**นิยามโมเดล DevOps**

DevOps คือการผสมผสานแนวความคิดเชิงวัฒนธรรม แนวทางปฏิบัติ และเครื่องมือต่างๆ ที่ช่วยเพิ่มความสามารถขององค์กรในการส่งมอบแอปพลิเคชันและบริการอย่างรวดเร็ว โดยพัฒนาและปรับปรุงผลิตภัณฑ์ต่างๆ ให้เร็วกว่ากระบวนการการพัฒนาซอฟต์แวร์และการจัดการโครงสร้างพื้นฐานแบบดั้งเดิม ความรวดเร็วนี้ช่วยให้องค์กรสามารถให้บริการแก่ลูกค้าของตนได้ดีขึ้น และสามารถแข่งขันในตลาดได้อย่างมีประสิทธิภาพมากขึ้น

**DevOps ทำงานอย่างไร**

สำหรับโมเดล DevOps ทีมพัฒนาและทีมปฏิบัติการจะไม่ทำงานแบบ “ต่างคนต่างทำ” อีกต่อไป บางครั้ง ทั้งสองทีมจะจับมือร่วมงานเป็นทีมเดียวกันโดยที่เหล่าวิศวกรจะทำงานตลอดวงจรการทำงานของแอปพลิเคชัน ตั้งแต่การพัฒนาและการทดสอบไปจนถึงการปรับใช้และการปฏิบัติงาน และพัฒนาขอบเขตความสามารถที่ไม่ได้จำกัดอยู่ที่การทำงานเพียงอย่างเดียว

ในบางโมเดลของ DevOps ทีมประกันคุณภาพและทีมรักษาความปลอดภัยอาจทำงานรวมกับทีมพัฒนาและทีมปฏิบัติการอย่างใกล้ชิดยิ่งขึ้นตลอดวงจรการทำงานของแอปพลิเคชัน ในเวลาที่การรักษาความปลอดภัยเป็นจุดสำคัญของทุกคนในทีม DevOps เราจะเรียกว่า DevSecOps

ทีมต่างๆ ใช้ข้อปฏิบัติในการเปลี่ยนกระบวนการต่างๆ ที่เคยทำงานแบบแมนนวลและเชื่องช้าให้ทำงานอัตโนมัติ พวกเขาใช้ชุดเทคโนโลยีและเครื่องมือต่างๆ ที่ช่วยให้พวกเขาดำเนินการและพัฒนาแอปพลิเคชันได้อย่างรวดเร็วและเชื่อถือได้ นอกจากนั้น เครื่องมือเหล่านี้ยังช่วยให้วิศวกรเหล่านี้ทำงานได้อย่างอิสระ (เช่น การปรับใช้โค้ด หรือการจัดเตรียมโครงสร้างพื้นฐาน) ที่โดยปกติแล้วจำเป็นต้องได้รับความช่วยเหลือจากทีมอื่น พร้อมทั้งยังช่วยทีมงานทำงานได้รวดเร็วยิ่งขึ้นอีกด้วย

**CI/CD**

**CI/CD** (Continuous Integration, Continuous Delivery) เป็นกระบวนการในการทำงาน ตั้งแต่การ Plan -> Code -> Build -> Test -> Release -> Deploy -> Operate -> Monitor หรือบางทีเรียกสั้นๆ ว่า Pipeline ซึ่งสมัยนี้ ก็มี tools ต่างๆ ที่ทำหน้าที่พวกนี้ เยอะมากทั้ง On-Premise และ On-Cloud ที่เรารู้จักกันดี ก็น่าจะเป็น Jenkins ที่เข้ามามีบทบาทมาก ในการทำ CICD

ขอบเขตของการทำงาน ของ DevOps แต่ละที่เท่าที่ผมได้เคยลงไปสัมผัส จะไม่เหมือนกัน ขึ้นอยู่ว่า scope ที่ทำได้ มีระดับไหน บางที่ ก็คือทำตั้งแต่ต้นน้ำ ยันปลายน้ำ คือตั้งแต่วางแผน สร้าง Infrastructure เอง ทำ ENV ให้ Dev ใช้ เขียน Pipeline ตลอดจน ทำ Load Test, Performance Test, Security Test และระบบ Monitor & Alert เองทั้งหมด แบบนี้ก็ดีตรงที่จะรู้และเข้าใจ ในแต่ละส่วนอย่างดี ทำให้งานออกมามีประสิทธิภาพ ควบคุมได้ แต่ถ้ามี หลายๆ Project ก็คงทำแบบนี้ไม่ไหว ..

บางที่ DevOps จะมีหน้าที่แค่ทำระบบให้ Dev มาใช้งาน แต่จะไม่มีสิทธิ ในการทำอย่างอื่นเอง แบบด้านบน แบบนี้ ก็จะทำให้ ควบคุมอะไรไม่ได้ทั้งหมด แต่ถ้ามีหลายๆ Project ก็จะรองรับการทำงาน ได้เต็มที่ ..

สำหรับแนวคิด และ Tools ในการทำงานแบบ DevOps (CI/CD) ไม่มีแบบไหนผิด แบบไหนถูก ขึ้นอยู่กับการเอามาประยุกต์ใช้งาน ให้เหมาะสมกับงานของเรา องค์กรของเรา เพื่อทำให้งาน เกิดประสิทธิภาพสูงสุด ตอบโจทย์ผู้ใช้งาน product ของเราให้ดีที่สุด .. และที่สำคัญ ต้อง Monitor ได้ ต้องมี Dashboard เอาไว้ Tracking Metric ต่างๆ ได้

**ประโยชน์จากการนำ DevOps ไปใช้งาน**

คำถามที่พบได้บ่อยๆก็คือบริษัทจะได้ประโยชน์อะไรเมื่อนำ DevOps ไปปรับใช้ จาก [State of DevOps Report ปี 2016](https://puppet.com/resources/whitepaper/2016-state-of-devops-report) ที่นำบริษัทที่ใช้ DevOps กับไม่ใช้มาเปรียบเทียบกัน ได้ผลลัพธ์ดังนี้

* สามารถ deploy software ได้บ่อยกว่า 200 เท่า
* สามารถแก้ปัญหาที่เกิดขึ้นกับระบบได้ไวกว่า 24 เท่า
* สามารถลดอัตราล้มเหลวของการ change ระบบได้มากกว่า 3 เท่า
* สามารถลดเวลาในการผลิต software ได้มากกว่า 2,555 เท่า
* สามารถลดเวลาในการแก้ปัญหาด้าน security ได้มากกว่า 50%
* สามารถลดเวลาการทำงาน ทำให้พนักงานมีเวลาเพิ่มขึ้นมากกว่า 29 %
* สามารถลด cost ให้บริษัทได้มากกว่า จาก cost ที่เกิดขึ้นเมื่อระบบมีปัญหา และ cost จากการจ้างบุคลากร

**แหล่งอ้างอิง**

หนังสือ DevOps Handbook : How to Create World-Class Agility, Reliability, and Security in Technology Organizations (2016). โดย Gene Kim, Jez Humble, Patrick Debois, John Willis.

หนังสือ The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win (2013). โดย Gene Kim, Kevin Behr, George Spafford.