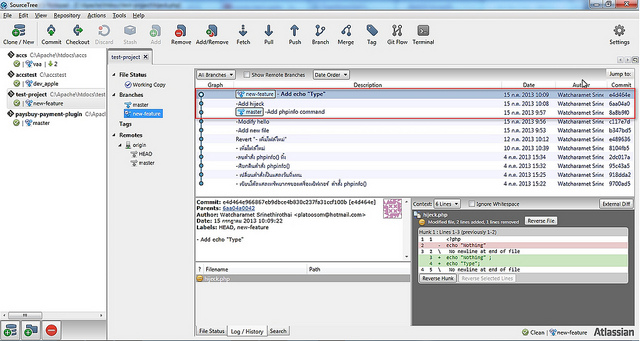
ขออธิบายคร่าวๆน่ะครับในการใช้งาน merg

การแตก branch ไม่ใช่เพียงเพื่อแตกออกมาให้ master มันมีแต่โค้ดที่ปราศจากบักเท่านั้น แต่เรายังสามารถแตก branch ออกมาเพื่อทดลองเขียนโค้ดส่วนตัว แบบว่า ลองดูซิถ้าเขียนอย่างนี้ๆมันจะได้มั้ย เราก็แตก branch ออกมา หลังจากเล่นกับโค้ดจนพอใจแล้ว เราก็ลบ branch นั้นทิ้งไป

การทดลองเขียนโค้ดของเราก็จะไม่ส่งผลกระทบใดๆกับเพื่อนร่วมงานต่อเมื่อแตก branch มาแล้วและพัฒนาโค้ดจนปลอดบักแล้ว จะปล่อยโค้ดที่ทำไปขึ้น production เราต้องรวมโค้ดจาก branch นั้นๆเข้ามาสู่ master เสียก่อน แล้วจึงเอาโค้ดจาก master ขึ้น production ต้องทำอย่างนี้เสมอนะครับ

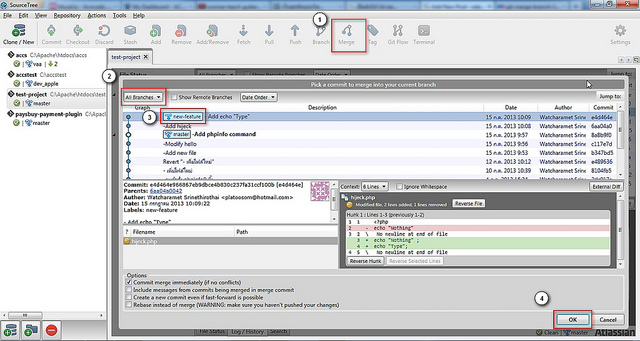
การ merge branch ทำได้ง่ายๆ เออ … การ merge branch ก็คือเอาโค้ดจากสองที่รวมเข้าด้วยกันนั่นเอง git มันจะวิเคราะห์ของมันเอง (เหมือนเดิม) ว่าบรรทัดไหนเพิ่มเข้ามาใหม่ ผมเองก็สงสัยอยู่ลึกๆว่าทำไมมันฉลาดจัง แต่ด้วยความขี้เกียจเลยไม่หาคำตอบ ซึ่งมันคงต้องไปดูอัลกอริธึ่มในการวิเคราะห์ของมัน

ถ้าท่านแตก branch ออกไปและมีการ commit ไปหลายๆครั้ง ท่านจะสังเกตเห็นอย่างหนึ่งว่า  master กับ branch ของท่านไม่ได้อยู่ตรงบรรทัดเดียวกัน

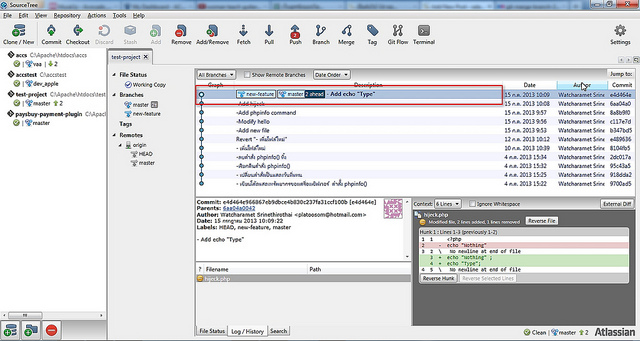
[](http://www.flickr.com/photos/platoosom/9287613199/)

รวม branch

ก่อนอื่นให้ checkout ไปที่ master ก่อน จากนั้นกดปุ่ม Merge > เลือก All Branches > คลิกที่ชื่อ branch ที่จะ merge > คลิกปุ่ม OK

[](http://www.flickr.com/photos/platoosom/9290427414/)

เมื่อไปดูที่หน้าหลัก จะสังเกตุเห็นว่า master มันวิ่งขึ้นไปอยู่บรรทัดเดียวกับ branch ที่ merge เข้ามา นั่นหมายความว่าการรวม branch เสร็จสมบูรณ์

[](http://www.flickr.com/photos/platoosom/9287678451/)

การรวม branch จะมีเหตุการณ์หนึ่งซึ่งอาจจะเกิดขึ้นได้ คือการชนกันของโค้ด เรียกด้วยภาษาทางเทคนิคเขาเรียก conflict เหตุการนี้จะเกิดขึ้นกับโปรเจ็กที่ร่วมกันทำหลายคนเท่านั้น ทำอยู่คนเดียวมันไม่เกิด เมื่อเกิด conflict โปรแกรมเมอร์จะต้องเป็นคนแก้ด้วยมือ ไว้ผมเจอแล้วผมจะสอนวิธีแก้ให้ มันไม่มีตัวอย่างผมก็ไม่รู้จะสอนยังไงเหมือนกัน