**1.** โดยส่วนตัวจะเลือกใช้ **Agile** เป็นหลักการในการพัฒนา software แบบใหม่ที่เน้น Rapid and flexible response to change ทำให้การพัฒนาว่องไว มีการทำเรื่อยๆไม่ต้องหยุด แม้มีอะไรมากระทบก็ไม่เป็นไร เมื่อมีการเปลี่ยนแปลง เราสามารถรองรับความเปลี่ยนแปลงนั้นได้อย่างรวดเร็ว ไม่ตายตัว

**วัตถุประสงค์ของ Agile**

เน้นว่าใครถนัดอะไร และการพูดคุยสื่อสารกัน มากกว่า การยึดติดที่เครื่องมือและกระบวนการ เช่นเปลี่ยนให้โปรแกรมเมอร์ไปคุยกับลูกค้าแทน ลูกค้าบอกอะไรมาก็ทำตามนั้นได้เลย

ให้ทำงานโดยยึดที่ผลผลิตหรือ software เป็นหลัก เช่น เดิมเน้นเอกสารแต่ Agile ไม่สนมากนัก แต่สนทีี่ว่าเรามี sw หรือของส่งให้ลูกค้าหรือยัง

ให้ความสำคัญเรื่องของการติดต่อสื่อสาร เช่น เดิมมีสัญญาหรือ contact กันแต่ Agile ไม่สนใจ ให้มองที่ความสัมพันธ์ระหว่างผู้พัฒนาและลูกค้า

ยอมรับความเปลี่ยนแปลง เช่น เดิมต้องวางแผนให้ครบเป็นอย่างดี และทำตามแผน(gantt chart) ให้ได้ แต่ Agile ไม่ต้องทำตามแผนแต่เน้นการสนองความเปลี่ยนแปลงที่เกิดขึ้นได้

**2.** ไม่มีโปรแกรมใดสำคัญกว่ากัน เพราะ โปรแกรมที่กล่าวมาทำงานคนหน้าที่และมีความสัมพันธ์กัน นำมาใช่ร่วมกันได้

**3.**

git checkout --orphan feature1

git add

git commit

git push feature1

**4.** ของแบบนี้ อยู่ที่ความรู้และความสารถของแต่ละคน

**5.** ผลลัพธ์ abcde"a".."e"

**6.**

**7.** MVC คือ สถาปัตยกรรมซอฟต์แวร์ชนิดหนึ่ง ซึ่งปัจจุบันมี Frameworks สำหรับสร้าง Web Apps จำนวนมาก แต่เกือบทั้งหมดมีโครงสร้างแบบ MVC ซึ่งแต่ละ Web Apps จะถูกแบ่งออกเป็นสามส่วน คือ

**1.Model** คือส่วนที่ใช้ในการติดต่อกับฐานข้อมูลซึ่งทำหน้าที่ในการดึงข้อมูลขึ้นมาจากฐานข้อมูลขึ้นมา และทำการจัดการข้อมูลต่างๆไว้ในรูปแบบที่เหมาะสม

**2.View** คือส่วนที่จะนำข้อมูลจาก Model ไปใช้แสดงผลให้ผู้ใช้เห็นผลลัพธ์ออกมาใน User Interface

**3.Controller** คือส่วนที่จะคอยรับ Input จาก Client เข้ามาแล้วนำคำสั่งไประมวลผล เพื่อสั่งงาน View และ Model ให้ประมวลผลออกมาเป็นอย่างไร

**กลไลการทำงาน**

1.เริ่มจาก Client ส่ง Request ไปที่ WebApp ซึ่งจะถูกส่งต่อให้ Controller    
ทำการตรวจสอบข้อมูลที่มาให้ (Request Method, Request Parameters)

2.แล้ว Controller จะเรียก Method ให้ทำงานเพื่อจัดการ Request นั้น

3.Model จะทำการคำนวณและอาจติดต่อกับ Database เพื่อจัดการกับ Request    
นั้น แล้วส่งผลลัพธ์กลับไปที่ Controller

4. เมื่อ Controller ได้ผลลัพธ์จาก Model แล้วก็ใช้ผลลัพธ์นั้นส่งต่อให้ View ทำงาน

5. View จะสร้าง Page สำหรับแสดงผลลัพธ์นั้น แล้วส่ง page กลับไปที่ Controller

6. Controller ส่ง Page นั้น (เป็น Response) กลับไปยัง Client

**8.** seagull framework (php)  เหมาะสำหรับเว็บขนาดใหญ่ สามารถเขียนออกเป็น module ต่างๆและนอกจากนั้น สามารถ ถอดและติดตั้ง module เพิ่มเข้าไปได้เช่นกัน นอกจากนี้แล้ว seagull framework ยังมีระบบ template อีกด้วย นับได้ว่าเป็นระบบที่ทำงานแบบ MVC ที่แบ่งการทำงานในแต่สะส่วนออกจากกัน รวมไปถึงรูปแบบของ code จะเป็น OOP มี CLASS , FUNCTION

**สรุป**- เป็น MVC Framework  
- มีระบบ template  
- ลดการเขียน code ที่ซ้ำไปซ้ำมา เพราะว่ามีการเขียนเป็น class , function  
- code ที่เขียนเป็น OOP  
- มีการวางโครงสร้างไฟล์ ไว้อย่างเป็นระบบ ระเบียบที่แน่นอน  
- มีระบบความปลอดภัยที่ดีกว่า การเขียนแบบธรรมดา  
- หา component อื่นๆที่มีคนเขียนไว้แล้วมาติดตั้งได้

**Ruby on rails framework : RoR หรือนิยมเรียกสั้นๆ ว่า เรลส์ (Rails)**เป็นระบบเฟรมเวิร์กสำหรับเว็บแอปพลิเคชันแบบโอเพนซอร์ส ที่พัฒนาด้วยภาษา Ruby โดยอิงแนวความคิดแบบ Model-View-Controller (MVC) เป็นสถาปัตยกรรมพื้นฐาน รูบีออนเรลส์ มีชื่อเสียงจากความง่ายในการใช้งาน โดยใช้การเขียนโปรแกรมจำนวนน้อยบรรทัดกว่า และมีการระบุค่าตั้งน้อยกว่าเฟรมเวิร์กที่ออกมาก่อนหน้าตัวอื่นๆ การติดตั้งรูบีออนเรลส์ทำได้โดยผ่านระบบ RubyGems ซึ่งเป็นระบบการจัดการส่วนประกอบในภาษารูบี้อย่างเป็นทางการ1  
         **ข้อดี**ของ framework ตัวนี้ คือ พื้นฐานสามารถ Generate webpage ได้คล้ายกับ CMS แต่มีการยืนหยุนในการใช้งานสูงกว่ามาก เหมาะมากสำหรับทำเป็น Web Application ใช้งานง่าย สะดวก โดยผู้ที่จะใช้งานต้องมีความรู้ด้านภาษา Ruby และความรู้เกี่ยวกับ CSS+MySQL+HTML บ้าง เพื่อให้การสร้างและประยุกติใช้งานต่างๆให้ตรงตามความต้องการมากยิ่งขึ้น

**จากการเปรียบเที่ยบอะไรจะดีกว่าหรือไม่นั้น อยู่ที่การเลือกใช้ งาน ความรู้พื้นฐานและความถนัดของผู้ใช้แต่ละคน**

**9.** Heroku เป็น Platform as a Service (Paas) ที่ให้เราใช้งานได้ฟรี (มีแบบเสียเงินด้วย) โดยรองรับภาษาโปรแกรมที่หลากหลาย เช่น Ruby, PHP, Node.js, Python, Java, Clojure, Scala และยังสามารถสร้าง buildpack สำหรับภาษาอื่นๆได้ เช่น Lua ที่รันอยู่บน OpenResty ได้อีกด้วย ทำให้การเขียน web ง่ายมากยิ่งขึ้น

10. เป็นวิชาว่าด้วยการออกแบบโปรแกรมคอมพิวเตอร์ตลอดจนการบริหารงานการพัฒนาโปรแกรมเพื่อให้ได้ผลิตภัณฑ์ซอฟต์แวร์ที่มีคุณภาพสูง ราคาถูกและภายในเวลาที่กำหนด  
การนำหลักวิทยาศาสตร์และคณิตศาสตร์มาช่วยพัฒนาซอฟต์แวร์ในเรื่อง 3 ประการคือ ให้ซอฟต์แวร์ถูกต้องเชื่อถือได้ การจัดการสร้างซอฟต์แวร์ให้มีคุณภาพ เพิ่มประสิทธิภาพการผลิตให้นักพัฒนานำไปใช้ในเรื่องต่างๆได้