

Návod pro implementaci a nasazení middleware aplikace s robotem NAO a OpenAI API

version 2.0 license MIT last commit november contributors 1 python 95.4%

Tento návod poskytuje detailní kroky pro nasazení middleware aplikace, která propojuje robota NAO s OpenAI API prostřednictvím platformy Google Cloud Run. Tento dokument zahrnuje všechny klíčové části procesu, od přípravy prostředí po zabezpečení komunikace mezi jednotlivými komponentami.

Obsah

- Úvod
- Příprava prostředí
 - Instalace potřebných nástrojů
 - Nastavení Google Cloud
- Nastavení Google Cloud
 - Vytvoření Docker repozitáře
 - Nastavení Dockeru pro komunikaci s Google Cloud
- Vytvoření a nasazení kontejneru
 - Vytvoření Docker image
 - Nasazení aplikace na Google Cloud Run
- Testování aplikace
 - Postup
- Připojení robota NAO k aplikaci

1.Úvod

Tento návod je součástí bakalářské práce, která se zaměřuje na integraci robota NAO s OpenAI API prostřednictvím middleware aplikace nasazené na Google Cloud Run. Cílem je ukázat, jak efektivně propojit roboty s velkými jazykovými modely pro přirozenou a plynulou interakci s uživateli.

2.Příprava prostředí

Pro nasazení aplikace na Google Cloud je třeba mít připravené následující nástroje:

2.1.Instalace potřebných nástrojů

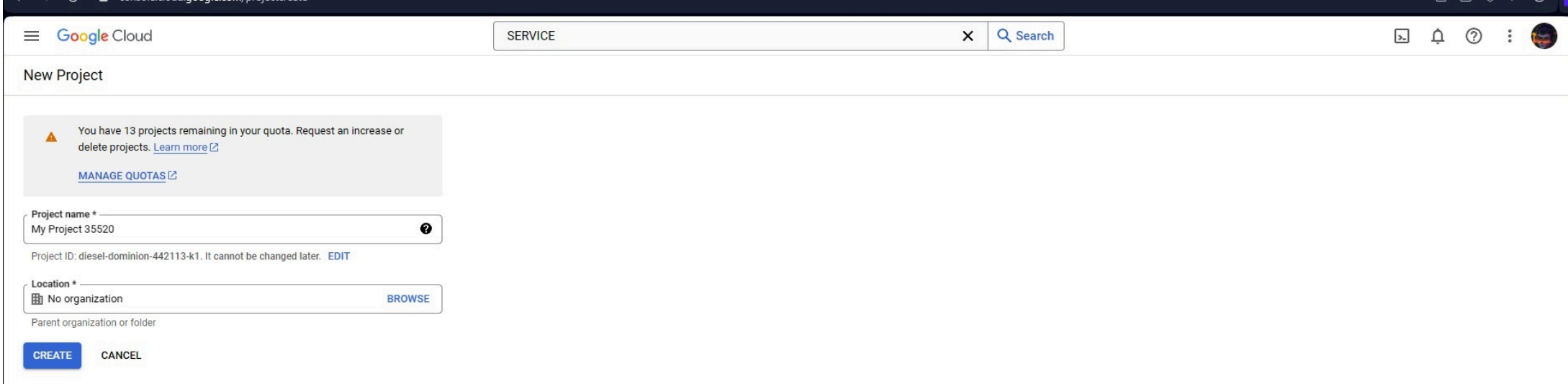
- Google Cloud SDK (CLI):** Slouží pro autentifikaci a komunikaci s Google Cloud.
 - Instalace Google Cloud SDK
- Python 3.x:** Tento jazyk je nutný pro běh aplikace.
 - Instalace Pythonu
- Docker (volitelné):** Je volitelný nástroj pro vytvoření kontejneru aplikace.
 - Instalace Dockeru
- IDE pro vývoj (volitelné):** Uspnadňuje vývoj aplikace, správu závislostí a debugging.
 - PyCharm (doporučeno pro větší projekty s Pythonem a Flaskem):
 - Instalace PyCharm
 - VS Code (lehčí a flexibilní IDE):
 - Instalace VS Code

Poslední 3 body jsou pouze v případě, že je potřeba provést revizi kódu, či aktualizaci. Jinak stačí stáhnout celý projekt, který je přiložen v přílohách u mojí bakalářské práci a pokračovat dále v tomto návodu.

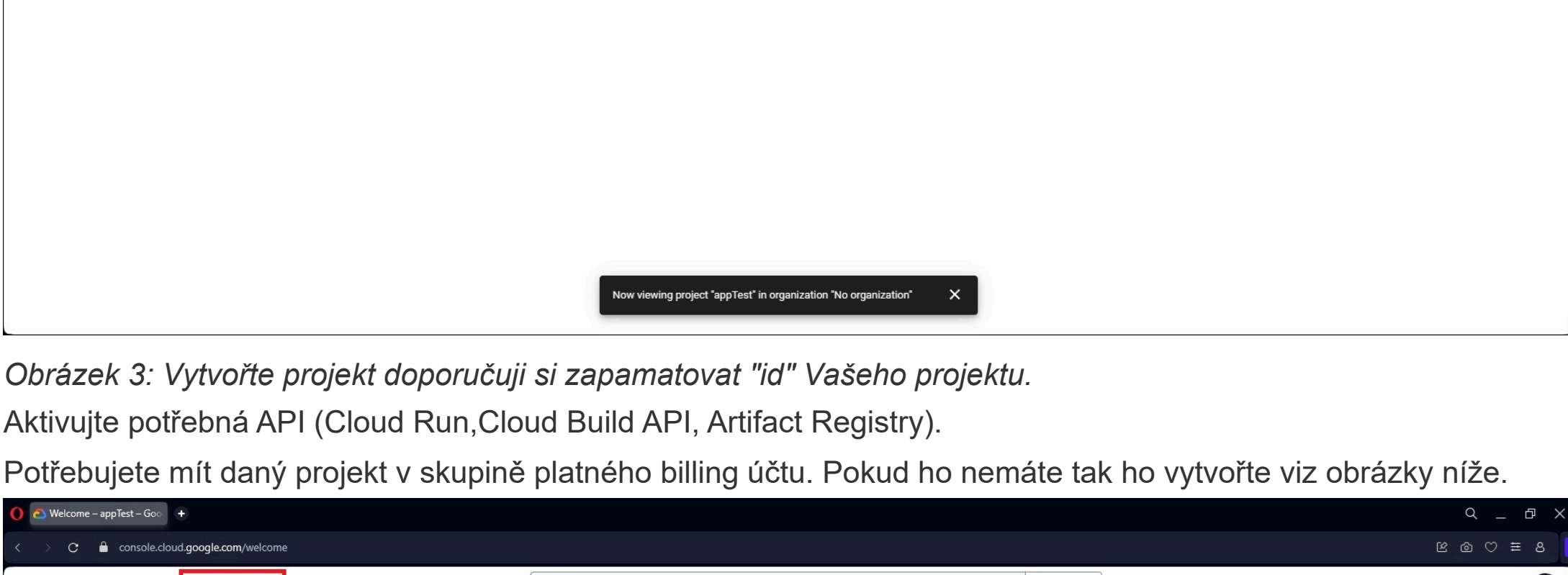
2.2.Nastavení Google Cloud

Předtím než se dostanete do Google Cloud tak musíte mít účet od Googlu a Billing účet. Pro správné fungování aplikace na Google Cloud musíte vytvořit projekt a nastavit potřebné API.

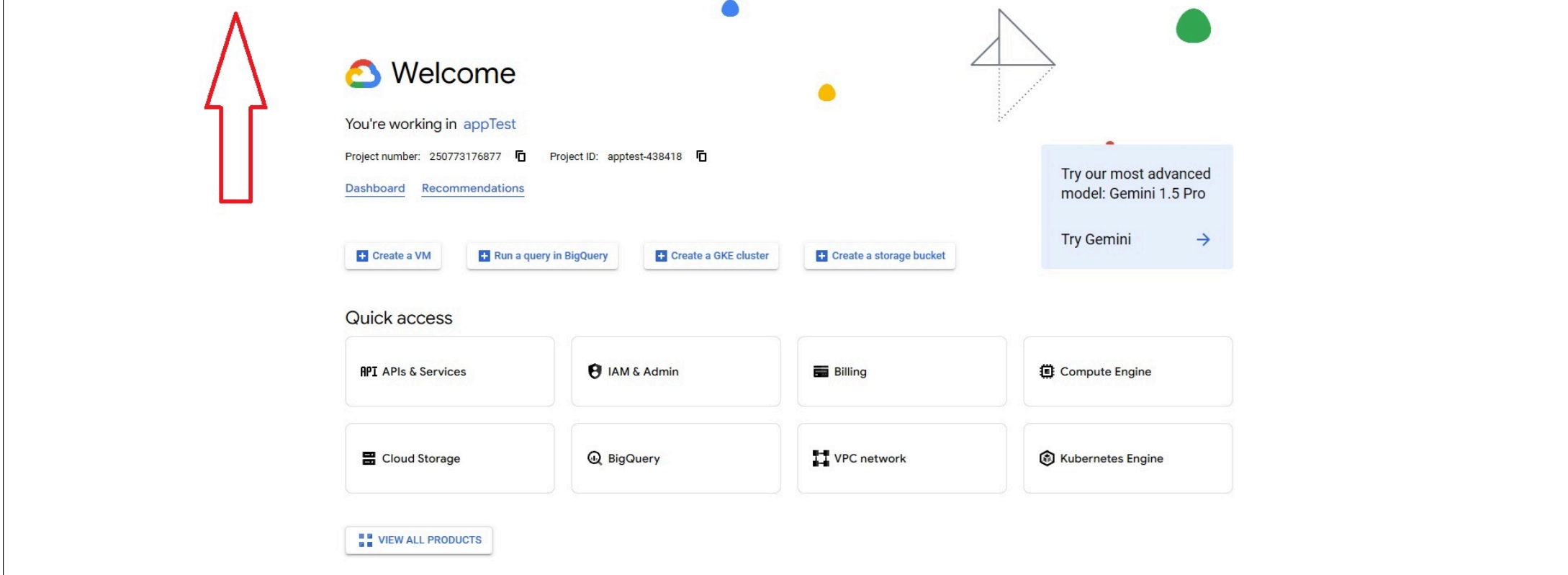
- Vytvořte projekt na [Google Cloud Console](#).



Obrázek 1: Klikněte na tlačítko vedle loga Google Cloud pro vytvoření projektu.

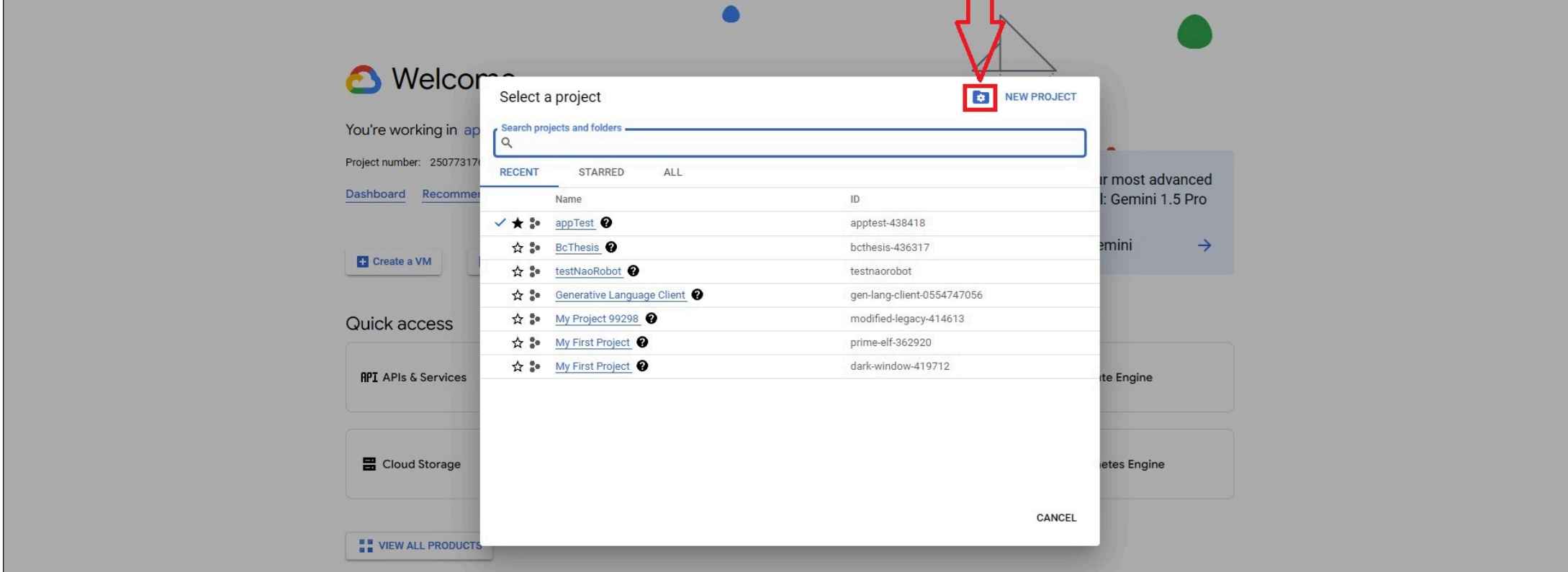


Obrázek 2: Klikněte na tlačítko create project.

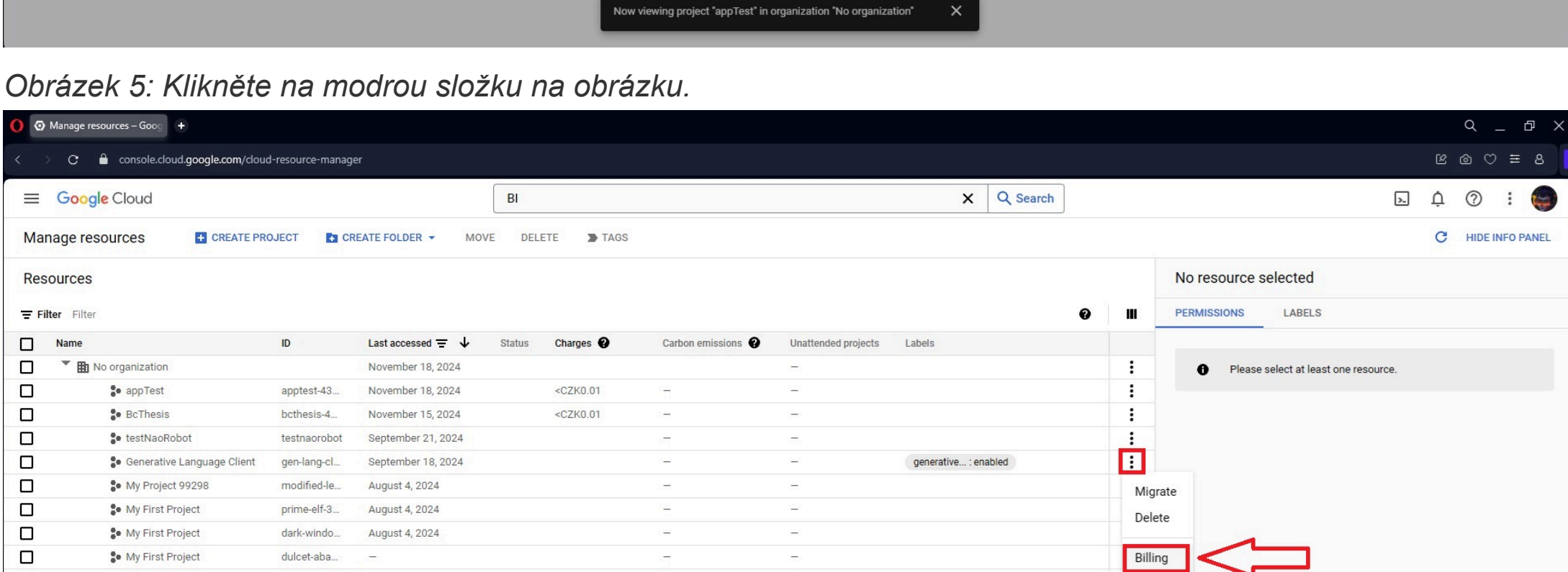


Obrázek 3: Vytvořte projekt doporučuji si zapamatovat "id" Vašeho projektu.

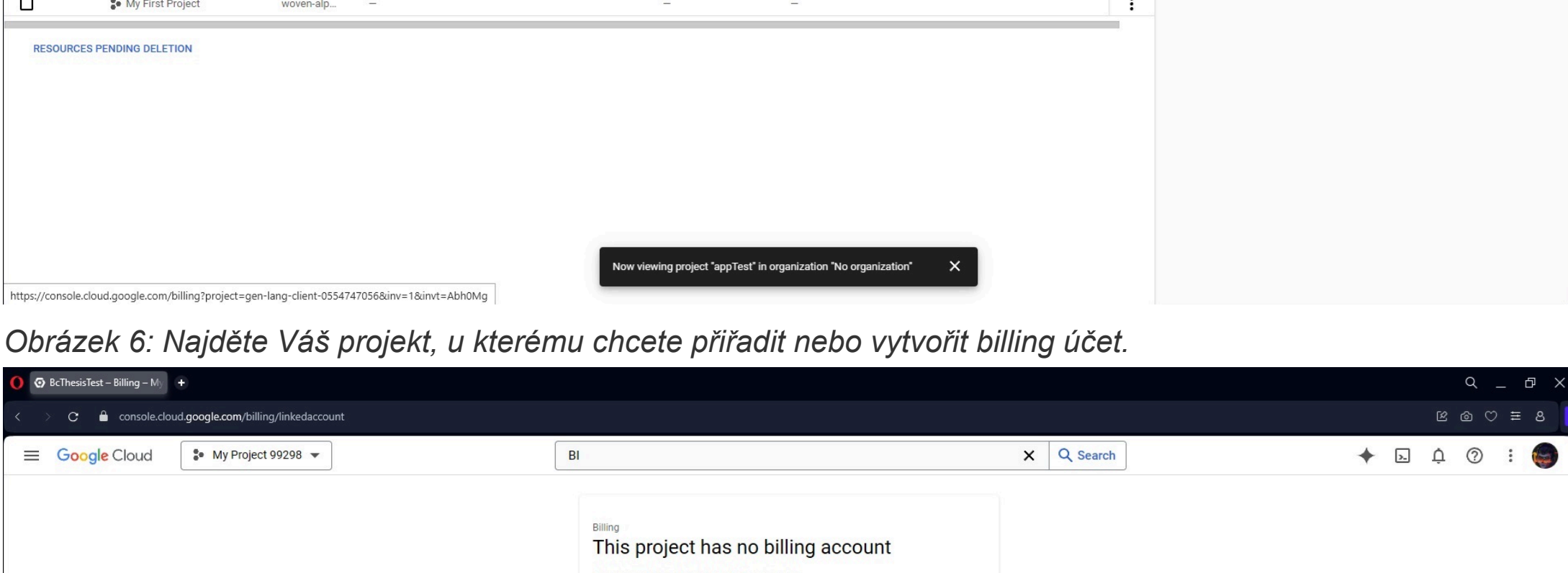
- Aktivujte potřebná API (Cloud Run, Cloud Build API, Artifact Registry).
- Pořadujete mít daný projekt v skupině platného billing účtu. Pokud ho nemáte tak ho vytvořte viz obrázky níže.



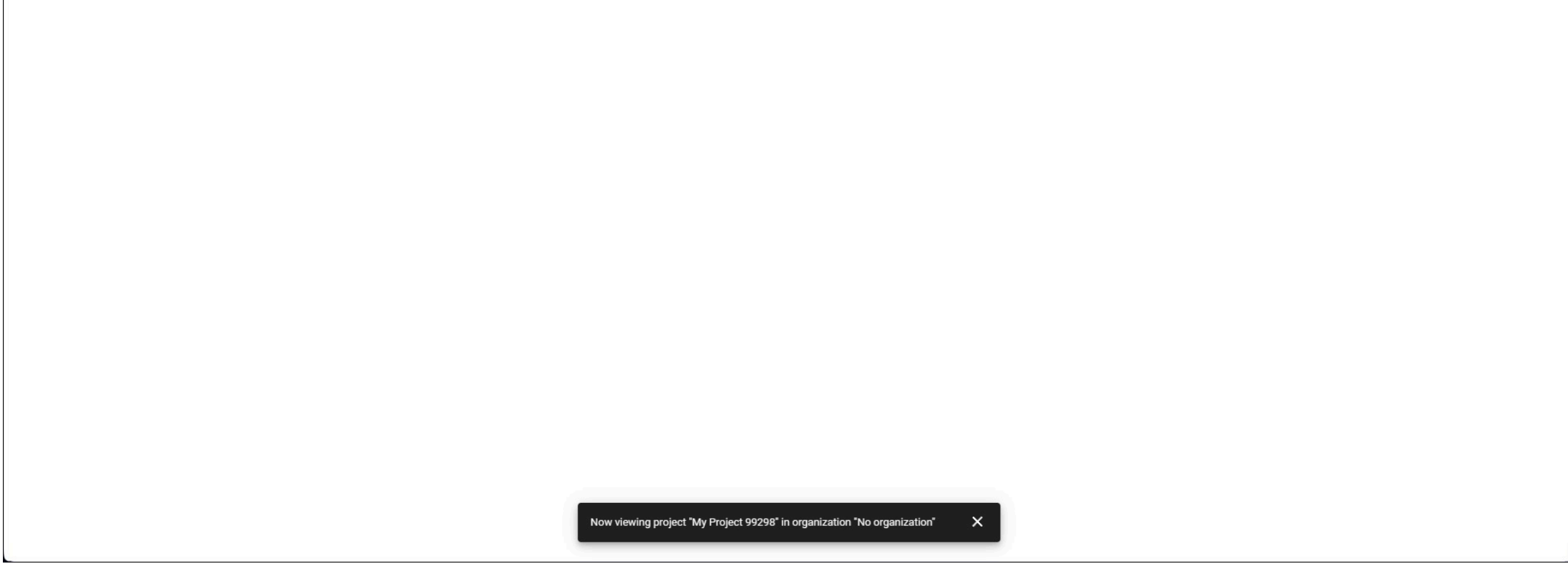
Obrázek 4: Znovu klikněte na tlačítko vedle loga Google Cloud pro vytvoření projektu.



Obrázek 5: Klikněte na modrou složku na obrázku.



Obrázek 6: Najděte Váš projekt, u kterému chcete přiřadit nebo vytvořit billing účet.



Obrázek 7: Zde máte na výběr na přiřazení billing účtu k Vašemu projektu nebo pokud ho nemáte tak klikněte na manage billing account, kde si ho můžete vytvořit.

- Pokud máte všechny předešlé kroky tak si otevřete Vaši Google Cloud SDK CLI u můžete začít s nastavení autentifikace pomocí příkazů:

```
gcloud auth login
gcloud config set project YOUR_PROJECT_ID # projekt id je viditelné na obrázku 2 nebo na hlavní stránce console
```

3.Nastavení Google Cloud

3.1.Vytvoření Docker repozitáře

Pokud ještě nemáte repozitář pro ukládání Docker image, vytvořte nový pomocí následujícího příkazu:

```
gcloud artifacts repositories create YOUR_REPO_NAME \
--repository-format=docker \
--location=europe-west4 \
--description="" \
--immutable-tags \
--async
```

3.2.Nastavení Dockeru pro komunikaci s Google Cloud

Pro konfiguraci Dockeru pro práci s Google Cloud:

```
gcloud auth configure-docker europe-west4-docker.pkg.dev
```

4.Vytvoření a nasazení kontejneru

4.1.Vytvoření Docker image

Přejděte do adresáře, kde máte svůj Dockerfile, a spusťte následující příkaz pro vytvoření Docker image:

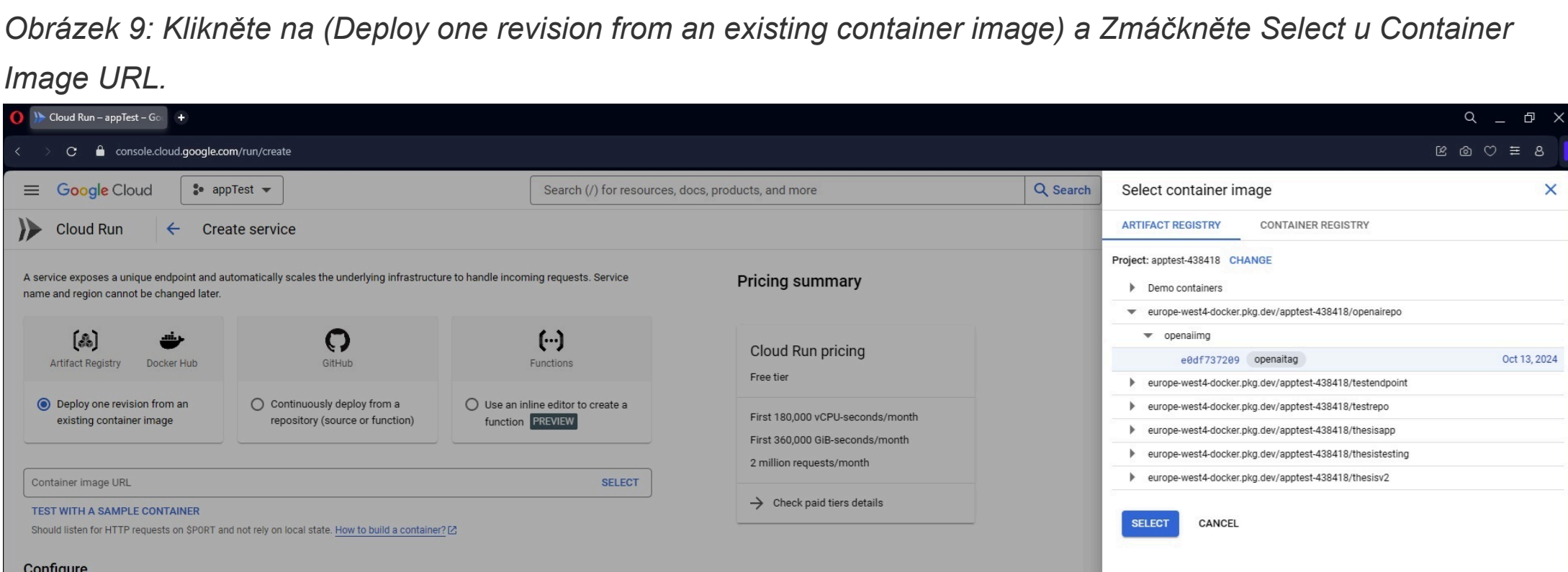
```
gcloud builds submit --tag europe-west4-docker.pkg.dev/YOUR_PROJECT_ID/YOUR_REPO_NAME/openaiimg:latest
```

Varování: Může se stát, že Vám neprojde nasazení proto je nutné jít do [IAM & Admin](#). Zde budete mít dva účty Váš osobní a dále druhý účet. U každého účtu bude tužka, která umožní přidávat a odebírat práva. Tak mu přidejte práva na 'Storage Object Viewer'.

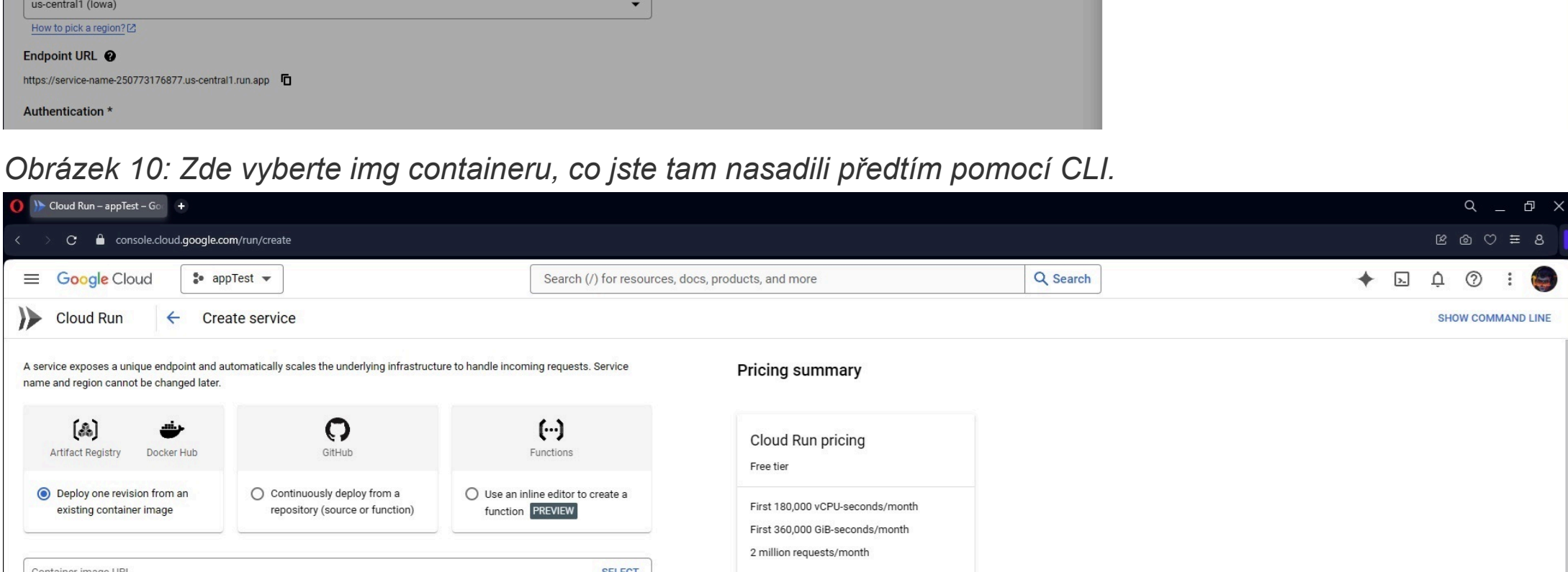
4.2.Nasazení aplikace na Google Cloud Run

Po vytvoření Docker image nasadíte aplikaci na Google Cloud Run. Jde to buď přes Google Cloud SDK CLI nebo přímo přes Google Cloud Run. V tomto případě můžete zavřít CLI a otevřít Google Cloud Run.

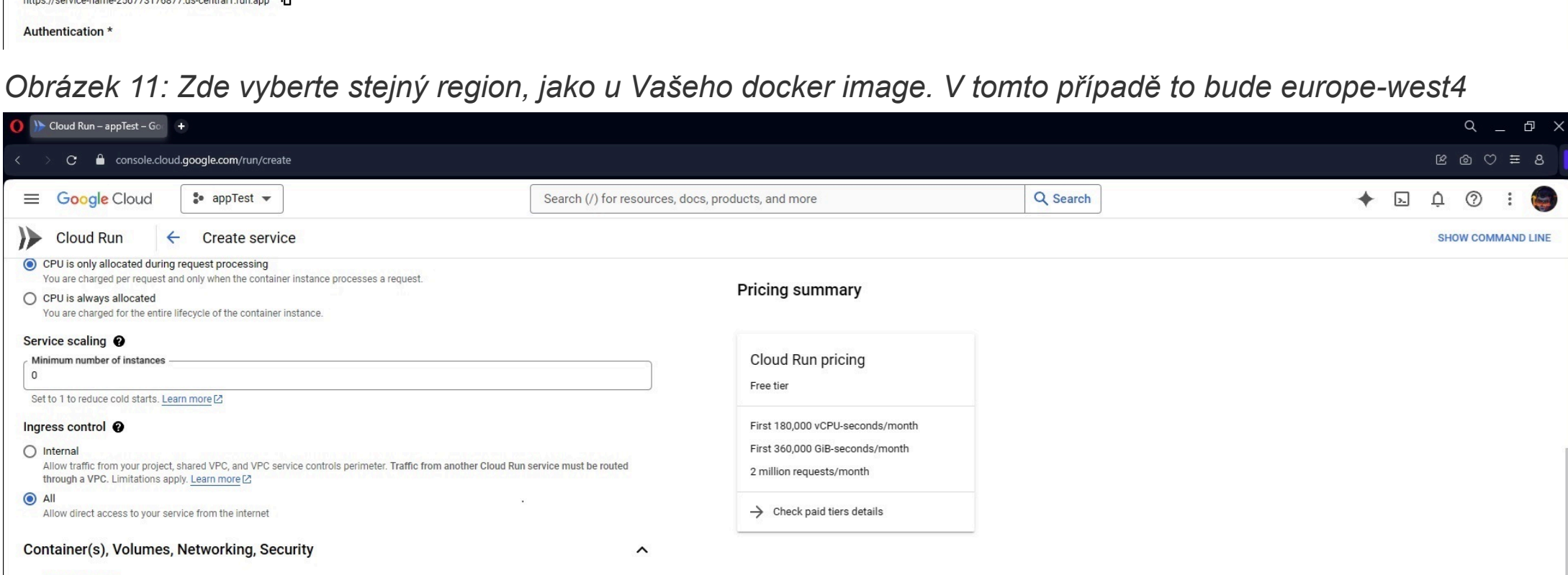
- Zadejte do vyhledávacího okna na Google Cloud název Cloud Run a klikněte na něj.



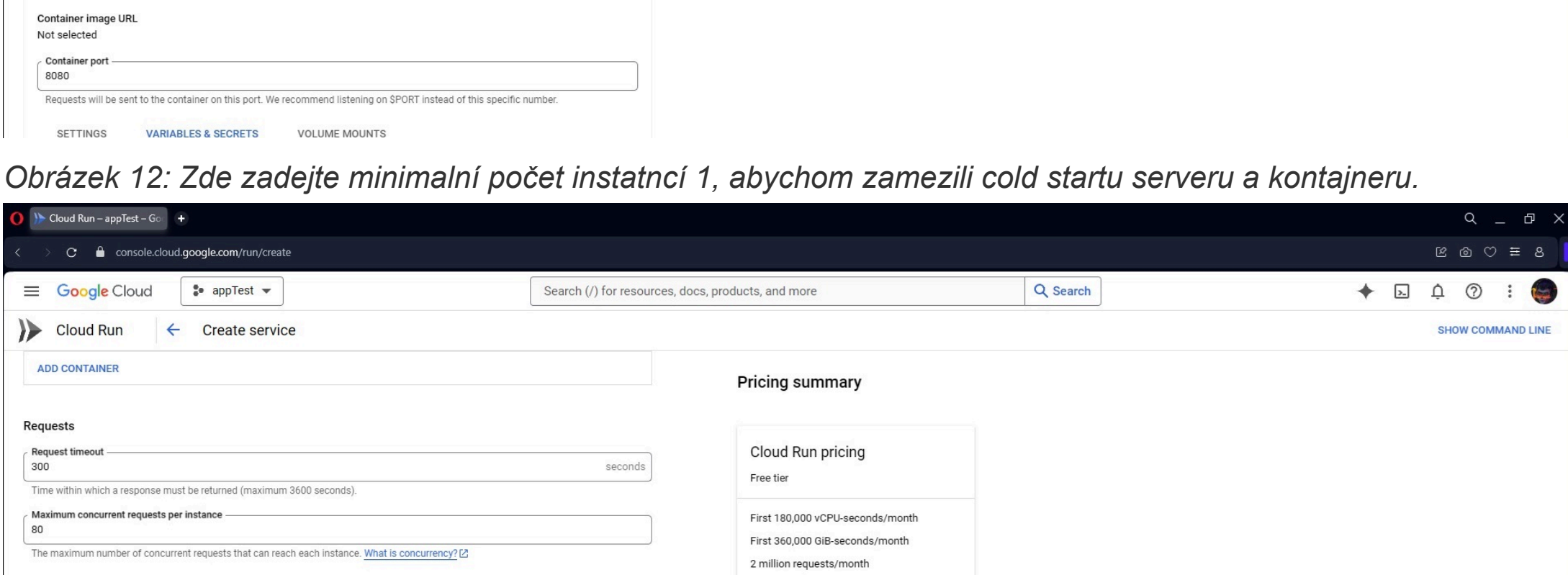
Obrázek 8: Klikněte na tlačítko Create Service.



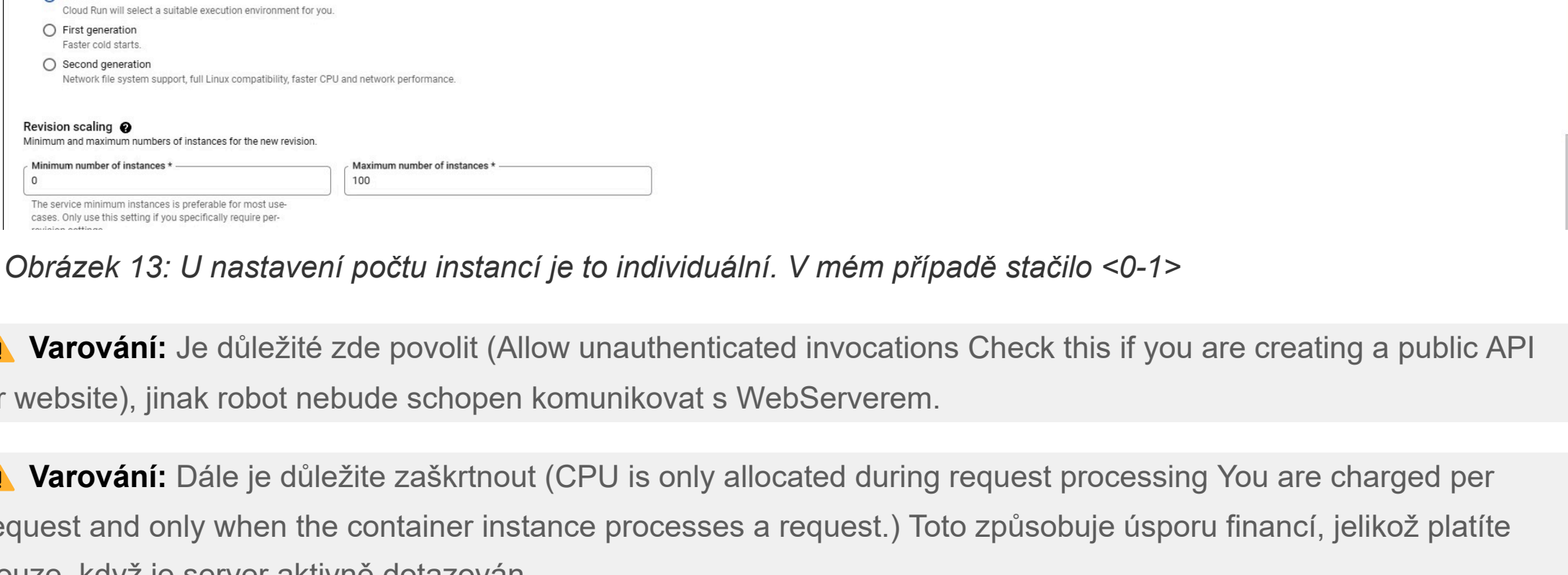
Obrázek 9: Klikněte na (Deploy one revision from an existing container image) a Zmáčkněte Select u Container Image URL.



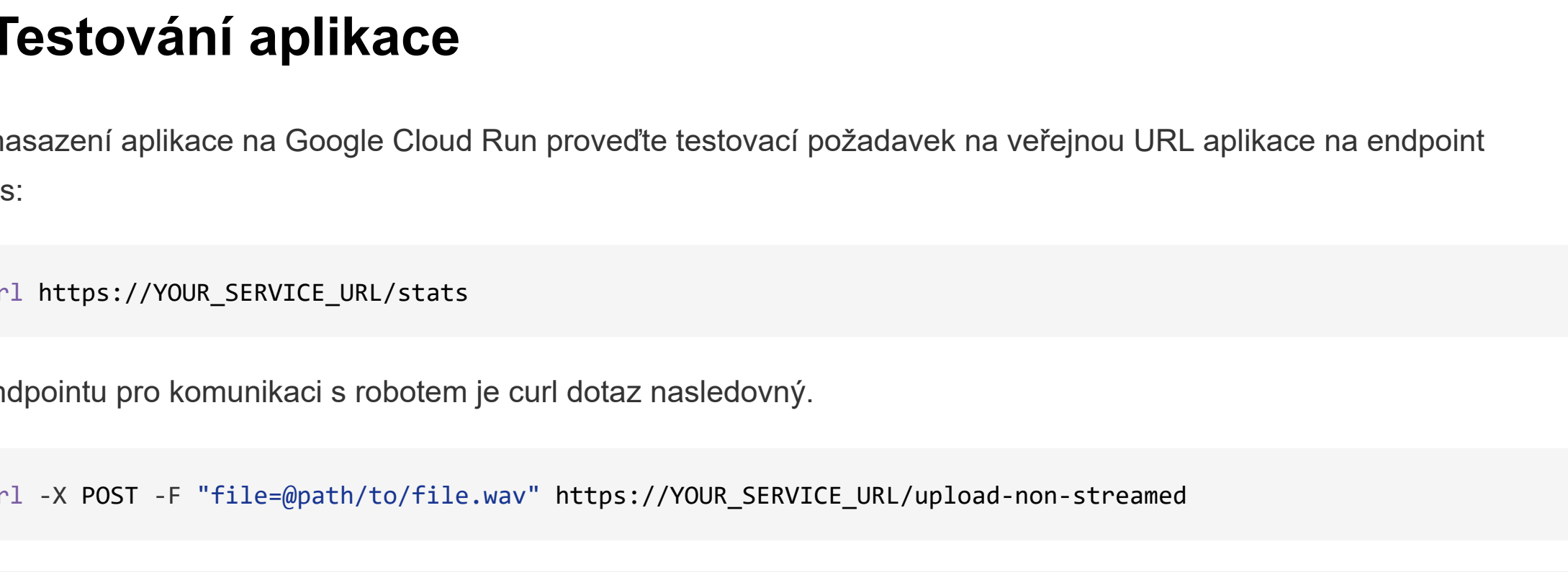
Obrázek 10: Zde vyberte img containeru, co jste tam nasadili předtím pomocí CLI.



Obrázek 11: Zde vyberte stejný region, jako u Vašeho image. V tomto případě to bude europe-west4



Obrázek 12: Zde zadejte minimální počet instancí 1, abychom zamezili cold startu serveru a kontajneru.



Obrázek 13: U nastavení počtu instancí je to individuální. V mém případě stačilo <0-1>

Varování: Je důležité zde povolit (Allow unauthenticated invocations Check this if you are creating a public API or website), jinak robot nebude schopen komunikovat s WebServerem.

Varování: Dále je důležité zaškrtnout (CPU is only allocated during request processing You are charged per request and only when the container instance processes a request.) Toto způsobuje úsporu financí, jelikož platíte pouze, když je server aktivně dotazován.

5.Testování aplikace

Po nasazení aplikace na Google Cloud Run proveďte testovací požadavek na veřejnou URL aplikace na endpoint /stats:

```
curl https://YOUR_SERVICE_URL/stats
```

U endpointu pro komunikaci s robotem je curl dotaz následovný:

```
curl -X POST -F "file=@path/to/file.wav" https://YOUR_SERVICE_URL/upload-non-streamed
```

Info: Odpověď bude v JSON formátu {"response": "Some text"}.

Pokud aplikace správně odpoví, znamená to, že nasazení bylo úspěšné.

6.Připojení robota NAO k aplikaci

Pro zajištění propojení robota NAO s middleware aplikací je nutné provést několik kroků. Pro usnadnění tohoto procesu je k dispozici připravený soubor ve formátu ZIP, který je součástí této bakalářské práce. Stačí jej stáhnout a provést úpravu jediné proměnné v modulu pro odesílání audia.

6.1.Postup:

- Stahování souboru ZIP:** Stáhněte si přiložený ZIP soubor, který obsahuje potřebné skripty a moduly pro integraci robota NAO s aplikací.
- Úprava URL:**
 - Otevřete soubor zodpovědný za odesílání audia z robota na middleware.
 - Vyhledejte řádek, který obsahuje proměnnou `self.api_url`.
 - Změňte hodnotu této proměnné z:

```
self.api_url = 'http://172.20.10.4:5000/upload'
```

na novou URL adresu serveru, který hostuje middleware aplikaci.

- Uložení a nasazení:** Uložte provedené změny a nahrajte upravený modul zpět do robota NAO.

Tímto krokem je robot NAO propojen s middleware aplikací a je připraven na interakci s OpenAI API prostřednictvím serveru.

Author

Filip Nachtmann

License

Copyright © 2024 Dsa-jpg. This project is MIT licensed.