הקמת CI/CD תחת שירותי הענן של AWS

# מבוא

Continuous delivery - זו שיטת פיתוח תוכנה שבא תהליך העלאת הגרסאות מתבצע באופן אוטומטי, כל שינוי קוד נבנה, נבדק ונפרס לסביבת הייצור בצורה אוטומטית לחלוטין, לפני העלאה לסביבת הייצור ניתנת הוראה ע"י בעל תפקיד "לדחוף" את הגרסה החדשה והגרסה יוצאת באופן אוטומטי ללקוחות החברה. בעצם בשיטה זו ניתן לשחרר כל שינוי קוד קטן באופן מידי לסביבת הייצור.

Continuous integration – זו שיטת פיתוח שבה חברי הצוות משתמשים במערכת ניהול גרסאות קוד (לדוגמא GIT) , ובצורה שוטפת מבצעים תהליך של אינטגרציה של הקוד שלהם לתוך הפרויקט באופן אוטומטי במטרה ל"תפוס" שגיאות מוקדם ככול האפשר. בשיטה זו הפוקוס הוא בניה ובדיקה של הקוד בניגוד ל CD ששם הדגש הוא תהליך פריסת הגרסאות לסביבת ייצור.

# מונחים

GIT – היא מערכת לניהול גרסאות קוד

GitHub – הינו שירות המאפשר לשמור ולנהל גרסאות של הקוד תוך שימוש בפרוטקול GIT

Artifactory – רכיב שמטרתו להחזיק תוצאי בניה (גרסאות תוכנה מהודרים), בשפת Java , קבצי WAR.

S3 – שירות אחסון אובייקטים, נשתמש ברכיב הזה כ Artifactory .

Code build – מטרתו לייצר תהליך של הידור התוכנה, הרצת בדיקות שונות ודחיפה של תוצרי הבניה ל artifactory

Code Deploy - רכיב שמטרתו לקחת את תוצרי הבנייה מ Artiifactory לפרוס אותם בצורה מסודרת לשרתים

Code Pipeline - מאפשר לחבר בין ה Code Build ל-Code Deploy

EC2 – Elastic compute - מכונה וירטואלית ע"ב התשתיות ענן של AWS.

# 

# ארכיטקטורה כללי



# תהליך בנייה

לרוב תהליך הבניה של הקוד יורכב מהחלקים הבאים:

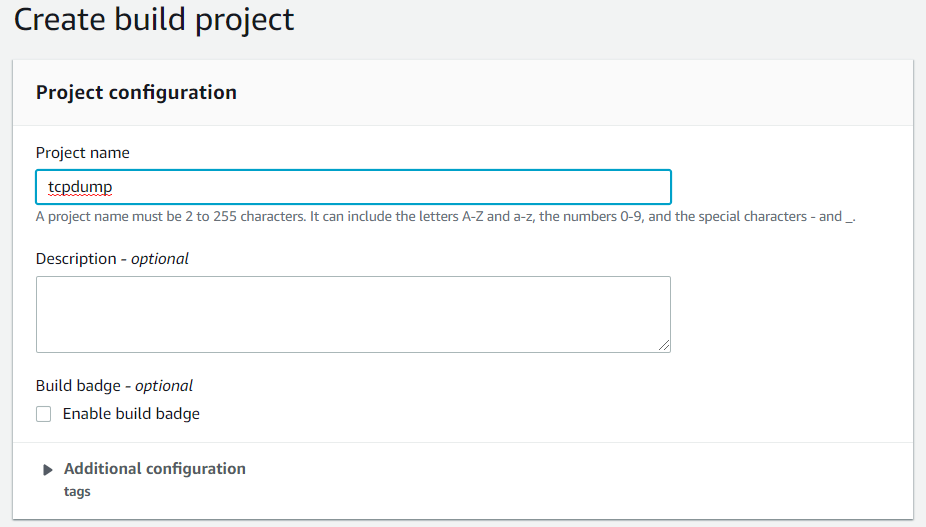


# מימוש התהליך ב AWS

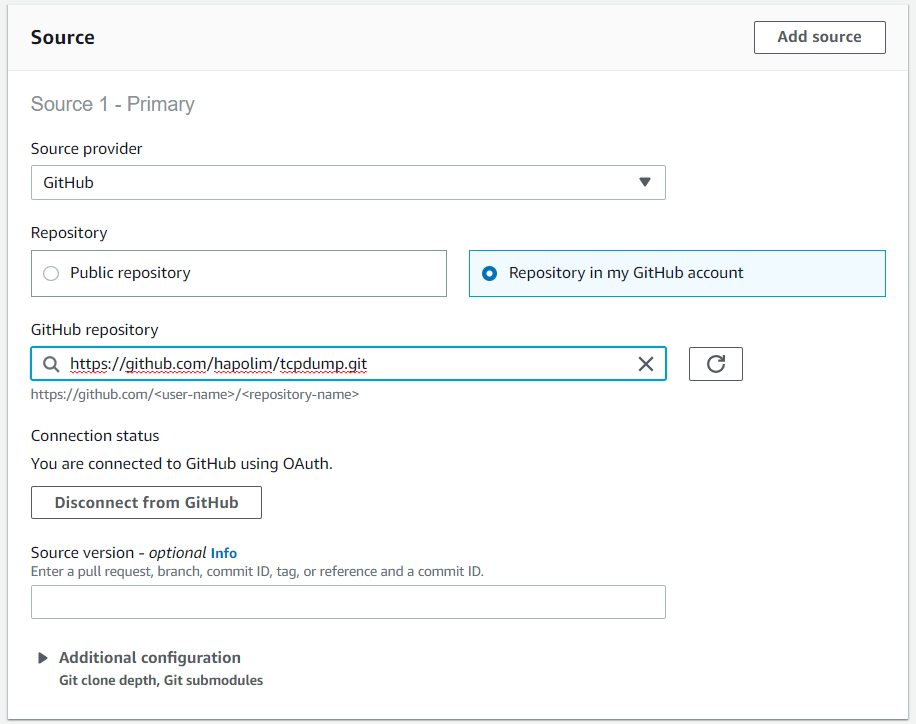
על מנת להתחיל את תהליך, אנו משתמשים בכלי CICD של אמזון שנקרא CodePipeline ,

הכלי מחולק לשלושה חלקים:

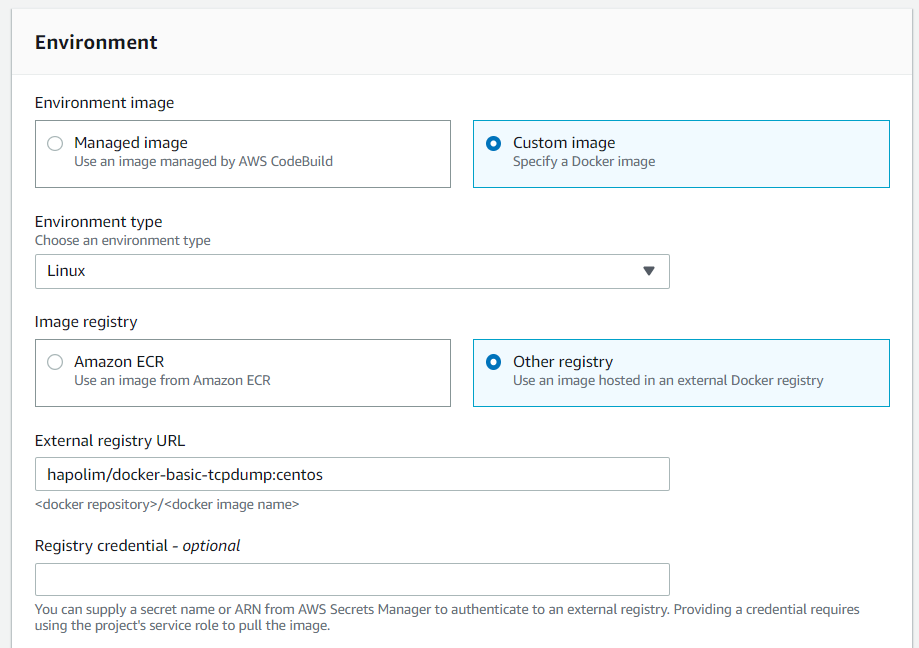
1. CodeBuild - אחראי על בניית הפרויקט
2. CodeDeploy - אחרי על פריסת הגרסאות על השרתים
3. Pipeline – מחבר את Github עם הCodeBuild ו CodeDeploy .
4. הגדרת תהליך הבנייה ב CodeBuild
   1. הגדרת שם הפרויקט



* 1. הגדרת מקור (Git repo )



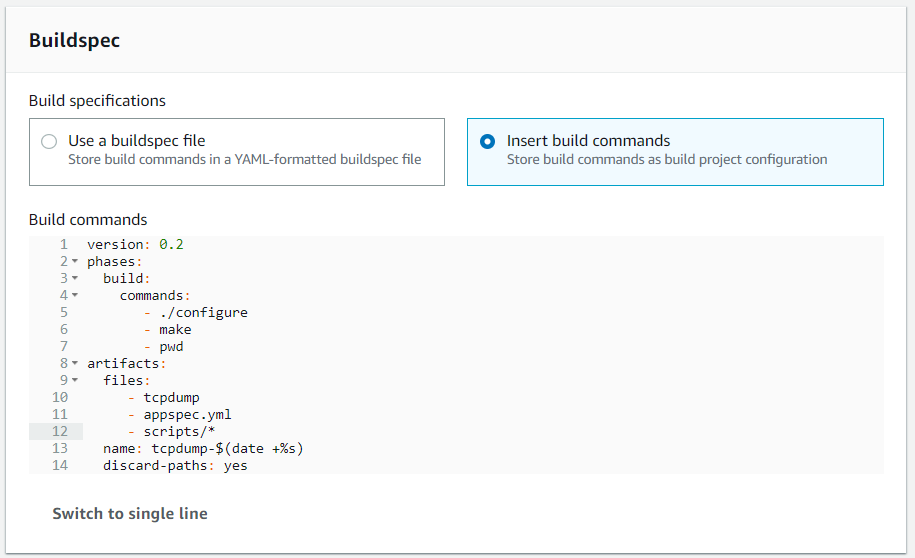
* 1. הגדרת סביבת בנייה



הערה: על מנת לקצר את תהליך הבנייה, הכנתי מראש Docker image שמכיל את כל דרישות קדם בשביל לקמפל את tcpdump .

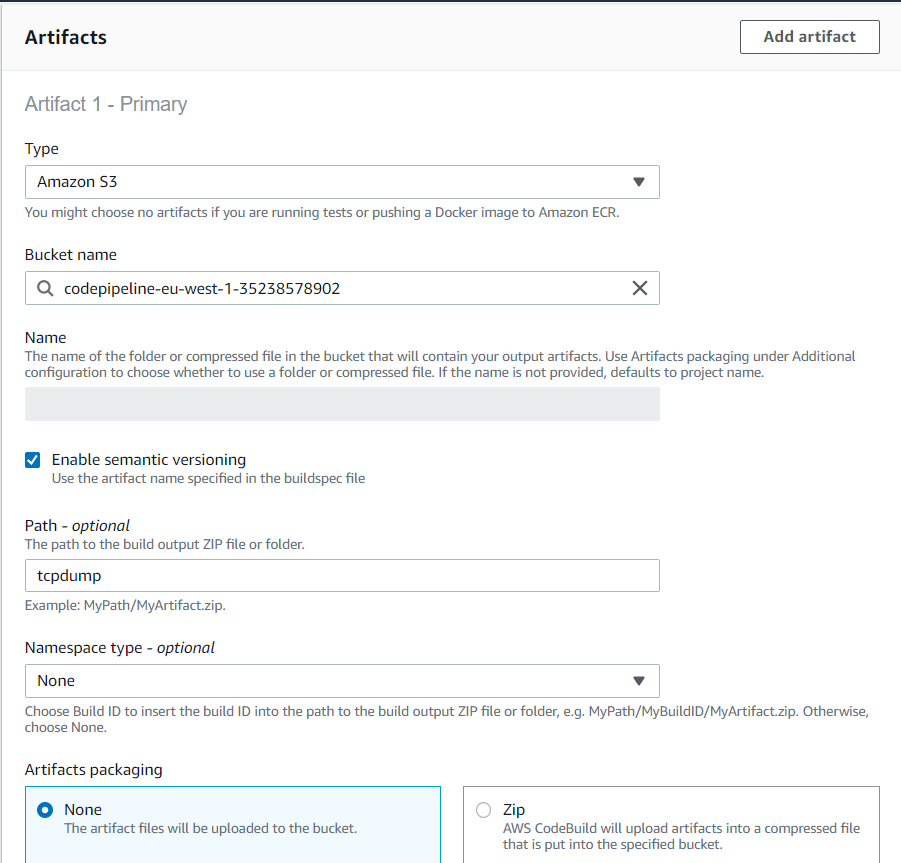
* 1. הגדרת Buildspec

בהגדרה של buildspec בעצם נגדיר איך יראה תהליך הבניה ומה לבצע עם תוצרי הבניה.

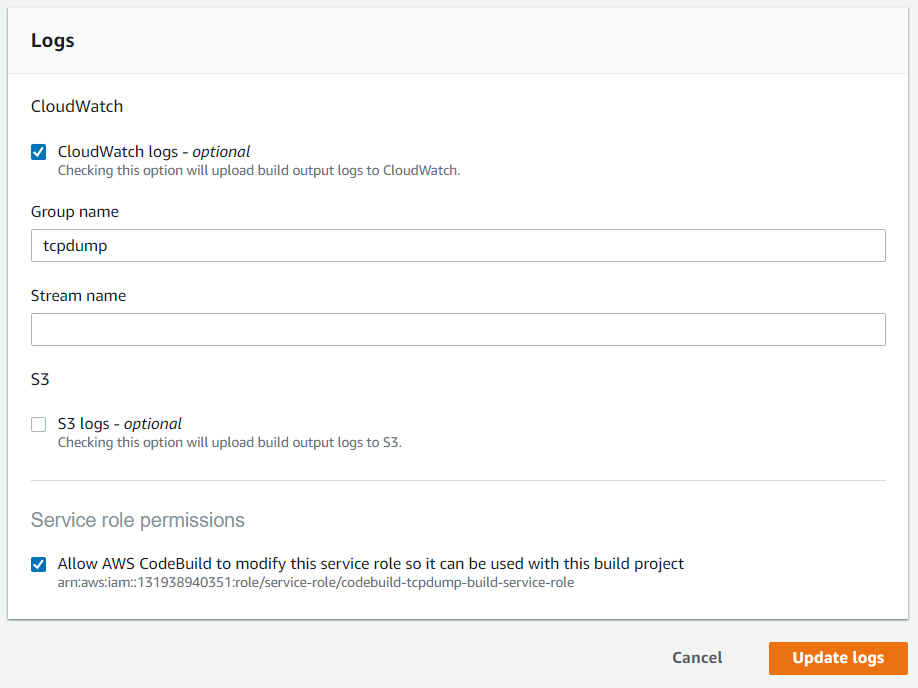


הערה: לרוב את ה buildspec נדחוף כחלק מGIT REPO של הפרויקט.

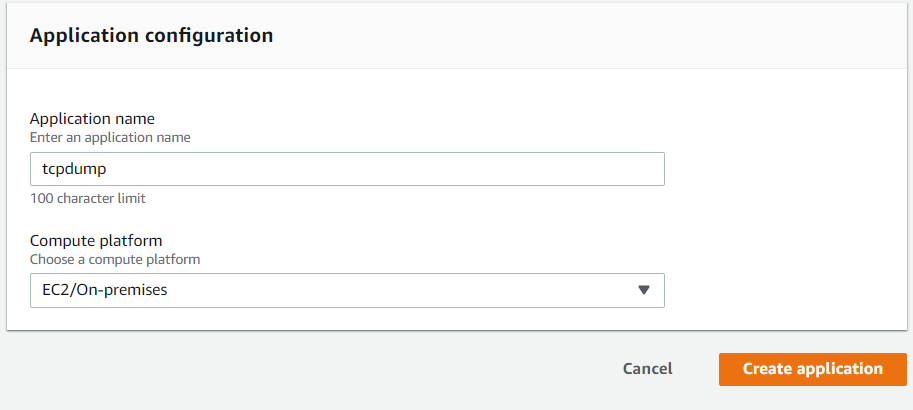
* 1. הגדרת תוצרי הבניה – לאן לשמור וכו'



* 1. הגדרת לוגים לבנייה

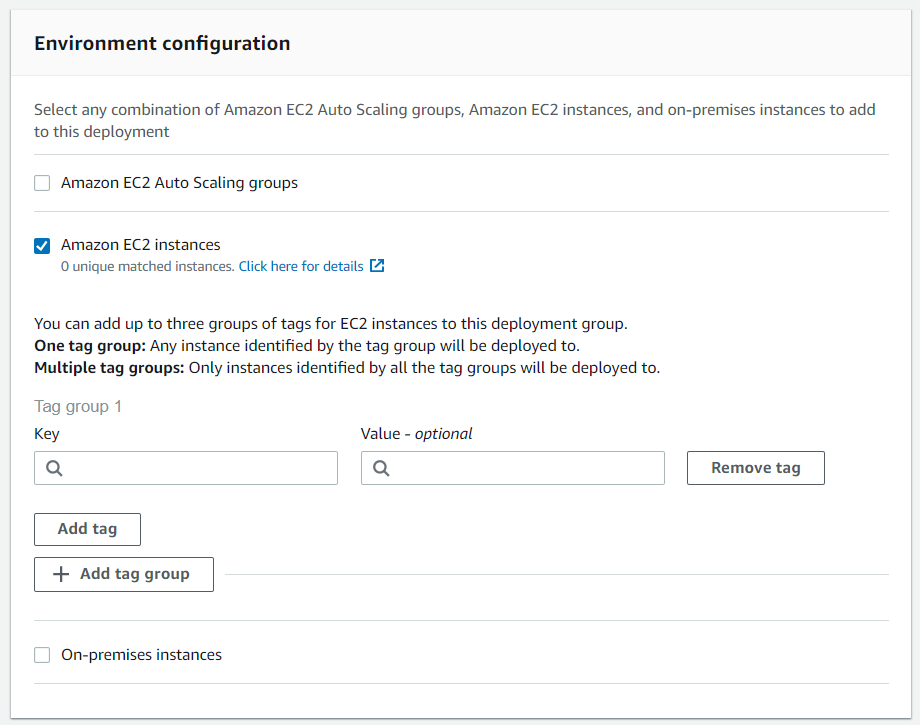


1. הגדרת פריסת גרסאות ב CodeDeploy
   1. נייצר אפליקציה חדשה

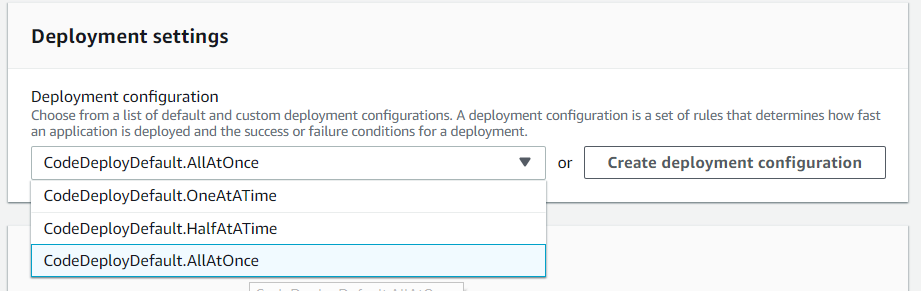


* 1. ניצר "קבוצת הפצה חדשה"

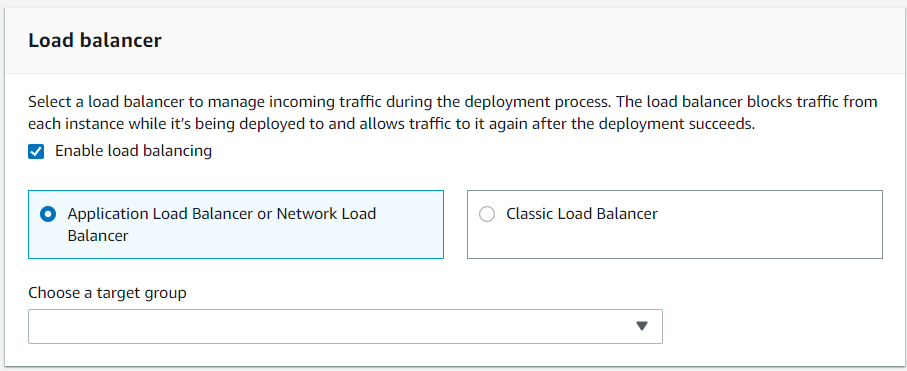
תחת קבוצת ההפצה נדרש להגדיר מה ה"יעד" שעליו נפרוס את האפליקציה



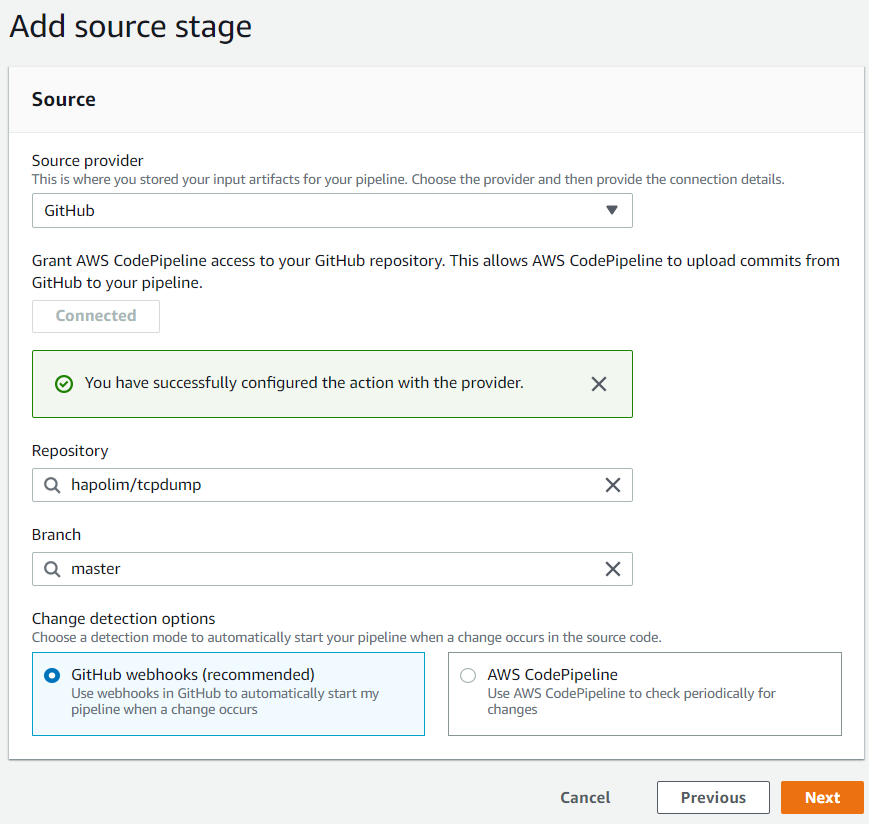
* 1. נידרש להגדיר שיטת הפצה –



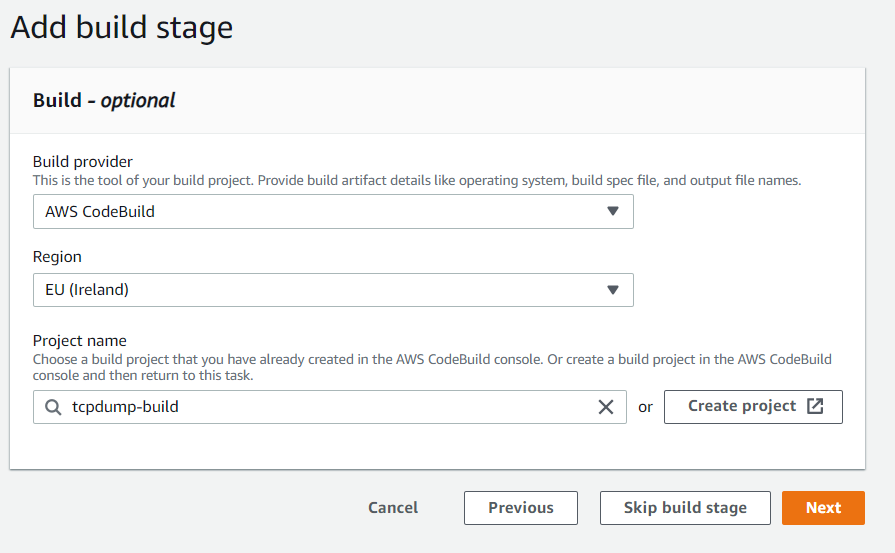
* 1. (אופציונלי) ניתן להגדיר Load balancer ע"מ שהפריסה באופן אוטומטי תוציא/תכניס את השרתים מPool .



1. Pipeline , חיבור של תהליך הBuild עם הפריסה.
   1. הגדרת Source control



* 1. הגדרת תהליך ה Build



* 1. הגדרת תהליך הפריסה

