



หลักสูตร "พัฒนาซอฟต์แวร์ และสร้างทีม Dev สมัยใหม่ด้วย Git, Docker และ DevOps"

วันที่ 15-18 ตุลาคม 2562 (4 วัน)

สถานที่: The Connecion@MRT Ladphrao กรุงเทพฯ

หลักการและเหตุผล

เนื่องจากปัจจุบันเทคโนโลยีการพัฒนา และสร้างซอฟต์แวร์มีการเปลี่ยนแปลงไปอย่างรวดเร็ว รวมถึงมีการเพิ่มคุณเทคโนโลยีใหม่ๆ มากมาย ทำให้นักพัฒนาที่มีความจำเป็นต้องเรียนรู้ ศึกษา และต้องสามารถนำคุณสมบัติใหม่ๆ มาปรับใช้กับการพัฒนาซอฟต์แวร์ขององค์กรได้ เทคโนโลยีใหม่นี้ จะทำให้การสร้าง และพัฒนาซอฟต์แวร์เป็นไปได้อย่างขึ้น สามารถส่งมอบผลงานได้เร็วกว่าเดิม มีประสิทธิภาพ เหมาะกับการทำงานเป็นทีม และมีความปลอดภัยสูง ด้วยเหตุผลนี้จึงได้จัดทำหลักสูตร "พัฒนาซอฟต์แวร์ และสร้างทีม Dev สมัยใหม่ด้วย Git, Docker และ DevOps" ขึ้น เพื่อให้นักพัฒนาได้นำความรู้ไปปรับใช้ และพัฒนาองค์กรต่อไป

วัตถุประสงค์ของหลักสูตร

- เพื่อให้ผู้เรียนสามารถเข้าใจหลักการทำงานของ Git และสามารถนำไปปรับใช้กับตัวเอง และทีมได้
- เพื่อให้ผู้เรียนมีความรู้ และใช้งานระบบปฏิบัติการ Linux (Ubuntu) การตั้งค่า และคำสั่งพื้นฐานต่างๆ ได้
- เพื่อให้ผู้เรียนมีความรู้ ความเข้าใจ เกี่ยวกับ Docker , Docker Compose และสามารถนำไปใช้งานจริงได้
- เพื่อให้ผู้เรียนเรียนรู้หลักการของ DevOps และสามารถนำไปใช้กับองค์กรได้
- เพื่อให้ผู้เรียนเข้าใจหลักการของ CI/CD เครื่องมือต่างๆ และสามารถประยุกต์ใช้กับการพัฒนาซอฟต์แวร์ได้
- เพื่อให้ผู้เรียนสามารถติดตั้ง Self-hosted Git เพื่อใช้ในองค์กรของตัวเองได้
- เพื่อให้ผู้เรียนสามารถติดตั้ง Container Registry เพื่อใช้งานในองค์กรของตัวเองได้
- เพื่อให้ผู้เรียนสามารถนำเทคโนโลยีใหม่ๆ ทั้งในส่วน Frontend และ Backend เช่น Angular, React, Node.js, PHP, ASP.NET Core มาสร้าง ทดสอบ และส่งมอบซอฟต์แวร์แบบอัตโนมัติได้
- เพื่อให้ผู้เรียนสามารถใช้งาน GitLab เพื่อเขียนโค้ด และทำงานร่วมกันเป็นทีมได้
- เพื่อให้ผู้เรียนได้เข้าใจถึงหน้าที่ และบทบาทต่างๆ ด้วยการทดลองทำงานเป็นทีม
- เพื่อให้ผู้เรียนเรียนรู้หลักการของ Kubernetes เบื้องต้น และสามารถนำไปใช้งานได้
- เพื่อให้ผู้เรียนเข้าใจหลักการทำงานของ Prometheus และการทำ Monitoring ให้กับระบบได้

พื้นฐานผู้เรียน

- เคยเขียนโปรแกรม หรือเคยพัฒนาซอฟต์แวร์มาก่อน

เนื้อหาการเรียนและตารางเรียน 4 วัน

เวลา	เนื้อหา
วันที่ 1	
09.00 – 10.30 น.	<p>- การเตรียมเครื่องมือและโปรแกรมต่างๆ</p> <ul style="list-style-type: none"> ● ติดตั้ง Git และตั้งค่า ● ติดตั้ง Git GUI ● ติดตั้ง Visual Studio Code (Editor) ● ติดตั้ง Extensions ของ Visual Studio Code ที่จำเป็น ● ติดตั้ง ssh client ● การตั้งค่าต่างๆ <p>- การใช้งาน Git พื้นฐาน</p> <ul style="list-style-type: none"> ● คำสั่ง Git พื้นฐาน และการตั้งค่าต่างๆ ● ทำความรู้จักกับไฟล์ .gitignore ● การสร้าง Git Repository ● โครงสร้างของ Repository ● การเพิ่มไฟล์เข้าไปยัง staging area ● การ Commit ไฟล์ต่างๆ ● การ Undo ไฟล์ต่างๆ ● การทำ Stashing
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	<p>- การใช้งาน Git (ต่อ)</p> <ul style="list-style-type: none"> ● การสร้าง Branches ใหม่ ● การ Checkout Branches ● การลบ Branches ● การ Merging branches ● การ push โค้ดไปยัง remote repository ● การ pull และ fetch โค้ดจาก remote repository ● การใช้งาน Tags ● การใช้งาน Rebasing ● การใช้งาน Git Flow ด้วย Git GUI
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	<p>- การใช้งาน GitLab พื้นฐาน</p> <ul style="list-style-type: none"> ● สมัครสมาชิก GitLab ฟรี ● การตั้งค่าต่างๆ ของ GitLab ● ทำความรู้จักกับ GitLab Workflow ● การสร้างกลุ่มของโปรเจค ● ทดลองสร้างโปรเจค หรือ Repository และ ทดลอง push โค้ดจาก Local ● การสร้างไฟล์ README, CHANGELOG

เวลา	เนื้อหา
	<ul style="list-style-type: none"> ● ทดลอง commit ไฟล์ต่างๆ ไปยัง GitLab ● เพิ่มสมาชิกเข้ามาในโปรเจค ● ทดลอง fetch โค้ดจาก GitLab ● ทดลองทำ Merge Requests ● การสร้าง Labels ● การใช้งาน Issues และ Issue tracking ● การสร้าง Milestones, Due date และ Issue Board ● การเชื่อมต่อกับ Slack เพื่อใช้แจ้งเตือน และคุยกันในทีม ● แนะนำฟีเจอร์อื่นๆ ของ GitLab เช่น Wiki, Snippets เป็นต้น <p>- คำสั่ง Linux พื้นฐาน (Ubuntu)</p> <ul style="list-style-type: none"> ● การสร้างไฟล์ใหม่ ● การแก้ไข และบันทึกไฟล์ ● การคัดลอก และลบไฟล์ ● การดูรายการไฟล์ต่างๆ ● ดู directory ปัจจุบัน ● แสดงเนื้อหาของไฟล์ ● การ start และ stop services ต่างๆ ● การอัปเดต และอัปเกรดระบบ ● คำสั่งอื่นๆ
14.30 – 14.45 น.	พักเบรก
14.45 – 17.00 น.	<p>- การใช้งาน Docker</p> <ul style="list-style-type: none"> ● Docker คืออะไร ● แนะนำการติดตั้ง Docker และ Docker Compose ● การใช้งานคำสั่งพื้นฐานของ docker ● ทำความเข้าใจกับไฟล์ “Dockerfile” ● การสร้าง Images และการใช้งาน Containers ● ทดลองสร้าง Images ● แนะนำ Docker Registry ● ทดลอง Push Docker Images ● ความรู้เกี่ยวกับ Docker Networking ● การใช้งาน Docker Compose เบื้องต้น
วันที่ 2	
09.00 – 10.30 น.	<p>- แนะนำ และเตรียม Workshop สำหรับการใช้งาน Git แบบทีม</p> <ul style="list-style-type: none"> ● แนะนำบทบาทของหัวหน้าทีม ● แนะนำบทบาทของสมาชิกในทีม ● แนะนำ Workflow ของ GitLab ● ให้ Workshop Repository เพื่อให้แต่ละทีมนำไปใช้ ● อธิบาย Workshop และฟีเจอร์ที่จะต้องทำ

เวลา	เนื้อหา
10.30 – 10.45 น.	พักเบรก
10.45 -12.00 น.	<p>- Workshop การใช้งาน Git และการทำงานเป็นทีม</p> <ul style="list-style-type: none"> • แบ่งกลุ่ม เพื่อทำงานเป็นทีม • เลือกหัวหน้าทีม และสมาชิกในทีม • หัวหน้าทีมตั้งค่า GitLab ทัวไป • หัวหน้าทีมเชิญสมาชิกในทีมร่วมพัฒนาซอฟต์แวร์ • หัวหน้าทีมสร้าง Repository และ Branches ต่างๆ • สมาชิกในทีมเขียนโค้ดและ push ทีเจอร์ต่างๆ • ทดลองสร้าง Issue และ Merge Request • ทดสอบ Workshop ที่ได้จนสามารถทำงานได้จริง
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	<p>- ติดตั้ง Self-hosted GitLab บน Ubuntu Server</p> <ul style="list-style-type: none"> • สมาชิกในทีมช่วยกันติดตั้ง GitLab Server ของกลุ่ม • ติดตั้ง Docker • ตั้งค่า GitLab พื้นฐาน และ Mail • ตั้งค่า SSH Key • ตั้งค่า Profile ของผู้ดูแลระบบ และสมาชิก • ทดลองสร้าง Repository ใหม่ <p>- ติดตั้ง GitLab Runner ให้กับ Server เพื่อเตรียมทำ CI/CD</p> <ul style="list-style-type: none"> • สมาชิกในทีมช่วยกันติดตั้ง GitLab Runner • จัดการ Registering Runners ให้กับ Server • ทดลองสร้าง Runner ให้กับกลุ่มโปรเจค <p>- ติดตั้ง Docker Registry ของกลุ่ม</p> <ul style="list-style-type: none"> • สมาชิกในทีมช่วยกันติดตั้ง และตั้งค่า Docker Registry
14.30 – 14.45 น.	พักเบรก
14.45 – 17.00 น.	<p>- ตั้งค่า Production Server</p> <ul style="list-style-type: none"> • สมาชิกในทีมช่วยกันทำการติดตั้ง Production Server • ทดลอง Deploy Web App ที่ทำขึ้นไปยัง Production Server (แบบ Manual)
วันที่ 3	
09.00 – 10.30 น.	<p>- แนะนำ DevOps</p> <ul style="list-style-type: none"> • DevOps คืออะไร • CI/CD คืออะไร • หลักการ และ การนำ CI/CD เพื่อใช้งานในกระบวนการพัฒนาซอฟต์แวร์ • แนะนำเครื่องมือต่างๆ สำหรับงาน CI/CD
10.30 – 10.45 น.	พักเบรก

เวลา	เนื้อหา
10.45-12.00 น.	<p>- การใช้งาน GitLab ร่วมกับ CI/CD</p> <ul style="list-style-type: none"> ● การตั้งค่า และใช้งาน Runners ● แนะนำ Pipelines และ การสร้าง Stages ● การใช้งาน Schedules ● การเขียนไฟล์ .gitlab-ci.yml ● หลักการ และการใช้ cache ● แนะนำ และใช้งาน Job Artifacts ● การใช้งาน Environment variables ● การตั้งค่าและใช้งาน Environment
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	<p>- CI/CD กับการใช้งานร่วมกับ Docker</p> <ul style="list-style-type: none"> ● การเขียน Docker Images ● การ Build Docker Images ● การใช้งาน Container Registry ● การนำ Docker Images ไปใช้งาน <p>- Workshop CI/CD อย่างง่าย (แบบเดี่ยว)</p> <ul style="list-style-type: none"> ● สร้าง Git Repository ของตัวเอง ● Clone git repository ของ Workshop ที่ให้ไป ● ทดลองแก้ไขโค้ด และ push โค้ดขึ้นไปยัง GitLab Repository ● ทดสอบ CI/CD ให้ผ่านทั้งหมด ● ทดลอง deploy โปรเจคที่ไป Cloud
14.30 – 14.45 น.	พักเบรก
14.45 – 17.00 น.	<p>- กรณีศึกษาการเขียนไฟล์ .gitlab-ci.yml</p> <ul style="list-style-type: none"> ● สำหรับ build และ deploy React ● สำหรับ build และ deploy Angular ● สำหรับ build และ deploy Node.js ● สำหรับ build และ deploy PHP (Laravel) ● สำหรับ build และ deploy ASP.NET Core ● อื่นๆ
วันที่ 4	
09.00 – 10.30 น.	<p>- Workshop การนำ DevOps มาใช้งาน (แบบทีม)</p> <ul style="list-style-type: none"> ● สมาชิกในทีมทำความเข้าใจกับ Workshop (มีโจทย์ให้แต่ละกลุ่ม) ● สร้าง Repository ใหม่ ● ช่วยกันตั้งค่าการทำ CI/CD ● ช่วยกันเขียนไฟล์ .gitlab-ci.yml จากโจทย์ที่กำหนดให้ ● ทดสอบระบบ

เวลา	เนื้อหา
10.30 – 10.45 น.	พักเบรก
10.45-12.00 น.	- Workshop การนำ DevOps มาใช้งาน แบบทีม (ต่อ) <ul style="list-style-type: none"> • นำ Workshop ของทีม deploy ขึ้น production server แบบอัตโนมัติ • ทดลองเพิ่ม feature ใหม่ให้กับ Workshop • ทดสอบ และ deploy อีกครั้ง
12.00 – 13.00 น.	พักรับประทานอาหาร
13.00 – 14.30 น.	- Workshop การนำ DevOps มาใช้งาน (แบบเดี่ยว) <ul style="list-style-type: none"> • ทดลอง build และ deploy Workshop ของตัวเอง • ทดลองเพิ่ม feature ใหม่ให้กับ Workshop ของตัวเอง • Deploy ทดสอบอีกครั้ง
14.30 – 14.45 น.	พักเบรก
14.45 – 16.30 น.	- แนะนำ Kubernetes <ul style="list-style-type: none"> • Kubernetes คืออะไร • ทดลองใช้งาน และ deploy ไปยัง Kubernetes cluster • การดูรายงาน Metrics • วิเคราะห์ และ Monitoring ด้วย Prometheus • แนะนำ GitLab zero configuration (Auto DevOps) <p>- สรุป และปิดอบรม</p>

- ติดต่อ: codingthailand@gmail.com โทร: 085-4952624 (อ.เอก)

สเปคเครื่องคอมพิวเตอร์ที่ต้องการ

- ผู้เข้าอบรมนำ Notebook หรือ Mac มาเอง
- Ram อย่างน้อย 4 GB ขึ้นไป (แนะนำ 8 GB)