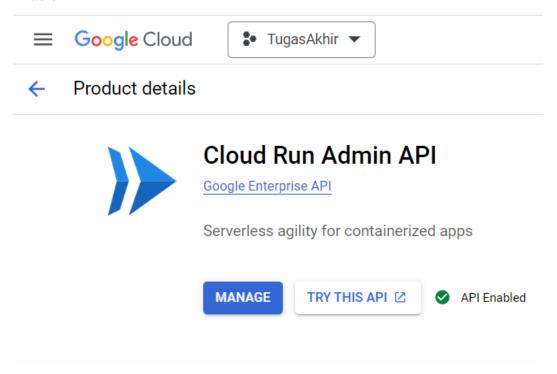
Tahapan Deploy Model (Docker) Kedalam Cloud Run

Sebelum melakukan deployment kedalam cloud run ada beberapa hal yang harus dipastikan :

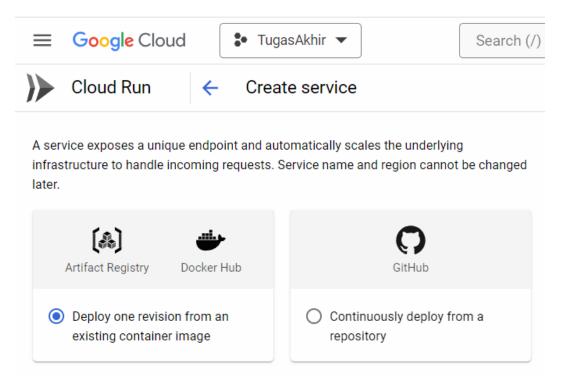
- 1. Sudah memiliki Akun Docker Hub
- 2. Sudah membuat API yang terintegrasi dengan Docker
- 3. Sudah melakukan Build Image docker dan push docker kedalam Docker
- 4. Menyalin username Docker hub beserta imagenya.

Berikut adalah tahapan ketika set up di cloud:

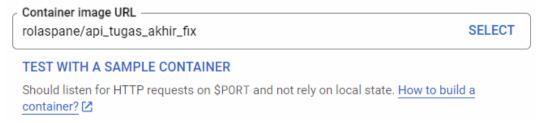
1. Enable API



2. Create Service di Cloud Run



Pada bagian ini, peneliti memilih *deploy* di *container*. Kemudian mengisi *container image* URL.



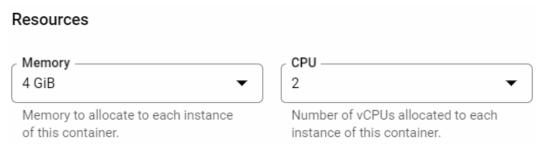
Selanjutnya, memasukkan nama *service name* dan memilih *region* apa yang ingin dipilih.



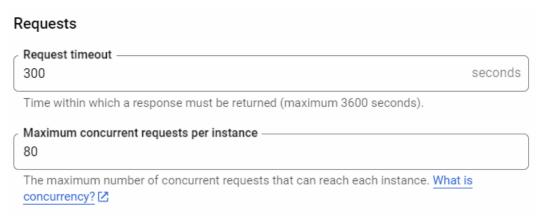
Pada bagian autentikasi, peneliti memilih 'allow unauthenticated invocations' agar dapat digunakan untuk semua orang.

Authentication *	
	Allow unauthenticated invocations Check this if you are creating a public API or website.
	Require authentication Manage authorized users with Cloud IAM.
Pada b	pagian ini, peneliti memilih CPU dan ingress secara default dan untuk service
autosc	caling peneliti memilih 1 agar dapat menghindari isu cold start.
CPU	allocation and pricing ②
	PU is only allocated during request processing ou are charged per request and only when the container instance processes a request.
_	PU is always allocated out on the container instance.
Service autoscaling ?	
Mini 1	mum number of instances
Ingre	ess control ②
Al	Iternal Ilow traffic from your project, shared VPC, and VPC service controls perimeter. Traffic from nother Cloud Run service must be routed through a VPC. Limitations apply. Learn more
A	
	llow direct access to your service from the internet pagian ini, peneliti menyesuaikan port berapa <i>container</i> akan berjalan dan
	ti menggunakan port bawaan dari Google Cloud Platform.
Cont	ainer image URL
	pane/api_tugas_akhir_fix
Con	ntainer port —
808	30
	juests will be sent to the container on this port. We recommend listening on DRT instead of this specific number.
D 11	.' 1 1 4 C'D 1 4 CDII 11 1 1

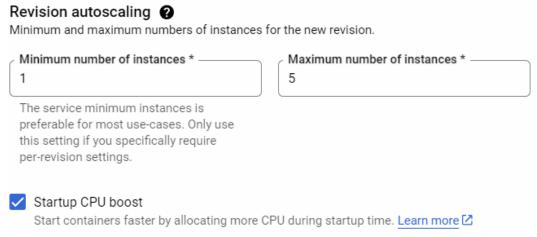
Peneliti menggunakan memory sebanyak 4 GiB dan 2 CPU sesuai kebutuhan.



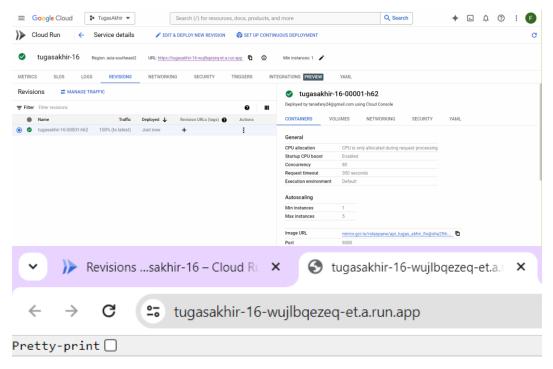
Pada bagian ini, peneliti menerapkan *request timeout* dan juga maksimum *requests* per *instance* sesuai bawaan dari Google Cloud Platform.



Pada bagian ini, peneliti mengisi 1 pada bagian minimum dan 5 pada bagian maksimum untuk memaksimalkan *autoscaling*. Kemudian klik *Create* untuk membuat *service* di Cloud Run.



Berikut merupakan tampilan ketika berhasil membuat service di Cloud Run dan berhasil men-*deploy container* di Cloud Run.



{"detail": "Method Not Allowed"}