1. Waterfall คือกระบวนการพัฒนาระบบซอฟต์แวร์ที่เป็นขั้นเป็นตอนในรูปลักษณะของชั้นน้ำตก หรือ ขั้นบันได นับจากบนสุดไปยังชั้นล่างสุดแต่ละชั้นจะกินระยะเวลานานมากเหมาะสำหรับระบบซอฟต์แวร์ที่มีความต้องการที่แน่นอนชัดเจน ไม่คลุมเคลือ เพราะแต่ละขั้นตอนจะไม่มีการวนกลับมาทำซ้ำอีกในขั้นที่สูงกว่า แต่บางกรณีที่แบบจำลองตัวนี้ถูกพัฒนาให้สามารถวนกลับมาทำซ้ำขั้นตอนข้างบนได้ชั้นนึง

เริ่มจากขั้นตอนแรกกันเลยนะครับ ชั้นบนสุด Requirement definition เป็นขั้นตอนการเก็บข้อมูลความต้องการของผู้ที่มีส่วนได้ส่วนเสียกับระบบ (Stakeholder)ที่เข้ามาเสนอให้เราพัฒนาระบบซอฟต์แวร์ให้ โดยจะมีวิธีการเก็บและวิเคราะห์ความต้องการ (Requirement Gathering and Analysis) การประเมินความเป็นไปได้ที่จะพัฒนาระบบซอฟต์แวร์นี้ (Project Feasibilities) หากมีความเป็นไปได้ที่จะพัฒนาระบบซอฟต์แวร์ นั่นหมายถึงว่างบประมาณและทรัพยากรเพียงพอที่จะรับงานเข้ามาก็ดำเนินไปสู่ขั้นตอนต่อไปคือการกำหนดหน้าที่ผู้รับผิดชอบในโครงการนี้โดยใช้ตารางกำหนดกิจกรรมหรือใช้ Gantt Chart

ผ่านจากขั้นตอนแรกก็เข้าสู่ขั้นตอนที่สอง System and Software Design การวิเคราะห์ออกแบบระบบจากความต้องการที่เราเก็บได้จากผู้ที่มีส่วนได้ส่วนเสียกับระบบ (Stakeholder) การวิเคราะห์ระบบซอฟต์แวร์ มาทำการวิเคราะห์ระบบโดยจะมีเครื่องมือและเอกสาร (Tool and Documentation) การวิเคราะห์ระบบโดยใช้แผนภาพเช่น Data flow Diagram , Flowchart อาจจะมีมากกว่านั้นตามความเล็กใหญ่ของระบบซอฟต์แวร์ หรือ ความซับซ้อนของระบบซอฟต์แวร์ เอกสารที่ได้จากการปฎิบัติงานในขั้นตอนนี้ก็คือ Software Requirement Specification ดู template Software Requirement Specification

แต่...

Agile

Rapid and flexible response to change

ทำให้การพัฒนาว่องไว

มีการทำเรื่อยๆไม่ต้องหยุด แม้มีอะไรมากระทบก็ไม่เป็นไร

เมื่อมีการเปลี่ยนแปลง เราสามารถรองรับความเปลี่ยนแปลงนั้นได้อย่างรวดเร็ว ไม่ตายตัว

2.ใช่ค่ะเพราะ github

1. ช่วยเก็บ code ของเรา online คล้ายๆ dropbox หรือ google drive แต่เก็บเฉพาะ code file
2. ช่วยให้เราเห็นข้อแตกต่างของ code ที่ถูกบันทึกในช่วงเวลาต่างๆ เช่น ก่อนหน้านี้ หรือ ที่ถูกพัฒนาโดยเพื่อนร่วมทีมหรือคนอื่น
3. เอื้ออำนวยให้การร่วมกันพัฒนา code หรือ project นั้นๆ ได้อย่างราบรื่น ยกตัวอย่างเช่น project A มีนักพัฒนาสามคนร่วมกันพัฒนาจาก original code แต่ละคนก็พัฒนาจาก original code แตกกิ่งก้านการพัฒนาออกไป เมื่อมีนักพัฒนาอยากจะพัฒนาต่อจากที่นักพัฒนาทำไว้แล้วก็สามารถทำได้ หรืออยากจะพัฒนาจาก original code ก็สามารถทำได้เช่นกัน
4. บันทึก code ได้ทั้งในเครื่องคอมพิวเตอร์ และ บน github ในเวลาเดียวกัน ง่ายๆ คือ การ backup การสมัครนั้นก็แสนจะง่ายครับเข้าไปลงทะเบียน [Github](https://github.com/) แล้วก็สามารถ download application ไว้ในเครื่องได้ค่ะ

3. $ git branch my-feature-branch

$ git branch

$ git checkout my-feature-branch

$ git branch

4. $ echo "Hi I was changed in master" >> hallo.txt

$ git commit -a -m "add line on hallo.txt"

$ git log --graph –all

$ git merge my-feature-branch

5. abcde”a”..”e”

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

-แล้ว Controller จะเรียก Method ให้ทำงานเพื่อจัดการ Request นั้น

-Model จะทำการคำนวณและอาจติดต่อกับ Database เพื่อจัดการกับ Request    
นั้น แล้วส่งผลลัพธ์กลับไปที่ Controller

-เมื่อ Controller ได้ผลลัพธ์จาก Model แล้วก็ใช้ผลลัพธ์นั้นส่งต่อให้ View ทำงาน

-View จะสร้าง Page สำหรับแสดงผลลัพธ์นั้น แล้วส่ง page กลับไปที่ Controller

-Controller ส่ง Page นั้น (เป็น Response) กลับไปยัง Client

8. framework เหมาะสำหรับเว็บขนาดใหญ่ สามารถเขียนออกเป็น module ต่างๆและนอกจากนั้น ท่านสามารถ ถอดและติดตั้ง module เพิ่มเข้าไปได้เช่นกัน นอกจากนี้แล้ว seagull framework ยังมีระบบ template อีกด้วย นับได้ว่าเป็นระบบที่ทำงานแบบ MVC ที่แบ่งการทำงานในแต่สะส่วนออกจากกัน รวมไปถึงรูปแบบของ code จะเป็น OOP ครับ มี CLASS , FUNCTION ซึ่งผมเขียนหรือผู้พัฒนา จำเป็นจะต้องรู้จักสิ่งเหล่านี้

Ruby เป็นภาษาคอมพิวเตอร์ Programming Language ที่มีหน้าตาน่าใช้ สะอาด ค่อนข้างเป็นที่นิยมในต่างประเทศ หลายเว็บไซต์ดังๆ พัฒนาด้วยเจ้านี่แหละ  
Rails เป็น Framework อาจจะมองว่าเป็นที่ ที่มีการวางกรอบการพัฒนาอย่างคร่าวๆ และมีการรวม Design Pattern ดีๆ เอาไว้ ทำให้การพัฒนา Software ขึ้นมามีระบบระเบียบที่ดี

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 จริงๆแล้ว มันเหมาะกับทุกคนแหละ เช่น นักศึกษาอยากลองเขียนเว็บด้วย PHP แต่ไม่ได้เช่า Hosting ก็สามารถใช้ Heroku ได้ หรือแม้แต่บริษัท Start up ที่ไม่อยากวางเครื่องเอง คอนฟิกเอง ก็ใช้ได้ เพราะมันสามารถ scale ให้รองรับผู้ใช้เยอะๆได้โดยง่าย

นอกจากรองรับภาษาโปรแกรมที่หลากหลายแล้ว ตัว Heroku มี App Store ของมันด้วยเรียกว่า [add-ons](https://addons.heroku.com/) สำหรับเพิ่มเติมบริการอื่นๆเข้าไปในแอปของเรา เช่น PostgreSQL, MongoDB, Redis เป็นต้น ซึ่งก็มีทั้งฟรี และไม่ฟรีให้เลือกใช้งาน มี app ตัวนึง ลองทําเล่นๆขึ้นมาชื่อ vchat ซึ่ง code อยู่ที่ [github repo](https://github.com/vtanathip/vChat" \t "_blank) และ application ที่ on air อยู่ที่ [vchat demo on heroku](http://vchat-demo.herokuapp.com/" \t "_blank)

10. เพื่อให้เรียนรู้ภาษาการเขียนโปรแกรมใหม่ๆ มีทางเลือกในการเขียนโปรแกรม, เว็บไซต์