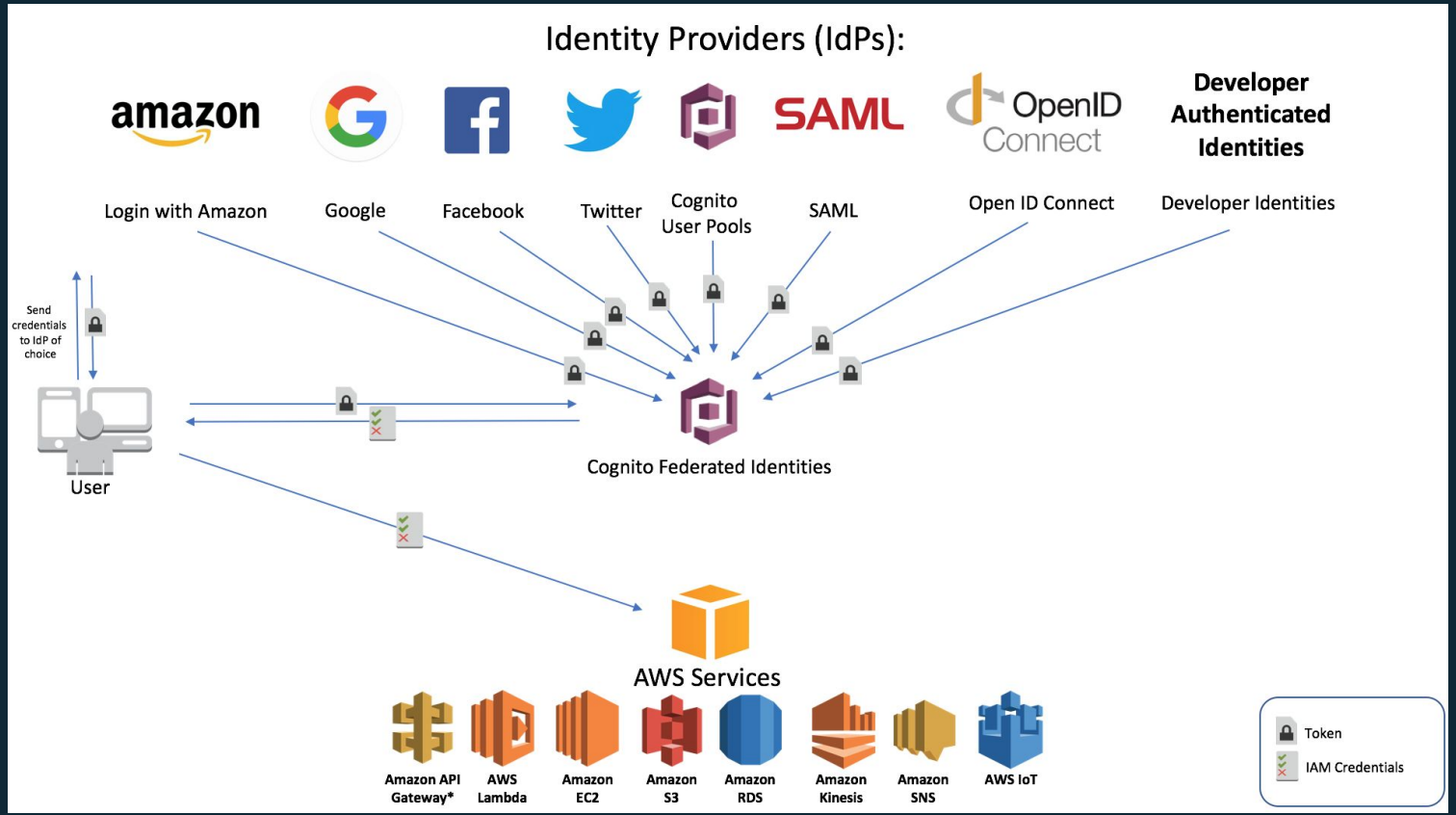
Globant
we are ready

aws



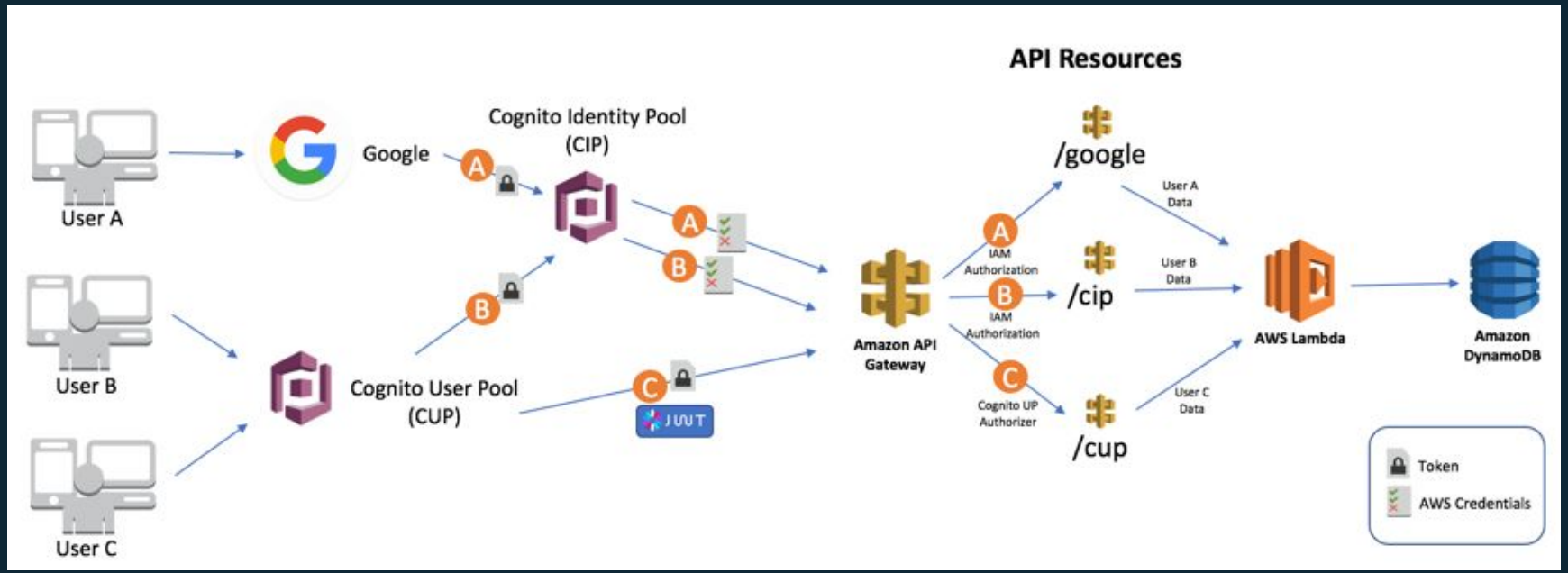
Globant
we are ready

aws



Globant
we are ready

aws



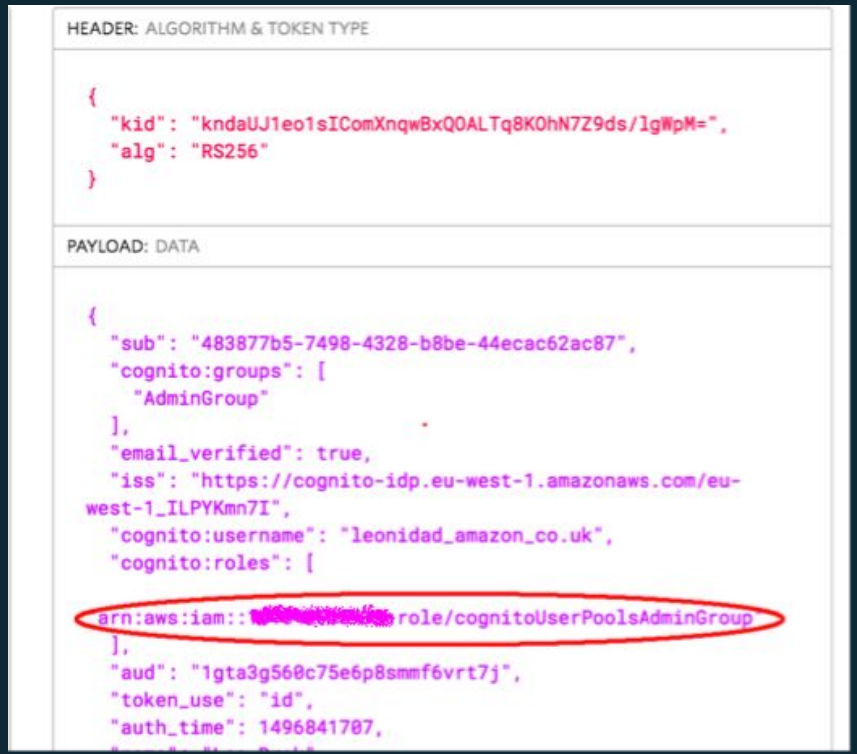
Globant
we are ready

aws

Features: Creando un User Pool

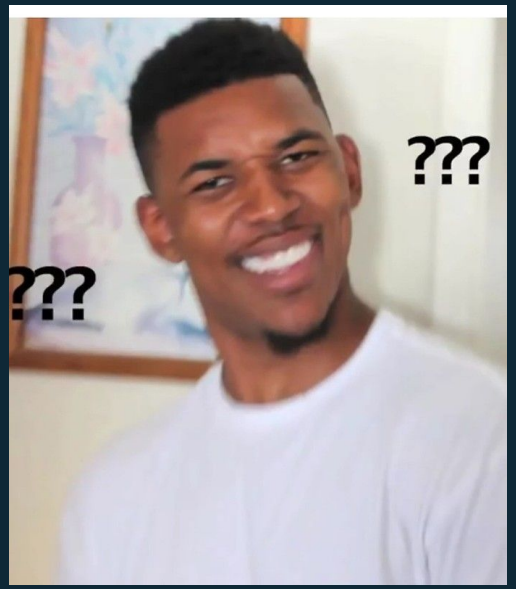
User pools: obteniendo tokens

- Amplify, AWS SDK, Cognito web UI
- OpenID Connect
- Access Token
- Id Token
- Refresh Token



Globant
we are ready

aws



So, what do I do with the tokens?

Globant
we are ready

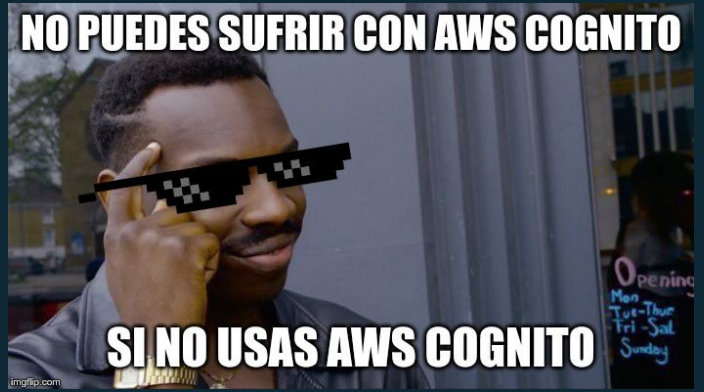
aws

Diferentes estrategias

- Intercambio de tokens por credenciales asociadas a IAM roles (identity pools)
- Manejo de tokens y storage de los mismos a través de backend (docker, AWS SDK)
- Envío de tokens directo a app cliente a través de web UI (access + id tokens)

Algunas reflexiones

- La documentación es mala. Expect headaches
- Conocimientos sobre Oauth 2.0 y OpenID Connect son casi no opcionales, sino necesarios.
- Aim for KISS, entonces itera
- Si puedes no usar Cognito, usa Amplify



Globant
we are ready

aws

Keep calm and breathe

Erase una vez

- Empresa mediana, creciendo rápido
- Onboarding necesita mejoras
- Integración con trello

Globant
we are ready

aws

La Misión

- Devs deben completar ciertas tareas en trello
  - HR necesita una tarjeta por candidato
    - Recordatorios diarios por email

La Misión

- Apis externas (trello)
- Tiempos de espera (recordatorios)
- Uso esporádico
- Parece ideal para serverless!

Serverless

¿Pero como coordinamos?

¿Llamamos lambdas desde lambdas?

¿y los tiempos de espera?

David Lay M

@davidlaym

www.davidlaym.com

Globant
we are ready

aws

Enter step functions

- Transiciones y estados
- Control de flujo
- Input / Output

Estados

- Task (ejecuciones, lambdas)
- Choice (decisiones)
- Wait (esperas)
- Parallel (ejecución en paralelo)

Estados

```
"TaskState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-1:1234556788:function:hello-world",
  "Next": "NextState",
  "TimeoutSeconds": 300
}
```

Globant
we are ready

aws

Estados

```
"TaskState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-1:1234556788:function:hello-world",
  "Next": "NextState",
  "TimeoutSeconds": 300
}
```

Globant
we are ready

aws

Estados

```
"Some Choice": {
  "Type": "Choice",
  "Choices": [
    {
      "Variable": "$.value",
      "StringEquals": "something",
      "Next": "Next State A"
    }
  ],
  "Default": "Next State B"
},
```

Globant
we are ready

aws

Estados

- Try / Catch (manejo de errores)
- Retry (reintentar)

Estados

```
    "Type": "Task",
    "Resource": "arn:aws:lambda:us-east-1:1234556788:function:hello-world",
    "Next": "NextState",
    "Retry": [{
      "ErrorEquals": [ "ErrorA", "ErrorB" ],
      "IntervalSeconds": 1,
      "BackoffRate": 2.0,
      "MaxAttempts": 2
    }, {
      "ErrorEquals": [ "ErrorC" ],
      "IntervalSeconds": 5
    }
  ],
  "Catch": [{
    "ErrorEquals": [ "ErrorA", "ErrorB", "ErrorC" ],
    "Next": "RecoveryState"
  }, {
    "ErrorEquals": [ "States.ALL" ],
    "Next": "TerminateMachine"
```

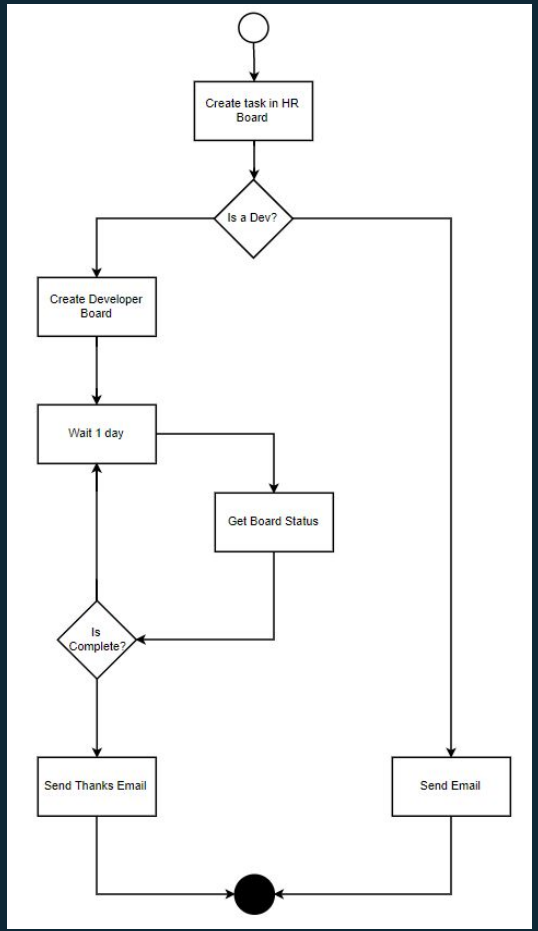
Globant
We are ready

aws

Volviendo

- Modelado de flujo simple

(link)



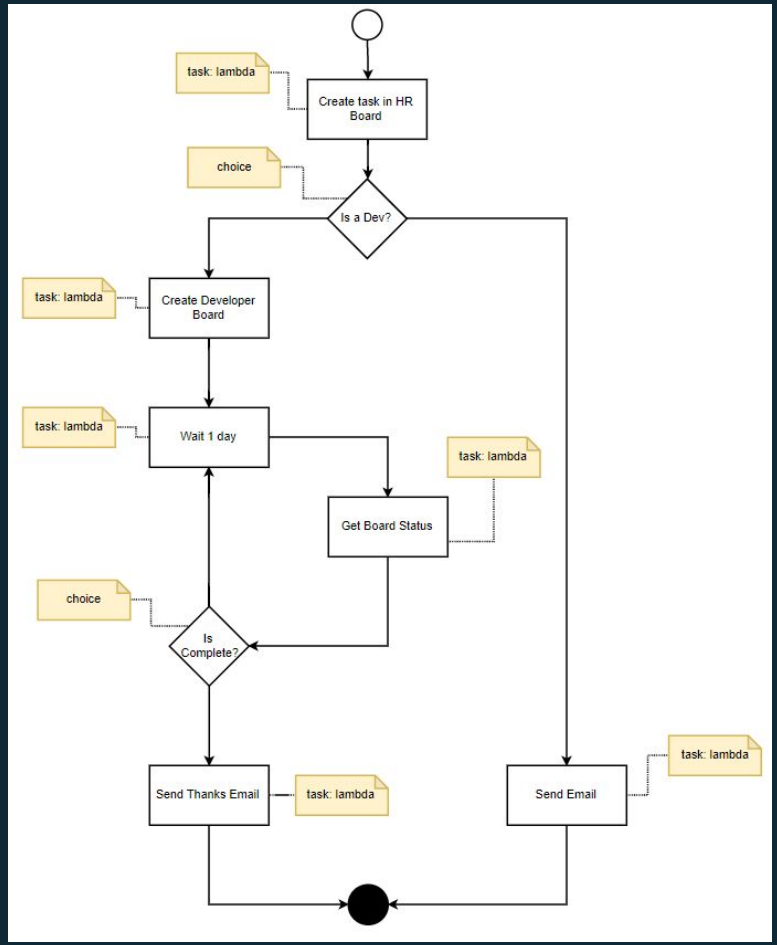
Globant
we are ready

aws

Hum..

Cada parte del diagrama se traduce a un estado de step functions

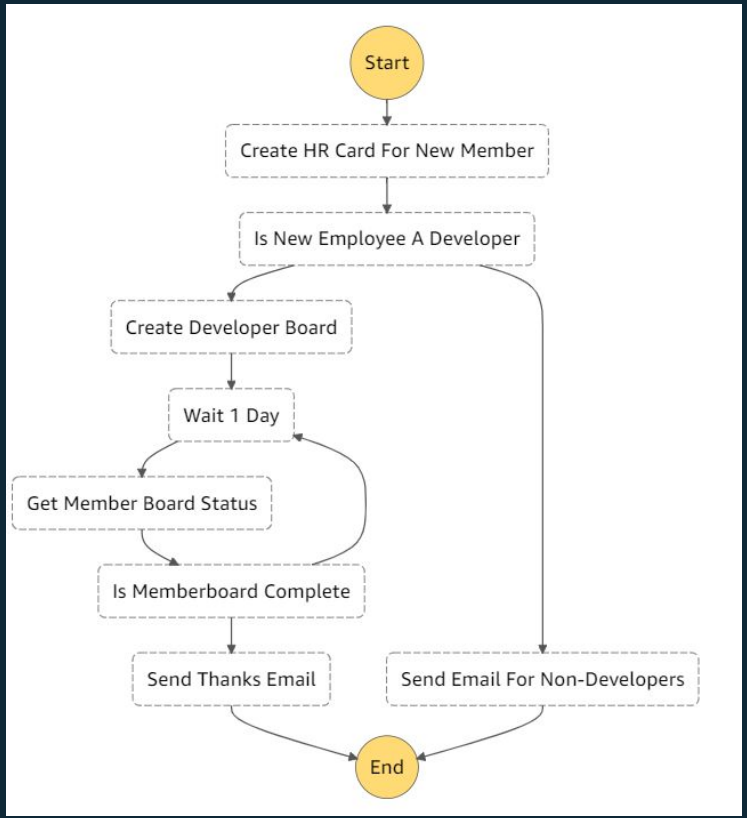
(link)



Globant
we are ready

aws

Demo



Globant
we are ready

aws

Beneficios

- Soporta ejecuciones que demoren hasta 1 año
- Fácil de configurar y ejecutar
- Integración con servicios AWS
- Desarrollo local (Docker, SAM)

Problemas

- Limite en cantidad de eventos (25k), cada estado genera varios eventos.
- Logs y monitoreo no es tan sencillo
- Puede volverse costoso rápidamente

Veredicto

- Límite en cantidad de eventos (25k), cada estado genera varios eventos.
- Logs y monitoreo no es tan sencillo
- Puede volverse costoso rápidamente

Gracias!

david.lay@globant.com
ian.sebastian@globant.com

Globant
we are ready

aws