



# CI/CD - GIVE APPLICATIONS AUTO-DEPLOY SUPERPOWERS

---

**Fundamentals and Benefits of CI/CD to Achieve, Build, and  
Deploy Automation for Cloud-Based Software Products**

# CI - CONTINUOUS INTEGRATION

- IN MODERN APPLICATION DEVELOPMENT, THE GOAL IS TO HAVE MULTIPLE DEVELOPERS WORKING SIMULTANEOUSLY ON DIFFERENT FEATURES OF THE SAME APP. HOWEVER, IF AN ORGANIZATION IS SET UP TO MERGE ALL BRANCHING SOURCE CODE TOGETHER ON ONE DAY, THE RESULTING WORK CAN BE TEDIOUS. THAT'S BECAUSE WHEN A DEVELOPER WORKING IN ISOLATION MAKES A CHANGE TO AN APPLICATION, THERE'S A CHANCE IT WILL CONFLICT WITH DIFFERENT CHANGES BEING SIMULTANEOUSLY MADE BY OTHER DEVELOPERS.
- CONTINUOUS INTEGRATION (CI) HELPS DEVELOPERS MERGE THEIR CODE CHANGES BACK TO A SHARED BRANCH, MORE FREQUENTLY. ONCE A DEVELOPER'S CHANGES TO AN APPLICATION ARE MERGED, THOSE CHANGES ARE VALIDATED BY AUTOMATICALLY BUILDING THE APPLICATION AND RUNNING DIFFERENT LEVELS OF AUTOMATED TESTING, TO ENSURE THE CHANGES HAVEN'T BROKEN THE APP. IF AUTOMATED TESTING DISCOVERS A CONFLICT BETWEEN NEW AND EXISTING CODE, CI MAKES IT EASIER TO FIX THOSE BUGS QUICKLY AND OFTEN.

# BENEFITS OF CONTINUOUS INTEGRATION

- PREVENT INTEGRATION PROBLEMS
  - INTEGRATION BUGS ARE DETECTED EARLY
  - ENFORCES DISCIPLINE OF FREQUENT AUTOMATED TESTING
  - FREQUENT CODE CHECK-IN PUSHES DEVELOPERS TO CREATE MODULAR, LESS COMPLEX CODE
  - AUTOMATE THE BUILD
- 
- IN CONCLUSION, IT'S WORTH INVESTING TIME AND MONEY INTO IMPLEMENTING CI. IT REDUCES THE STRESS ON YOUR DEVELOPERS AND ALLOWS THEM TO FOCUS ON THEIR CORE TASK, WHICH IS DEVELOPING NEW FEATURES.

# CD - CONTINUOUS DEPLOYMENT

- CONTINUOUS DEPLOYMENT AUTOMATES RELEASING AN APP TO PRODUCTION.
- IN PRACTICE, CONTINUOUS DEPLOYMENT MEANS THAT A DEVELOPER'S CHANGE TO A CLOUD APPLICATION COULD GO LIVE WITHIN MINUTES OF WRITING IT (ASSUMING IT PASSES AUTOMATED TESTING). THIS MAKES IT MUCH EASIER TO CONTINUOUSLY RECEIVE AND INCORPORATE USER FEEDBACK. TAKEN TOGETHER, ALL OF THESE CONNECTED CI/CD PRACTICES MAKE DEPLOYMENT OF AN APPLICATION LESS RISKY, WHEREBY IT'S EASIER TO RELEASE CHANGES TO APPS IN SMALL PIECES, RATHER THAN ALL AT ONCE. THERE'S ALSO A LOT OF UPFRONT INVESTMENT.

# BENEFITS OF CONTINUOUS DEPLOYMENT

- **DEPLOY CODE FAST**
- **IMPROVE QUALITY**
- **INCREASE EFFICIENCY**
- **RELEASE AUTOMATICALLY**
- **REDUCE COST**