

1. DevOps คืออะไร มีความสัมพันธ์กับการพัฒนาซอฟต์แวร์อย่างไร

ในกระบวนการผลิตซอฟต์แวร์ อาจจะมีการขัดแย้งกันระหว่าง 2 ฝ่ายหลักๆ (1) คือ ฝ่าย Development กับ ฝ่าย Operations เนื่องจากฝ่าย Development ต้องการผลิตซอฟต์แวร์อย่างรวดเร็วส่งมอบให้ทันเวลา และฝ่าย Operations ที่ทำหน้าที่ดูแลด้านความเสถียรของซอฟต์แวร์เพื่อให้มีความผิดพลาดน้อยที่สุด ทำให้ในกระบวนการผลิตจึงจะต้องเลือกระหว่างความเสถียรภาพหรือความเร็ว และเพราะการต้องการทั้งสองอย่าง จึงเกิดการรวมกันของ Development (Dev) + Operation (Ops)(2) โดยมีจุดมุ่งหมายคือ เพื่อให้สามารถนำงานขึ้น Production ได้เร็วที่สุด รวมถึงส่งมอบงานที่มีคุณภาพและความเสถียรของตัวระบบมากที่สุด

DevOps ย่อมาจาก Development Operations (3) เป็นการที่เชื่อมกระบวนการพัฒนาโปรแกรม การส่งขึ้นไป Deploy ให้ผู้ใช้เข้าถึงได้อย่างมีประสิทธิภาพ มีคุณภาพ ไปจนถึงการติดตามสถานะการทำงานของโปรแกรมเพื่อนำกลับมาพัฒนาปรับปรุงกระบวนการพัฒนาโปรแกรม เปรียบได้กับ System Admin ในสมัยก่อน ที่ทำหน้าที่เอาโปรแกรมขึ้นไบนารีบน Server ให้ได้ แล้วคอยดูไม่ให้ระบบล่ม แต่ DevOps ได้มีการพัฒนาทำเรื่อง Automation ด้วย คือแก้ไขกระบวนการทั้งหมดที่ต้องทำมือในสมัยก่อน ให้สามารถทำงานโดยอัตโนมัติได้

2. CI/CD คืออะไร มีความสัมพันธ์กับ DevOps อย่างไร

CI/CD คือ กระบวนการที่ช่วยในพัฒนา Software ให้มีประสิทธิภาพมากขึ้น ทั้งในแง่ของระยะเวลาการพัฒนาและคุณภาพของ Software (4) โดย CI ย่อมาจาก Continuous Integration(CI) คือ กระบวนการรวม source code ของทีมพัฒนาเข้าด้วยกัน และมีการ test ด้วย test script เพื่อให้แน่ใจว่าสามารถใช้งานได้จริง และไม่มีข้อผิดพลาด ในส่วนใดๆ ของโปรแกรม จึงทำการ commit ไปที่ branch master (5) โดยในการพัฒนานั้น นิยมใช้ Build Server เข้ามาช่วย โดยจะเริ่มทำการ Integration เมื่อมีการเปลี่ยนแปลง Source Code ที่ Repository กลาง ระบบจะเริ่มทำการตรวจสอบ Code หลังจากการเปลี่ยนแปลง

CD ย่อมาจาก Continuous Delivery หรือ Continuous Deployment กระบวนการที่ช่วยเหลือสามารถ ส่งมอบซอฟต์แวร์ ที่ผ่านขั้นตอน CI แล้ว สามารถ Deploy Software ได้อย่างมีประสิทธิภาพ โดยการนำ Source Code ที่ผ่านการ Build และ Testing มาแล้ว ที่อาจจะอยู่ในรูปแบบที่แตกต่างกัน Deploy ขึ้นไปอยู่บน Server ตามที่เราต้องการและสามารถใช้งานได้ถูกต้อง ซึ่ง CD นั้นแบ่งออกเป็น 2 แบบ คือ แบบ Automate และแบบ Manual ซึ่งแตกต่างกันที่แบบ Automate จะสามารถ deploy ได้แบบอัตโนมัติ และสามารถ deploy ได้ทันที แต่แบบ Manual จะสามารถ deploy ได้หลังจากมีการอนุมัติจากผู้รับรอง ที่ จะทำการตรวจสอบก่อนอนุญาตให้นำขึ้น server

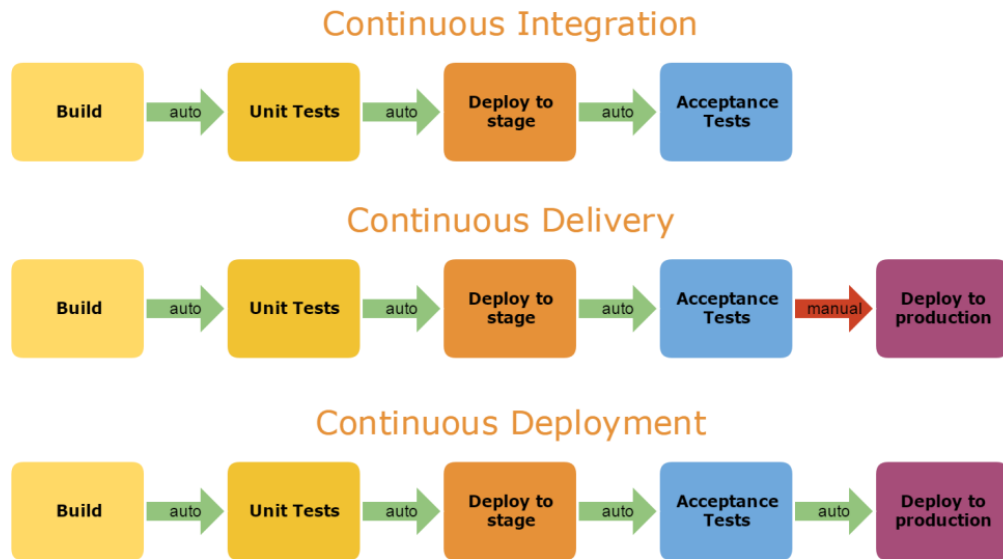


Figure from <https://clm-consulting.com/services/it-transformations/>

DevOps มีความสัมพันธ์กับขั้นตอนในส่วน CI เพราะทุกการเปลี่ยนแปลงต้องมีการทดสอบแบบอัตโนมัติ จึงใช้ DevOps เข้ามาช่วยในการทดสอบซึ่งการนำ DevOps มาใช้กับ CI มีข้อดีคือ สามารถรับรู้สถานะของระบบได้ตลอดเวลา และทราบความคืบหน้าของงานได้ แต่หากเขียน Script Test ไม่ดีพอ อาจจะทำให้การ deploy ของระบบผิดพลาดได้

3. หากองค์กรต้องการนำเอา DevOps และ CI/CD เข้ามาเป็นส่วนเสริมในกระบวนการพัฒนาซอฟต์แวร์จะต้องทำอย่างไรบ้าง

การนำ DevOps มาใช้ (6) ต้องระบุปัญหาก่อนว่ามีอะไรบ้าง และทำการเรียกลำดับตามความสำคัญหรือผลกระทบที่เกิดขึ้นว่าจะแก้ไขอะไรก่อนหลัง จากนั้นลดความหลากหลายของการทำงาน เช่น Environment และเครื่องมือต่าง ๆ ควรจะเป็นในแนวเดียวกัน ควรรู้ขั้นตอนการทำงานตั้งแต่ต้นจนจบ จากนั้นทำการสร้างเอกสาร เพื่ออธิบายการทำงานตั้งแต่ต้นจนจบไว้ จากนั้น share ให้ทุกคนที่เกี่ยวข้องจะได้ทำความเข้าใจ แก้ไขหรือปรับปรุงกันไป จะทำให้เราเห็นว่า การทำงานปัจจุบันเป็นอย่างไรตรงไหนมีปัญหา และควรแก้ไขที่ใด

ขั้นตอนต่อมา เข้าสู่กระบวนการพัฒนาและส่งมอบ software การพัฒนาและส่งมอบ software ที่ดี ประกอบไปด้วย การจัดการ source code หรือ Source Control Management (SCM) กระบวนการ build ของ software กระบวนการ deploy software การ configuration หรือ provisioning environment ต่าง ๆ สำหรับการ deploy software ซึ่งเราอาจจะต้องกำหนดการทำงานเหล่านี้ให้ชัดเจน หรือดีที่สุดคือ ช่วยกันสร้างระบบที่ทำงานแบบอัตโนมัติขึ้นมาให้มากที่สุด เพื่อช่วย validate ว่า software

ของเรายังคงทำงาน ได้อย่างถูกต้องตามที่คาดหวัง ที่สำคัญต้องทำงานได้รวดเร็วด้วย จากนั้นพิจารณาความต้องการขององค์กรว่าต้องการเพิ่มในส่วนไหนเข้าไปอีก

ในส่วนของการนำ CI/CD มาใช้นั้น ทีมพัฒนาระบบของเราจะแบ่ง environment เป็น 4 ส่วน (7) คือ Local , Alpha , Staging , Production มีขั้นตอนดังนี้

1. Developer เมื่อทำการพัฒนา feature เสร็จ จะทำการ build, test และ run บนเครื่องของตัวเอง (Local) เพื่อให้แน่ใจว่าระบบทำงานได้ถูกต้องและให้แน่ใจว่าสิ่งที่เปลี่ยนแปลงไม่กระทบส่วนอื่น ๆ
2. ทำการดึง source code ล่าสุดจาก Repository ของระบบ เพื่อตรวจสอบว่ามีการเปลี่ยนแปลงหรือไม่ถ้ามีการเปลี่ยนแปลงก็ให้ทำ merge ที่เครื่องของ Developer ก่อน จากนั้นจึงทำการ build, test และ run อีกรอบ เมื่อทุกอย่างผ่านทั้งหมด ให้ทำการส่งการเปลี่ยนแปลงไปยัง Repository กลาง
3. เมื่อ Repository กลางมีการเปลี่ยนแปลง จะต้องมีการ CI ทำการ build, หลังจาก build จะส่งต่อไป run unit testing ก่อน ถ้าผ่านทั้งหมดจึงจะส่งต่อไปยังระบบ Continuous Delivery เพื่อ deploy to alpha environment
4. เมื่อ source code ถูก deploy to alpha environment แล้วจะ trigger ไปสั่งให้ run job automated testing ใน level ของทดสอบ smoke test ซึ่งเป็นชุดทดสอบย่อย ๆ ไม่เยอะมากเฉพาะในส่วน of feature code ที่ถูก deploy มาเท่านั้น
5. หลังจาก run smoke test เสร็จแล้วถ้าเกิดว่า run มีบางส่วนไม่ผ่านทั้งหมดจะไม่ส่งต่อไปยังระบบ Continuous Delivery เพื่อ deploy to staging environment QA จะทำการ investigate ว่าเกิดจากอะไร เป็นที่ระบบมี Bug เกิดขึ้นจริงหรือไม่ ถ้ามี bug ก็ให้ dev แก้ไข และ deploy มาใหม่ วน loop ใหม่
6. กรณีหลังจาก run smoke test ผ่านทั้งหมดจะส่งต่อไปยังระบบ Continuous Delivery เพื่อ deploy to Staging environment เมื่อ source code ถูก deploy to staging แล้ว จะ trigger ไปสั่งให้ run job automated testing ใน level ของทดสอบ regression test และ QA ก็ทำการทดสอบ Acceptance testing ไปด้วยพร้อมๆกันที่ Staging environment นี้ เมื่อมีการ deploy ซ้ำ ๆ เพื่อ fixed bug จากที่ QA เจอ หรือที่พบเจอจากการ run regression test แล้ว fail ก็จะเป็นการวน loop ตั้งแต่ต้นจนจบ จนกระทั่ง ทุกอย่างผ่านทั้งหมด

อ้างอิง

1. Developer. DevOps คืออียังวะ ? | BorntoDev Digital Academy [Internet]. BorntoDev เริ่มต้นเรียนเขียนโปรแกรม ขั้นเทพ ! 2020 [cited 2021 Feb 25]. Available from: <https://www.borntodev.com/2020/05/15/devops-คืออียังวะ/>
2. Boontae K. DevOps คืออะไร? [Internet]. TAMA Blog. 2020 [cited 2021 Feb 25]. Available from: <https://blog.tamacorp.co/devops-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/>
3. Chaonithi K. DevOps คืออะไร นำมาประโยชน์ได้ยังไง และตัวอย่างการทำ DevOps ที่ Credit OK [Internet]. SPICYDOG's Blog. 2019 [cited 2021 Feb 25]. Available from: <https://www.spicydog.org/blog/introduction-to-devops-and-the-practical-use-cases-at-credit-ok/>
4. ทำความรู้จักกับ CI/CD services บน Google Cloud [Internet]. [cited 2021 Feb 26]. Available from: <https://cloud-ace.co.th/blog/w0w7e1-ci-cd-services-google-cloud>
5. Junthep T. เมื่อหัดทำ CI/CD ครั้งแรก [Internet]. Medium. 2018 [cited 2021 Feb 26]. Available from: <https://medium.com/@thraithep.a/%E0%B9%80%E0%B8%A1%E0%B8%B7%E0%B9%88%E0%B8%AD%E0%B8%AB%E0%B8%B1%E0%B8%94%E0%B8%97%E0%B8%B3-ci-cd-%E0%B8%84%E0%B8%A3%E0%B8%B1%E0%B9%89%E0%B8%87%E0%B9%81%E0%B8%A3%E0%B8%81-f4f79151311c>
6. แนวทางในการนำ DevOps มาใช้ปรับปรุงการพัฒนา software [Internet]. [cited 2021 Feb 26]. Available from: <https://www.somkiat.cc/think-before-start-devops/>
7. narissara. การนำ Automated testing มารวมเข้ากับ กระบวนการ CI/CD [Internet]. Medium. 2017 [cited 2021 Feb 26]. Available from: <https://medium.com/@narissara.tan/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%99%E0%B8%B3-automated-testing-%E0%B8%A1%E0%B8%B2%E0%B8%A3%E0%B8%A7%E0%B8%A1%E0%B9%80%E0%B8%82%E0%B9%89%E0%B8%B2%E0%B8%81%E0%B8%B1%E0%B8%9A-%E0%B8%81%E0%B8%A3%E0%B8%B0%E0%B8%9A%E0%B8%A7%E0%B8%99%E0%B8%81%E0%B8%B2%E0%B8%A3-ci-cd-58c97e31e230>