นิธิภัทร นิธิทวีกุล 57160525

โปรแกรม version control มีประโยชน์อย่างไร

ดเก็บการเปลี่ยนแปลงที่เกิดขึ้นกับไฟล์หนึ่งหรือหลายไฟล์เพื่อที่คุณสามารถเรียกเวอร์ชั่นใดเวอร์ชั่นหนึ่งกลับมาดูเมื่อไรก็ได้

* 1. ข้อได้เปรียบของ distributed version control เทียบกับ centralized version control คืออะไร

แบบ distributed version control

แต่ละคนไม่เพียงได้ก๊อปปี้ล่าสุดของไฟล์เท่านั้น แต่ได้ทั้งก๊อปปี้ของ repository เลย หมายความว่าถึงแม้ว่าเซิร์ฟเวอร์จะเสีย client ก็ยังสามารถทำงานร่วมกันได้ต่อไป และ repository เหล่านี้ของ client ยังสามารถถูกก๊อปปี้กลับไปที่เซิร์ฟเวอร์เพื่อกูข้อมูลกลับคืนก็ได้ การ checkout แต่ละครั้งคือการทำสำรองข้อมูลทั้งหมดแบบเต็ม

ในขณะที่แบบ centralized version control

ทุกอย่างรวมกันอยู่ที่เซิร์ฟเวอร์ที่เดียว ถ้าเซิร์ฟเวอร์นั้นล่มซักชั่วโมงนึง หมายความว่าในชั่วโมงนั้นไม่มีใครสามารถทำงานร่วมกันหรือบันทึกการเปลี่ยนแปลงงานที่กำลังทำอยู่ไปที่เซิร์ฟเวอร์ได้เลย หรือถ้าฮาร์ดดิสก์ของเซิร์ฟเวอร์เกิดเสียขึ้นมาและไม่มีการสำรองข้อมูลเอาไว้

* 1. ข้อได้เปรียบของ centralized version control เทียบกับ distributed version control คืออะไร

centralized version control ข้อมูลจะเป็นความลับกว่าเนื่องจากได้มาเพียงข้อมูลบางส่วนไม่ได้ทั้ง repository และสามารถช่วยกันทำงานจขากศูธนย์กลางได้

distributed version control ข้อมูลมีโอกาสลั่วไหลได้มากกว่าเนื่องจาก

ผู้ใช้แต่ละคนมี repository ทั้งหมด และอาจประสานงานกันได้ยากกว่า แบบแรกเนื่องจากไม่ได้ทำงานแบบรวมศูนย์

1.4บอกแนวทางการแก้ไข conflicit ที่เกิดขึ้นเมื่อมีการ merge โปรแกรมของผู้พัฒนาหลายคนเข้าด้วยกัน

วิธีแก้ก็แค่ลบพวกโค๊ดส่วนเกินออก แล้วแก้ไขใหม่ให้เรียบร้อย จากนั้นลองเช็คสถานะ จะขึ้น

1.5บอกแนวทางการลดconflicit ที่เกิดขึ้นเมื่อมีการ merge โปรแกรมของผู้พัฒนาหลายคนเข้าด้วยกัน

1. ทำการ Merge บ่อย

### การพูดคุย การสื่อสาร มันสำคัญอย่างมาก

### 3 [Mob programming](http://mobprogramming.org/)

เป็นวิธีการที่ทรงประสิทธิภาพอย่างมาก  
และเชื่อได้เลยว่า แก้ไขปัญหา Merge conflict ได้ 100%  
เนื่องจากทุกคนมานั่งทำงานด้วยกัน  
ใช้เครื่องทำงานเดียวกัน  
ดังนั้น ไม่มีทางที่ source code จะขัดแย้งกัน

1.6 git คืออะไร แตกต่างจากgit hub อย่างไร

**Git** คือ Version Control ตัวหนึ่ง ซึ่งเป็นระบบที่มีหน้าที่ในการจัดเก็บการเปลี่ยนแปลงของไฟล์ในโปรเจ็คเรา มีการ backup code ให้เรา สามารถที่จะเรียกดูหรือย้อนกลับไปดูเวอร์ชั่นต่างๆของโปรเจ็คที่ใด เวลาใดก็ได้ หรือแม้แต่ดูว่าไฟล์นั้นๆใครเป็นคนเพิ่มหรือแก้ไข หรือว่าจะดูว่าไฟล์นั้นๆถูกเขียนโดยใครบ้างก็สามารถทำได้

ส่วน git hub เป็นเว็บเซิฟเวอร์ที่ให้บริการในการฝากไฟล์ Git

1.7 จุดประสงค์หลักในการ branch คืออะไร

 เป็น feature ที่ช่วยให้นักพัฒนาสามารถที่จะทำงานได้สะดวกขึ้น ยกตัวอย่างเช่น เรามีโค๊ดที่ดีอยู่แล้ว แต่อยากจะทดลองอะไรนิดๆหน่อย หรือแก้ไขอะไรก็ตาม ไม่ให้กระทบกับตัวงานหลัก ก็เพียงแค่สร้าง branch ใหม่ขึ้นมา เมื่อแก้ไขหรือทำอะไรเสร็จแล้ว ก็ค่อยเซฟกลับมาที่ master เหมือนเดิม

* 1. Fast forward merge คืออะไรทำไมการ push ไปที่ repo จึงควรต้องทำ merge แบบนี้

ซึ่งหมายถึงหมายเลข commit ของ ตัวนึง ย้ายไปใช้ตัวเดียวกับของ ตัวนึงนั้นเอง ยังไม่ใช่การ merge ที่แท้จริงแต่อย่างใด

1.9 หน้าที่หลักของ git pull คืออะไร

git pull ใช้ดึงความเปลี่ยนแปลงจาก remote มายัง local และรวมเข้าด้วยกัน (มีค่าเท่ากับfetch+merge)