1. แบ้กอัพให้เราเอง ถ้าเราทำอะไรผิดขึ้นมา เราสามารถจะย้อนมันกลับไปตอนที่ยังทำงานได้ดีๆ อย่างสบายๆ
2. distributed version control เราสามารถแยก version control ออกเป็น repo ย่อยๆ ได้ แต่ละ repo ย่อยก็สามารถทำ versioning ของตัวเองได้อีกที ทำให้ สามารถทำงานแบบ offline ได้ ส่วน centralized version controlนั้น ทุกครั้งที่ commit นั้นต้องไปอัพเดท version กับ server ถ้า offline อยู่ หรือเชื่อมต่อกับ server ไม่ได้ ก็จะไม่สามารถ commit ได้
3. centralized version control จะใช้เลขจำนวนเต็ม(Integer) รันไปเรื่อยๆ ซึ่งแน่นอนว่า ยิ่งเรา commit บ่อยเท่าไร ก็จะทำให้เลข revision มีเยอะขึ้นๆ แต่ distributed version control จะใช้ hash ด้วย SHA-1 hash algorithm เนื่องจาก git จะมีการ commit สู่ local repo. ก่อน เพราะถ้าใช้เป็นเลขรันไปเรื่อยๆเหมือน centralized version control ก็จะทำให้เกิดเลข revision ซ้ำกันได้
4. แก้ไขการ Merge conflict ได้ง่าย ๆ มากเลยนะเพียงแค่เราคุยกันในเรื่องที่ควรจะคุยกันมากขึ้น, Mob programming
5. ทำการ Merge บ่อย ๆ สิ ,หนึ่ง class ใน หนึ่ง method นั้นควรจะมีหน้าที่การทำงานเพียงอย่างเดียวเท่านั้นหรือในแต่ละ class แต่ละ method ควรมีเหตุผลเดียวในการเปลี่ยนแปลงเท่านั้น
6. **Git** เป็นระบบ Version Control ตัวหนึ่งครับ **GitHub** เป็นผู้ให้บริการ Git เจ้าหนึ่ง คือจริงๆ เราไม่จำเป็นต้องใช้ GitHub เราก็ใช้ Git จัดการซอร์สโค๊ดในเครื่องเราคนเดียวได้ แต่ถ้าเครื่องเราพัง ข้อมูลก็หายด้วยไง เค้าเลยอัพข้อมูลไปแปะบน GitHub อีกที ข้อมูลจะได้ไม่หายเวลาเครื่องพัง
7. เพื่อให้สาขา master ปลอดภัย ไม่พังจากการแก้ไขโปรแกรมเรา
8. If Master has not diverged, instead of creating a new commit, git will just point master to the latest commit of the feature branch.This is a “fast forward.”
9. จริงๆแล้ว git pull ก็คือรวมโค๊ดจาก remote มายัง local โดยที่เราไม่สามารถรู้ได้เลยว่าจะรวมโค๊ดอะไรบ้าง รู้แค่หลังจาก pull เสร็จแล้วนั่นเอง ซึ่งจริงๆแล้ว git pull มันก็คือการทำ git fetch และต่อด้วย git merge อัตโนมัตินั่นเอง
10. การทำ version control