## horizontal line



DevConnect

Django WebApp

07-08.11.2023

https://github.com/TamarNidam/Devconnect-App

Tamar Nidam

tamar.nidam1@gmail.com

# תוכן עניינים:

[**תוכן עניינים: 1**](#_l4nztzx6hpiz)

[**Overview 2**](#_au51mny0sx6)

[**Goals 2**](#_3at9u9s4e0vp)

[**הקמת סביבה: 3**](#_2izwz9bnlwg)

[תקלות: 4](#_8wfbz8gnzklr)

[1. 4](#_amct1isds6m0)

[2. 5](#_shwsoe9wy9lr)

[3. 5](#_xh217dyl601a)

[4. 5](#_7c4l9d4l7b8u)

[5. 6](#_lgt8ejng576m)

[**DEPLOYMENT: 6**](#_6cmb6gfrx74)

[תקלות: 8](#_6cyukmhs9e47)

[2. 9](#_lko081vl242y)

[3. 9](#_j33pv8yo75dg)

[4. 9](#_7zhknlbm43ik)

[**:CI/CD 11**](#_kpxin51sos0o)

[תקלות: 12](#_9d9mnv9orzju)

[1 12](#_h9uz82czha6p)

[2. 12](#_fg006wnrx9ul)

# Overview

In a rapidly evolving digital landscape, DevOps has become the driving force behind innovation and efficiency in the world of software development and IT operations.

However, as the DevOps field expands, professionals are finding it increasingly challenging to connect with like-minded individuals, share best practices, and keep up with the latest industry trends. This gap in communication and collaboration has sparked the creation of DevConnect, a revolutionary social web app designed to bring the global DevOps community together.

With a clear vision in mind, the founders assembled a dedicated team of DevOps engineers, software developers, and designers to bring DevConnect to life.

The founders chose you as a DevOps engineer for their biggest product The DevConnect web app.

You are an essential part of this ambitious project, entrusted with the responsibility to ensure the smooth deployment, scaling, and maintenance of the platform. Your expertise in automation, continuous integration, and continuous delivery is critical for the success of

DevConnect.

Go &quot;DevOps&quot; your way through this project, following best practices and GCP technologies to create an outstanding platform that unites the global DevOps community.

# Goals

1. Dockerization
2. Deployment
3. CI/CD

The app: <http://localhost:8000/>

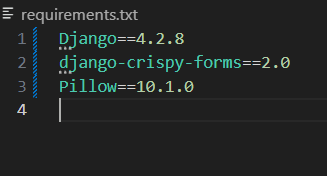
Clusters:h[ttps://console.cloud.google.com/kubernetes/list/overview?project=devconnect-final-project](https://console.cloud.google.com/kubernetes/list/overview?project=devconnect-final-project)

Instance: https://console.cloud.google.com/compute/instances?project=devconnect-final-project&pageState=(%22instances%22:(%22p%22:2))

# הקמת סביבה:

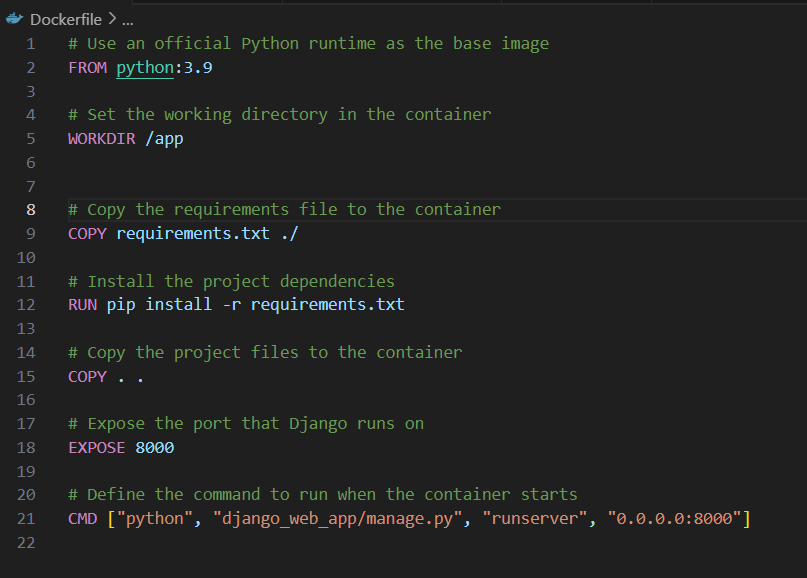
Django היא מסגרת אינטרנט ברמה גבוהה של Python המאפשרת למפתחים לבנות במהירות יישומי אינטרנט. הוא עוקב אחר הדפוס הארכיטקטוני של Model-View-Controller (MVC) ומתמקד בשימוש חוזר ובפיתוח מהיר. Django מספקת קבוצה של כלים וספריות המפשטות משימות פיתוח אינטרנט נפוצות, כגון טיפול בניתוב כתובות אתרים, ניהול מסדי נתונים, טיפול בטפסים ואימות משתמשים.

**הורדת קובץ הZIP. יצירת docker file מתאים, וקובץ requirements.txt עבור הפרויקט DJANGO.**



הקובץ טקסט נוצר כדי להבטיח שכל התלות הדרושות מותקנות בעת הגדרת פרויקט Django.

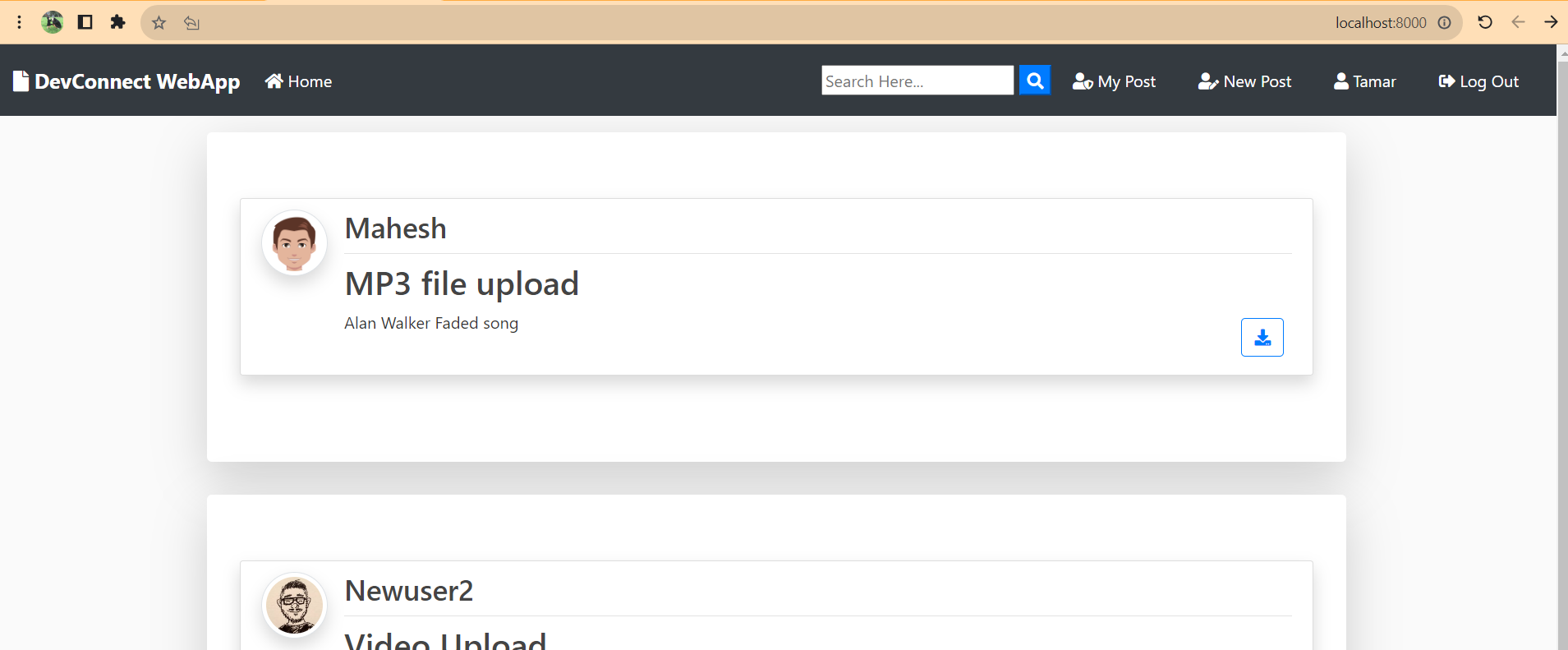
וכן בדוקרפייל מריצים אותו כדי להבטיח שכל הקוד והתלויות הנדרשות יתקנו בתוך הקונטיינר.



ניתן להחליף את הפקודה CMD בENTRYPOINT:  
CMD משמש כדי לספק ארגומנטים של ברירת מחדל שניתן לעקוף, בעוד ENTRYPOINT מגדיר את הפקודה הראשית שתבוצע. CMD משמש בדרך כלל להגדרת התנהגות ברירת מחדל או מתן ארגומנטים, בעוד ENTRYPOINT שימושי כאשר ברצונך להגדיר את הפקודה הראשית עבור המיכל.

**בנייה והרצה בDOCKER.**

**יצירת קבצי INIT וDELETE לאוטומציה.  
בדיקת פעילות תקינה של האפליקציה.**

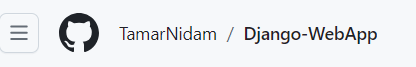
****

**הפעלת קונטיינר עם נפח מותקן.**

הפעלת קונטיינר עם אמצעי אחסון מותקן כרוכה בחיבור ספרייה ספציפית במחשב המארח לספרייה מתאימה בתוך הקונטיינר. זה מאפשר למכולה לקרוא ולכתוב נתונים למערכת הקבצים של המארח, ולספק עמידות נתונים גם אם המיכל נעצר או נמחק.

הרכבה של אמצעי אחסון מועיל מכיוון שהיא מבטיחה ששינויים שנעשו בתוך המכולה (למשל, עדכוני נתונים, שינויי תצורה) מאוחסנים חיצונית, בנפרד מהאחסון החולפ של המכולה. גישה זו מאפשרת למיכל לשמור על מצבו במהלך הפעלה מחדש או כאשר מכולה חדשה מסתובבת. זה גם מאפשר גישה קלה וגיבוי של נתונים מחוץ לקונטיינר. על ידי מיפוי נפח, אתה יכול לשמור על עקביות ושלמות הנתונים, להבטיח חוויה חלקה למשתמשים באינטראקציה עם האפליקציה.

לכן כשניכנס לאפליקציה לא תדרש ממנו הרשמה נוספת..

אחרי בניית התמונה(לפני ההרצה) נריץ את הפקודה **docker volume create db**

**העלאת הפרויקט לGITHUB**

## תקלות:

## 1.

**פתרוון:**

שינוי מיקום הקובץ בדוקרפייל ל: django\_web\_app/manage.py

## 2.

הקונטיינר רץ כרגיל אבל לא מוצג כראוי בדפדפן  
**פתרון:**

בעיה בקובץ BASE.HTML:

שינוי השורה ל **{% load static %}**

בגרסאות קודמות של Django (לפני 1.9), אפליקציית הקבצים הסטטיים כונתה staticfiles. לכן, ספריית תגי התבניות שטענה את הקבצים הסטטיים הייתה {% load staticfiles %}.

החל מ-Django 1.9, השם של אפליקציית הקבצים הסטטיים שונה לstatic. כתוצאה מכך, ספריית תגיות התבניות לטעינת קבצים סטטיים שונתה גם היא ל{% load static %} כדי לשקף את מוסכמות השמות החדשה הזו.

## 3.

הכניסה להרשמה והתחברות עם תקלותת  
**פתרון:**

הסרת משתני הסביבה **ENV PYTHONUNBUFFERED=1** מהדוקרפייל

והרצה שוב ושובבב

הוספת השורה:



## 4.

python: can't open file '/app/django\_web\_app/manage.py': [Errno 2] No such file or directory

**פתרון:**

סידור ניתוב התקייה בה הפרויקט יורץ לתיקייה בה יש את קבצי הפרוייקט

docker run -d -p 8000:8000 -v db:/app django-webapp

## 5.

remote: Permission to smahesh29/Django-WebApp.git denied to TamarNidam.

fatal: unable to access 'https://github.com/smahesh29/Django-WebApp.git/': The requested URL returned error: 403

**פתרון:**  
עריכה שנית של הרפוזטורי:

>git remote set-url origin https://github.com/TamarNidam/Django-WebApp

# DEPLOYMENT:

**יצירת ארטיפקט רג'יסטרי.**

****

**יצירת קלאסטר.**

נגדיר TAINT.

טיינטס הם מאפיינים של Nodes ב-Kubernetes, המאפשרים למנהלי מערכת לשלוט בכך שרק פודס מסוימים יוכלו להקצות לעצמם את הצמתים הללו.

פה- פוד עם התגית webapp=mywebapp יוכל להקצות לעצמו את הצומת, אך פוד עם התגית webapp=otherwebapp לא יוכל.

טיינטס יכולים לשמש למספר מטרות, כגון:

למנוע מפודס מסוימים להקצות לעצמם צמתים עם מאפיינים מסוימים. לדוגמה, ניתן להשתמש בטיינט כדי למנוע מפודים עם דרישה גבוהה למשאבים להקצות לעצמם צמתים עם משאבים מוגבלים.

לאלץ פודים מסוימים להקצות לעצמם צמתים עם מאפיינים מסוימים. לדוגמה, ניתן להשתמש בטיינט כדי לאלץ פודים עם דרישות אבטחה מסוימות להקצות לעצמם צמתים עם סט מסוים של עדכונים.



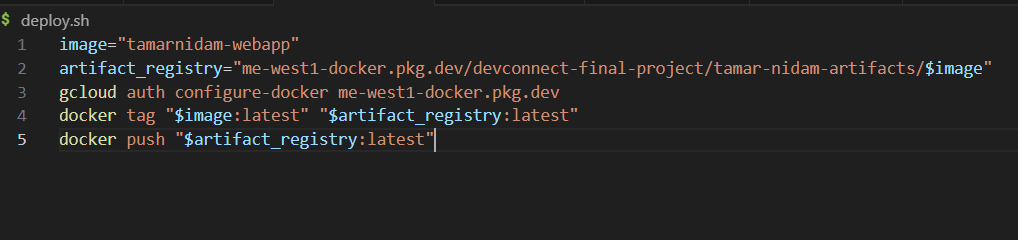
**חיבור בין הדוקר לארטיפקט רג'יסטרי.**

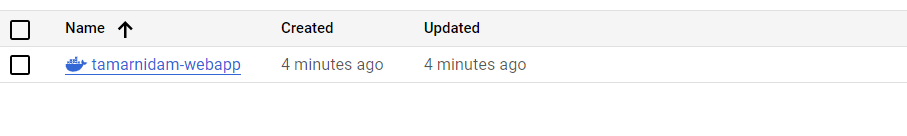
https://www.googlecloudcommunity.com/gc/Developer-Tools/Permission-quot-artifactregistry-repositories-uploadArtifacts/m-p/543870

gcloud config unset auth/impersonate\_service\_account

gcloud auth login

gcloud config set project devconnect-final-project

**יצירת קובץ DEPLOY**

**kubectl get images**

**יצירת NAMESPACE production ע"י יצירת deployment -> workloads**

**replicas=1**

*https://kubernetes.io/docs/concepts/workloads/controllers/deployment*/

הnamespace נוצר בקוברנטיס קלאסטר. הוא מייצג קבוצה של משאבי קוברנטיס כמו Pods, Deployments, Services וכו

הוא מאפשר ארגון המשאבים בצורה לוגית ולארגן קבוצות עם הגדרות משותפות.

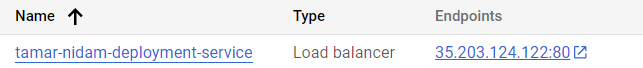
**Deployment.yaml**

****

**יצירת SERVICE המציין את הסוג כ-LoadBalancer וממפה אותו לפריסה.**

על ידי שימוש בשירות Load Balancer, אפשר להבטיח שהאפליקציה שלך נגישה מכל מקום באינטרנט. זה עוזר לנהל את התעבורה הנכנסת, מפיץ אותה על פני מספר תרמילים עורפיים, ומספק דרך אמינה וניתנת להרחבה לגשת לאפליקציה דרך דפדפן אינטרנט.

**service.yaml**



**קבלת כתובת הIP**

loadBalancer:

ingress:

- ip: 35.203.124.122

## תקלות:

1.

לאחר יצירת הקלאסטר לא ניתן לערוך אותו ולהוסיף לו NODE

התקלה נוצרה עקב חוסר בכתובות IP

**פתרון:**  
יצרו VPC נוסף (DEFUALT2) וניצור לשם את הקלאסטר



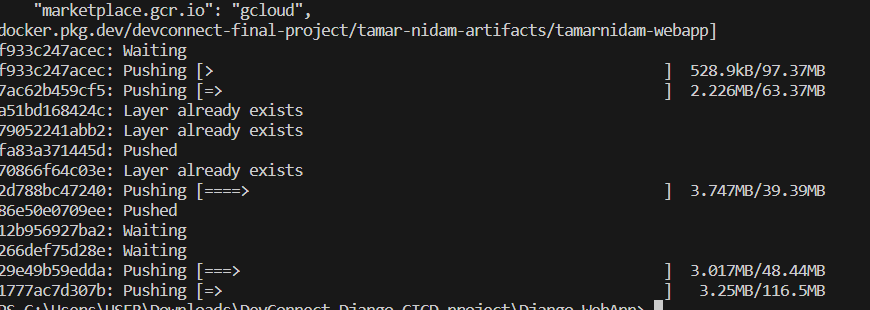
## 

## 2.

  
  
בדיקת ZONES זמינים:  
curl -X GET \

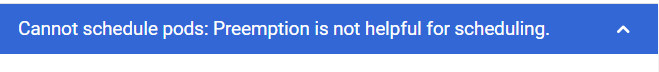
'<https://compute.googleapis.com/v1/zones?filter=region:us-central1>'

## 3.



בעיות רשת…

## 4.

**פתרון:**  
  
הוספת:

tolerations:

- key: "webapp"

operator: "Equal"

value: "mywebapp"

effect: "NoExecute"

הבלוק tolerations מאפשר לך לציין תנאים מסוימים עבור הפודים שנוצרו על ידי הפריסה. זה עוזר לקבוע אילו צמתים באשכול יכולים להפעיל את התרמילים האלה. במקרה זה, הבלוק tolerations כולל תנאי יחיד עם המאפיינים הבאים:

- key: "webapp": תנאי זה תואם לתכונה ספציפית הנקראת "webapp".

- operator: "Equal": זה מציין שערך התכונה חייב להיות התאמה מדויקת.

- value: "mywebapp": זה מגדיר את הערך הצפוי של התכונה ל-"mywebapp".

- effect: "NoExecute": זה מציין את הפעולה שיש לנקוט אם התרמיל אינו עומד בדרישות הסבילות. במקרה זה, האפקט מוגדר ל-"NoExecute", כלומר הפוד לא יתזמן בצמתים אלא אם כן יש להם את התכונה שצוינה עם הערך "mywebapp".

על ידי הוספת הסבילות הללו לקובץ ה-Deployment, אתה מאפשר לתזמן את הפודים רק בצמתים בעלי תכונה ספציפית בשם "webapp" עם הערך "mywebapp". אם לצומת אין תכונה זו, הפוד לא יתוזמן באותו צומת. זה עוזר להבטיח שהתרמילים נפרסים על צמתים מתאימים בהתבסס על הסבילות שצוינו.

# :CI/CD

**יצירת Compute engine instance**

tamar-nidam-jenkins, service account : DevOps-sa, e2-medium, me-west1(Tel-Aviv).  
Automation - install docker engine.:

#!/bin/bash

apt-get update

apt-get install -y docker.io

## 

**יצירת jenkins\_lab מקומי**

mkdir jenkins\_lab

יצירת Free-style projec עם טריגר Poll SCM

docker build -t tamar-nidam-jenkins

**דוקר TAG+PUSH לרגיסטר**  
**יצירת דוקר קומפוס**

version: "3.9"

services:

jenkins:

image: <your-artifact-registry>/<your-jenkins-image-name>

volumes:

- ./jenkins\_lab:/var/jenkins

ports:

- "8080:8080"

networks:

- jenkins\_net

networks:

jenkins\_net:

## תקלות:

## 1

Create VM instance "tamar-nidam-jenkins" and its boot disk "tamar-nidam-jenkins"in 1 minute

DevConnect-project

The CPUS-per-project-region quota maximum in region me-west1 has been exceeded. Current limit: 24.0. Metric: compute.googleapis.com/cpus.

**פתרון:**יש חריגה מהגבול המותר של כמות המעבדים המרבית בפרויקט שבאזור me-west1. הגבול הנוכחי הוא 24.0 מעבדים. יצירת ה-VM חורגת מגבול זה.ולכן ניצור אינסטנס עם בחירת ריג'ן אחר בו אפשר ליצור…

## 2.

חוסר בהרשאות מתאימות לאימות עם Registry Artifact

**פתרון:**gcloud auth configure-docker tamar-nidam-artifact