**โมเดล DevOps**

DevOps คือ การรวมแนวคิด แนวทาง และเครื่องมือ ที่ช่วยเพิ่มความสามารถในการส่งมอบ Product อย่างรวดเร็ว โดยพัฒนาและปรับปรุงให้กว่ากระบวนการพัฒนาซอฟต์แวร์แบบดั้งเดิม ความรวดเร็วนี้ช่วยให้บริการแก่ลูกค้าของตนได้ดีขึ้น และสามารถแข่งขันในตลาดได้อย่างมีประสิทธิภาพมากขึ้น[1]

**การทำงานของ DevOps**

ในโมเดล DevOps ทีมพัฒนาและทีมปฏิบัติการจะอร่วมงานเป็นทีมเดียวกันโดยที่เหล่าวิศวกรจะทำงานตลอดวงจรการทำงานของแอปพลิเคชัน ตั้งแต่การพัฒนาและการทดสอบไปจนถึงการปรับใช้และการปฏิบัติงาน และพัฒนาขอบเขตความสามารถที่ไม่ได้จำกัดอยู่ที่การทำงานเพียงอย่างเดียว

**การนำโมเดล DevOps มาใช้**

DevOps เป็นการขจัดกำแพงกั้นระหว่างทีมพัฒนาและทีมปฏิบัติการที่ปกติต่างคนต่างทำงาน สำหรับ DevOps ทั้งสองทีมจะทำงานด้วยกันเพื่อปรับให้อัตราผลผลิตของนักพัฒนาและความเชื่อถือได้ของทีมปฏิบัติการอยู่ในระดับที่เหมาะสม ทั้งสองทีมจะต้องพยายามสื่อสารกันบ่อยครั้ง เพิ่มประสิทธิภาพ และปรับปรุงคุณภาพของบริการที่มอบให้แก่ลูกค้า

**ข้อปฏิบัติของ DevOps**

**การบูรณาการอย่างต่อเนื่อง**

ข้อปฏิบัติในการพัฒนาซอฟต์แวร์โดยที่นักพัฒนานำการเปลี่ยนแปลงโค้ดของตนมารวมอยู่ในพื้นที่เก็บข้อมูลส่วนกลางอย่างสม่ำเสมอ ซึ่งหลังจากนั้นจะดำเนินการสร้างและทดสอบโดยอัตโนมัติ

**การส่งมอบอย่างต่อเนื่อง**

ข้อปฏิบัติในการพัฒนาซอฟต์แวร์โดยที่การเปลี่ยนแปลงโค้ดถูกสร้างขึ้น ทดสอบ และจัดเตรียมสำหรับการออกสู่การใช้งานจริงโดยอัตโนมัติ ซึ่งจะขยายตามการบูรณาการอย่างต่อเนื่องโดยนำการเปลี่ยนแปลงโค้ดทั้งหมดมาปรับใช้กับสภาพแวดล้อมการทดสอบและ/หรือสภาพแวดล้อมการใช้งานจริงภายหลังขั้นตอนการสร้าง

**ไมโครเซอร์วิส**

แนวทางการออกแบบในการสร้างแอปพลิเคชันเดียวโดยเป็นชุดของบริการขนาดเล็ก แต่ละบริการจะทำงานตามกระบวนการของตนและสื่อสารกับบริการอื่นผ่านอินเทอร์เฟซที่กำหนดไว้เป็นอย่างดีโดยใช้กลไกน้ำหนักเบา

**โครงสร้างพื้นฐานเป็นโค้ด**

ข้อปฏิบัติโดยที่โครงสร้างพื้นฐานได้รับการจัดเตรียมและจัดการโดยใช้โค้ดและเทคนิคการพัฒนาซอฟต์แวร์ เช่น การควบคุมเวอร์ชัน และการบูรณาการอย่างต่อเนื่อง โมเดลที่ขับเคลื่อนด้วย API ของคลาวด์ทำให้นักพัฒนาและผู้ดูแลระบบสามารถตอบโต้กับโครงสร้างพื้นฐานได้ในทางโปรแกรมในขนาดต่างๆ แทนที่จะต้องมาตั้งค่าและกำหนดค่าทรัพยากรด้วยตนเอง

**การตรวจสอบและการบันทึก**

องค์กรจะตรวจสอบตัวชี้วัดและบันทึกต่างๆ เพื่อดูว่าประสิทธิภาพของแอปพลิเคชันและโครงสร้างพื้นฐานมีผลกระทบต่อประสบการณ์ของผู้ใช้ผลิตภัณฑ์ในขั้นปลายอย่างไร การรวบรวม จำแนกประเภท และวิเคราะห์ข้อมูลและบันทึกที่สร้างโดยแอปพลิเคชันและโครงสร้างพื้นฐาน

**การสื่อสารและการทำงานร่วมกัน**

การสื่อสารและการทำงานร่วมกันที่เพิ่มขึ้นในองค์กรเป็นประเด็นเชิงวัฒนธรรมข้อหนึ่งของ DevOps การใช้เครื่องมือ DevOps และการทำงานอัตโนมัติของกระบวนการส่งมอบซอฟต์แวร์ก่อให้เกิดการทำงานร่วมกันโดยนำลำดับการทำงานและความรับผิดชอบของฝ่ายพัฒนาและฝ่ายปฏิบัติการเข้ามาอยู่ด้วยกัน

**CI/CD**

Continuous Integration (CI) คือ กระบวนท่าที่ใช้สำหรับการรวบรวมซอฟต์แวร์ที่มีการพัฒนาแยกส่วนกันอย่างอัตโนมัติ อาจจะโดยหนึ่งหรือหลายนักพัฒนาก็ตามที สุดท้ายแล้วซอฟต์แวร์ที่พัฒนาชิ้นเล็กๆ ที่พัฒนาขึ้นมาจะต้องนำมารวมกันเป็นชิ้นใหญ่หนึ่งชิ้น

Continuous Deployment (CD) คือ ในทุกๆ ขั้นตอนจนถึงการ deployment ขึ้น production จะทำแบบอัตโนมัติทั้งหมด[2]

Continuous Delivery (CD) 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[3]

**ความสัมพันธ์ระหว่าง CI/CD กับ DevOps**

ในการเปลี่ยนแปลง แก้ไข เล็กๆ บ่อยๆ แล้วทำการทดสอบในทุกระดับ ทีม Development กับ ทีม Operation จะต้องคุยกันเพื่อลดความเข้าใจผิดในการเปลี่ยนแปลงหรือแก้ไข

**การนำเอา DevOps และ CI/CD เข้ามาเป็นส่วนเสริมในกระบวนการพัฒนาซอฟต์แวร์**

ในการพัฒนาซอฟต์แวร์จะรวมทีม Development กับ ทีม Operation เข้าด้วยกันเพื่อทำให้เพิ่มความสามารถในการส่งมอบ Product และบริการอย่างรวดเร็ว เมื่อถึงขั้นตอนที่ทำการรวมซอฟต์แวร์ที่มีการพัฒนาแยกส่วนจะทำแบบอัตโนมัติ และในการทดสอบหลังจากการรวมซอฟต์แวร์จนก่อนถึงขั้น Deployment จะเป็นการทำแบบอัตโนมัติ หลังจากนั้นจะเป็นไปตามรูปแบบของ CD ที่เลือกใช้

**อ้างอิง**

[1] “DevOps คืออะไร - Amazon Web Services (AWS),” *Amazon Web Services, Inc.* https://aws.amazon.com/th/devops/what-is-devops/ (accessed Feb. 25, 2021).

[2] narissara, “การนำ Automated testing มารวมเข้ากับ กระบวนการ CI/CD,” *Medium*, Dec. 10, 2017. https://medium.com/@narissara.tan/การนำ-automated-testing-มารวมเข้ากับ-กระบวนการ-ci-cd-58c97e31e230 (accessed Feb. 25, 2021).

[3] Administrator, “การพัฒนาระบบตามแนวทาง CI/CD และ DevOps คืออะไร?” https://www.softmelt.com/article.php?id=664 (accessed Feb. 25, 2021).