

Golang para DevOps

Miguel Angel Torres Govea
SG Virtual Conference
Software Guru



SG
VIRTUAL
CONFERENCE

Contenido de la Platica

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones





1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones

Acerca de mi

Y de mi trabajo...

Ingeniero en Sistemas Computacionales
ESCOM - IPN

PriceTravel

Arquitecto de Software & Líder SysAdmin Sr VoIP/CTI

Universidad de Quintana Roo

Profesor impartiendo Temas Selectos de VoIP, Redes Inalámbricas, Sistema operativos basados en Unix, Sistemas Operativos de Red, Diseño de Redes

Universidad Aztlan

Profesor de Diplomado en Desarrollo Web



Golang

¿Qué es?

Lenguaje de alto nivel de programación, escrito estáticamente y compilado

Creado por Google en 2009 por Robert Griesemer, Rob Pike y Ken Thompson.

Diseñado para simplicidad y eficiencia.

También conocido como Golang

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



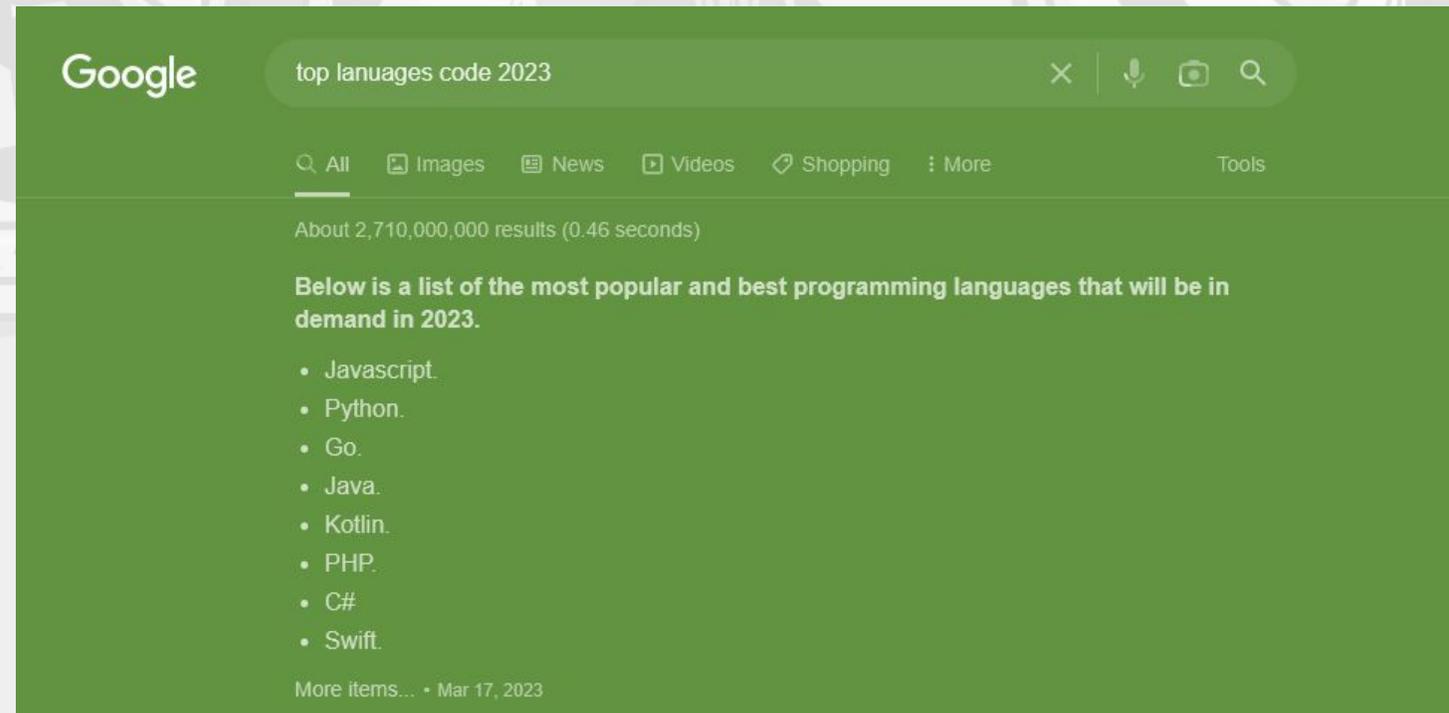
Golang



¿Por qué para DevOps?

Uno de los lenguajes mas utilizados recientemente.

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Google top languages code 2023

All Images News Videos Shopping More Tools

About 2,710,000,000 results (0.46 seconds)

Below is a list of the most popular and best programming languages that will be in demand in 2023.

- Javascript.
- Python.
- Go.
- Java.
- Kotlin.
- PHP.
- C#.
- Swift.

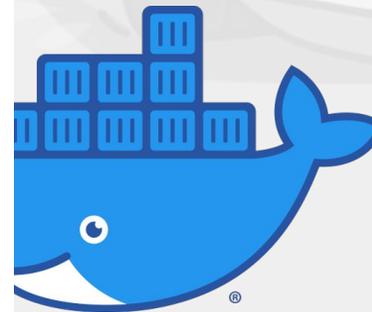
More items... • Mar 17, 2023



Golang

¿Por qué para DevOps?

Es ampliamente usado por varis proyectos como Docker, Telegraf, InfluxDB, CockroachDb, Dgraph, etc.



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones

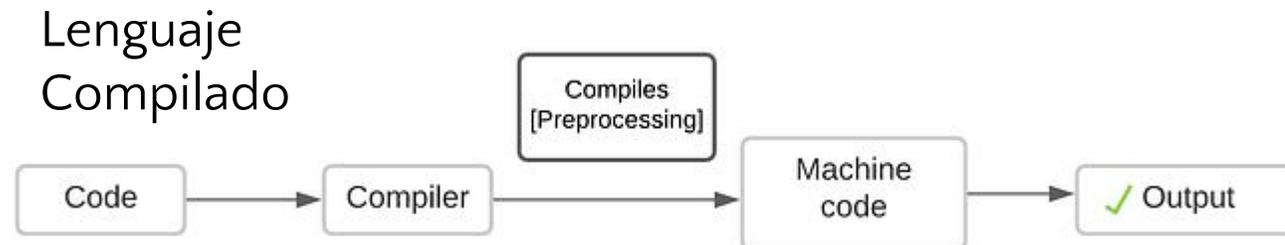
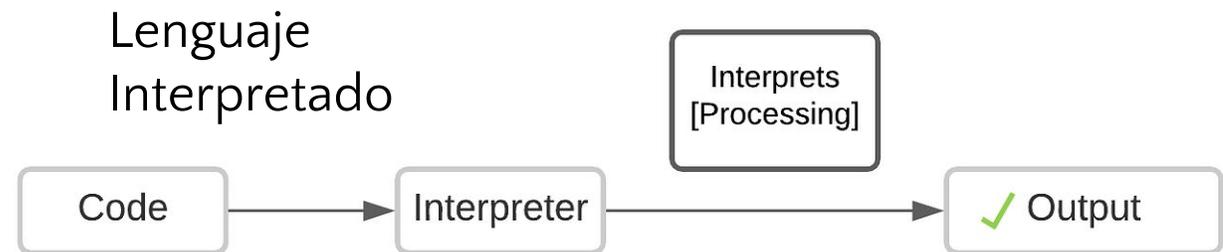


Golang



¿Por qué para DevOps?

Es un lenguaje compilado y no interpretado como algunos otros lenguajes permitiendo que se ejecute mas rápido ya que consume menos tiempo en procesador.



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Golang



¿Por qué para DevOps?

Se implementa la habilidad de poder importar módulos y librerías al código que escribimos y añadir funcionalidades a este.

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Golang

¿Por qué para DevOps?

Varias Compañías como Uber, Netflix, Drobbox y Soundcloud lo usan para diferentes aplicaciones como:

Sistemas Distribuidos, su implementación de concurrencia lo hace perfecto para esta tarea.

Redes, permite subir configuraciones de dispositivos o controlarlos.

Aplicaciones Web, soporta nativamente el protocolo HTTP.

MicroServicios, su tiempo de compilación rápida y eficiencia de ejecución lo hacen per perfecto para esta tarea.

DevOps, poder controlar y configurar mas eficiente nuestra infraestructura de Cloud u OnPremise.



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Golang



Beneficios

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones

Concurrencia, corre parte del programa n cantidad de veces y estas partes se comunican entre si.

OpenSource, es un lenguaje no privativo y permite colaborar con el desarrollo del proyecto.

Multi Plataforma, permite generar binarios para diferentes arquitecturas.

Rápida ejecución, el tiempo de ejecución es relativamente corto en varios programas.



Golang



Beneficios

Escalable, permite crecer el código de una forma mas sencilla.

Diseñado para la Nube, este lenguaje fue diseñado para implementarse directamente en servidores.

Manejo de Memoria, se hace una gestión automática del uso de memoria por cada proceso.



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Golang



Desventajas

No cuenta con GUI, no hay librerías estándar para generar interfaces de usuario..

No soporta OOP, técnicamente no soporta Programación Orientada a Objetos pero se puede implementar de una forma.

No Soporta Herencia, no permite implementar herencia en el código.

1. Acerca de mi

2. ¿Qué es Golang?

3. ¿Por qué Golang para DevOps?

3.1. Beneficios

3.2. Golang vs otros Lenguajes

4. Usando Golang en Devops

4.1 AWS

4.2 Azure

4.3 Mando

5. Conclusiones



Golang



Comparación con Otros Lenguajes

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones

Language comparison

	Python	Ruby	JS/ Node.js	C/C++	Java	Go
semicolons	N	N	Y	Y	Y	N
curly braces	N	N*	Y	Y	Y	Y
static types	N	N	N	Y	Y	Y
easy-to-use concurrency	N	N	Y	N	N	Y
multi-core concurrency	N	N	N	Y	Y	Y
compiled	N	N	N	Y	Y	Y
OO: classes, inheritance	Y	Y	Y	Y	Y	N*



Golang



Comparación con Otros Lenguajes

	ADVANTAGES	DISADVANTAGES
Go (Golang)	<ul style="list-style-type: none"> + Fast execution speed + Minimal runtime errors due to static typing + Concurrent + Cross-platform + Built-in garbage collection + Single executable 	<ul style="list-style-type: none"> - Less flexible due to static typing - Limited developer familiarity
Python	<ul style="list-style-type: none"> + Modular + Flexibility due to dynamic typing + Object-oriented + Extensive third-party libraries and modules 	<ul style="list-style-type: none"> - Slow execution speed - Runtime errors due to dynamic typing - Lower memory efficiency - Potentially complicated runtime environment setup
Scala	<ul style="list-style-type: none"> + Similar syntax to Java + Support for highly scalable/concurrent systems + Immutable objects + First-class functions 	<ul style="list-style-type: none"> - Runs in the JVM, which can cause restraints, depending on runtime environment - Limited developer familiarity - Smaller community
Ruby	<ul style="list-style-type: none"> + Large and active community + Extensible with many third-party modules + Built-in garbage collection 	<ul style="list-style-type: none"> - Lower performance than compiled languages, notably in boot and execution speed - Documentation less widely available - Tight Active Record coupling
C&C++	<ul style="list-style-type: none"> + Fast execution speed + Extensive developer base and documentation + Extensive feature set, depending on language choice 	<ul style="list-style-type: none"> - No garbage collection - Long build times - Different compiler requirements per OS - ABI incompatibility due to compiler requirements

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Usando Golang en DevOps



Slack

> Crear notificaciones



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Usando Golang en DevOps



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones

AWS

- > Lanzar una instancia EC2
- > Manipular S3 Buckets

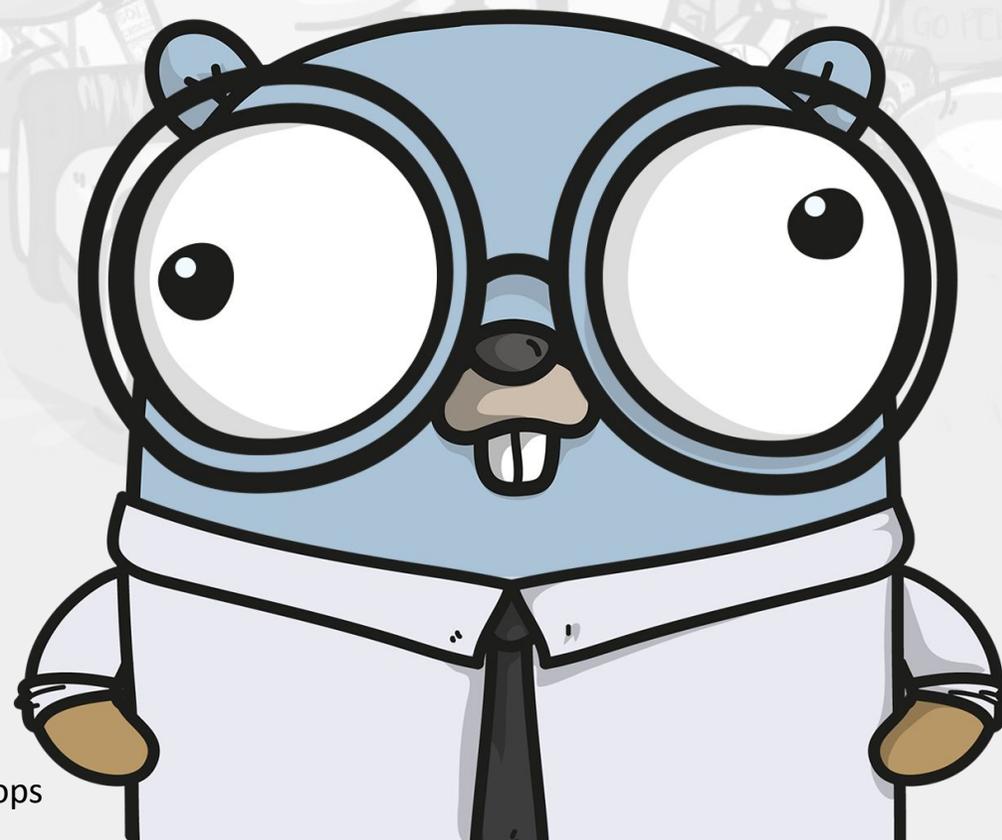


Usando Golang en DevOps



Azure

>Lanzar Maquina Virtual.



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones

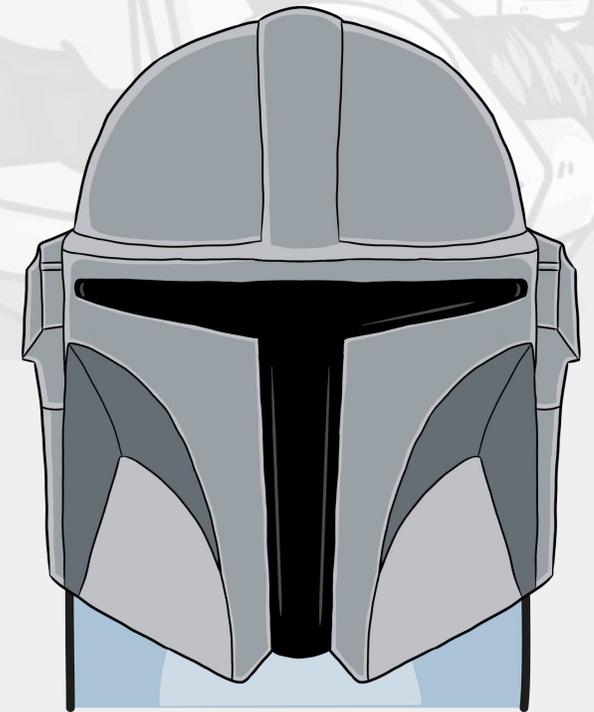


Usando Golang en DevOps

Mandopher

This is the way....

Herramienta desarrollada en golang para deployar API's y procesos hechos en golang.



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones

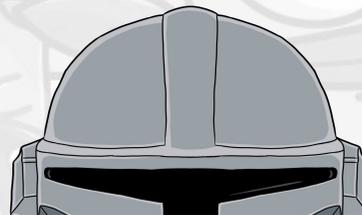


Usando Golang en DevOps

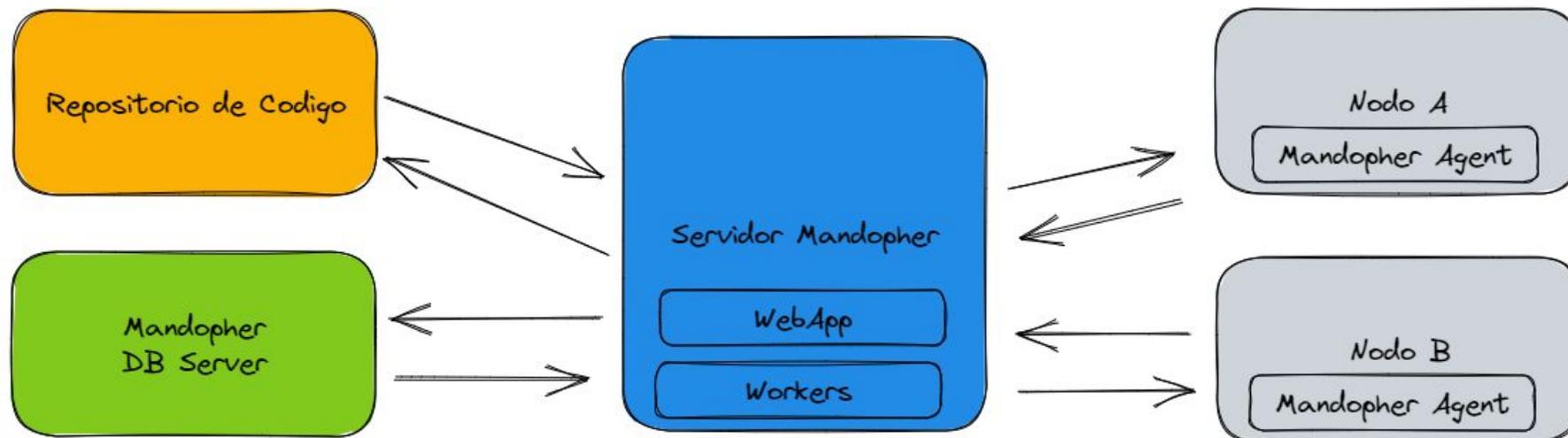


Mandopher

github.com/mafairnet/mandopher
Release Beta 1-Jul-2023



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Golang para Devops



Conclusiones

Golang es una muy buena herramienta para alguien que se dedica al OnPremise y el Cloud para poder automatizar y crear código que le permita manejar de forma eficiente toda su infraestructura.

Al mismo tiempo le da la habilidad de tener más control y funcionalidad sobre lo que hace.



1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



Q&A

GO

1. Acerca de mi
2. ¿Qué es Golang?
3. ¿Por qué Golang para DevOps?
 - 3.1. Beneficios
 - 3.2. Golang vs otros Lenguajes
4. Usando Golang en Devops
 - 4.1 AWS
 - 4.2 Azure
 - 4.3 Mando
5. Conclusiones



C'est fini

Miguel Angel Torres Govea

LinkedIn, Github: mafairnet

Correo: miguel@maf.mx, mafairnet@gmail.com

Telefono: 5513044149

Pagina Web: www.maf.mx

Material de la platica: <https://github.com/mafairnet/golang-para-devops>

The logo for the SG Virtual Conference is displayed on a white rectangular background. It features the letters 'SG' in a large, bold, green font. To the right of 'SG' is a green silhouette of a globe. Below 'SG' and the globe, the words 'VIRTUAL' and 'CONFERENCE' are stacked in a smaller, green, sans-serif font. The entire logo is set against a solid green background that occupies the right half of the image.

SG 
VIRTUAL
CONFERENCE