

สรุปเกี่ยวกับ DevOps และ CI/CD

DevOps คืออะไร

DevOps คือ การรวมคำว่า Developer กับ Operator เข้าด้วยกัน โดยสมัยก่อน Developer เป็นคนพัฒนา code แต่ไม่มีสิทธิ deploy code ทำให้การทำงานยุ่งยาก เกิดความผิดพลาด และใช้เวลานานในการ deploy แต่ละครั้ง เพื่อแก้ปัญหาในการ deploy code จึงเกิดเป็นแนวทาง CI/CD โดยทำงานตั้งแต่การ Plan, Code, Build, Test, Release, Deploy, Operate, Monitor หรือบางที่เรียกสั้นๆ ว่า Pipeline

CI คืออะไร

CI (Continuous Integration) คือ กระบวนการรวม source code ของคนในทีมพัฒนาเข้าด้วยกัน และมีการ test ด้วย test script เพื่อให้แน่ใจว่าไม่มี error ในส่วนใดๆ ของโปรแกรม แล้วจึงทำการ commit ไปที่ branch master อีกต่อหนึ่ง โดยในการพัฒนานั้น มักใช้ Build Server มาช่วย กล่าวคือจะเริ่มทำการ Integration กันตั้งแต่เมื่อมีการเปลี่ยนแปลง Source Code ที่ Repository กลาง ระบบจะทำการตรวจสอบ Code หลังจากการเปลี่ยนแปลงว่าทำงานร่วมกันได้หรือไม่ตั้งแต่ Compile, Testing

CD คืออะไร

1. CD (Continuous Deployment) คือ การ Deploy ขึ้น production โดยจะทำทุกขั้นตอน ตั้งแต่ compile build ไปจนถึง deploy ขึ้น production แบบอัตโนมัติทั้งหมด
2. CD (Continuous Delivery) คือ การทำทุกขั้นตอนคล้ายกันกับ Continuous Deployment ต่างกันตรงที่จะไม่มีการ deploy ขึ้น production ขึ้นในทันที แต่จะเป็นการทำ manual deploy หรือจะเป็นแบบ one click deploy ก็ได้ หลังจาก QA หรือ ฝ่าย Business พอใจในตัว product ที่ทีมทำออกมา

ซึ่งสมัยนี้ก็มี tools ต่าง ๆ ที่ทำหน้าที่พวกนี้เยอะมากทั้ง On-Premise และ On-Cloud ที่เรารู้จักกันดี ก็น่าจะเป็น Jenkins ที่เข้ามามีบทบาทมากในการทำ CI/CD [1]

การพัฒนาซอฟต์แวร์

เริ่มแรกควรใช้ Git ในการจัดการ Source Code และ DevOps มีความเกี่ยวข้องกับ Git อย่างแรกเลยก็คือ มันช่วยให้ Developer สามารถทำงานร่วมกัน พร้อมกันได้ ช่วยเพิ่ม Productivity อย่างมาก

Code จาก Developer ที่ถูก Push ที่ส่งขึ้นไป Repository จะต้องมีการตรวจสอบความถูกต้องอยู่ตลอดเวลา เพื่อให้มั่นใจว่า เราไม่ได้ส่งโปรแกรมที่มีปัญหาไปให้ User วิธีการก็คือ การสร้างระบบที่จะถูก Trigger ให้รันโดยอัตโนมัติจากการ Push Code ขึ้นมานั่นเอง ระบบที่ว่านี้เรียกว่า Continuous Integration and Continuous Delivery (CI/CD) และเพิ่มคำ

ว่า Pipeline เข้าไป เพื่อบอกว่ามันเป็นการทำงานที่ต่อเนื่องการโดยอัตโนมัติ โดยขั้นตอนเหล่านี้จะรันบนเครื่อง Server ที่สร้างขึ้นมาเพื่อรันงานพวกนี้โดยเฉพาะ (Pipeline Runner)

DevOps จะต้องเขียน Script ว่า หลังจากมี Code ส่งขึ้นมาแล้ว จะต้องให้มันรันคำสั่งอะไรยังไงบ้าง โดยกระบวนการทำ CI/CD ก็มักจะมีขั้นตอนที่คล้ายๆ กัน ประมาณนี้

1. **Build** ขั้นตอนนี้คือการเตรียมโปรแกรมให้พร้อมสำหรับการใช้งาน ขั้นตอนนี้จะโหลด Dependency ต่าง ๆ แล้วรัน Code สำหรับ Build โปรแกรมให้พร้อมสำหรับการ Test
2. **Test** นำโปรแกรมที่ได้จากขั้นตอนก่อนหน้านี้มารัน Script เพื่อทดสอบการทำงาน รันให้ครบทุกอัน ให้มั่นใจว่าโปรแกรมายังทำงานได้ปกติอยู่แม้จะมีการเปลี่ยนแปลง Source Code ไป ถ้ามี Test ที่ไม่ผ่านขึ้นมา นั่นคือสัญญาณว่า Test ที่เราเขียนขึ้นมามันมีประโยชน์แล้ว นอกจากนั้นเรายังจับบั๊กได้ก่อนจะไปถึงมือ User ด้วย แต่ปกติจะต้อง Test ให้ผ่านทั้งหมด ซึ่งเมื่อผ่านแล้วก็จะไปที่ขั้นตอนต่อไป
3. **Package** ขั้นตอนนี้คือการเอาโปรแกรมมาจับใส่กล่องเตรียมพร้อมสำหรับการนำออกไปใช้งาน กล่องที่ว่าก็คือ Docker Image ที่นี้ก็พร้อมสำหรับการนำโปรแกรมนี้ไปรันที่ไหนก็ได้ที่มี Docker และส่ง Docker Image ที่ได้ไปไว้บน Host ที่เรียกว่า Docker Registry ซึ่งจะได้ URL มาหนึ่งอันสำหรับนำไปใช้งาน
4. **Deploy** สุดท้ายคือการเอาโปรแกรมไปรันบน Server จริง โดยเราติดต่อเข้าไปหา Server ที่เราใช้งาน อาจจะเป็น Kubernetes หรืออื่น ๆ เมื่อต่อเข้าไปแล้ว ให้เอา Docker Image URL ที่ได้มาจากขั้นตอนก่อนหน้านี้ เอามาใช้ใน Deployment ที่ต้องการ ต่อไประบบ Container Orchestration ก็จะทำการ Download Docker Image ลงมาสร้าง Container และทำการรัน เมื่อทุกอย่างเรียบร้อยแล้ว มันจะลบ Container เวอร์ชันเก่าโดยอัตโนมัติ เป็นอันสิ้นสุดกระบวนการ CI/CD [2]

อ้างอิง

- [1] Softmelt. (2021). การพัฒนาระบบตามแนวทาง CI/CD และ DevOps คืออะไร?, สืบค้นเมื่อ 26 กุมภาพันธ์ 2564, จาก <https://www.softmelt.com/article.php?id=664>
- [2] Kriangkrai Chaonithi. (2019). DevOps คืออะไร นำมาประโยชน์ได้ยังไง และตัวอย่างการทำ DevOps, สืบค้นเมื่อ 26 กุมภาพันธ์ 2564, จาก <https://www.spicydog.org/blog/introduction-to-devops-and-the-practical-use-cases-at-credit-ok/>