

Kanban

จัดทำโดย

นายประกาศิต จำนงค์	รหัสนักศึกษา 55010706	กลุ่ม 2
นายประภัทร ถนอมศักดิ์	รหัสนักศึกษา 55010707	กลุ่ม 2
นางสาวประภาพรรณ กุลบุตร	รหัสนักศึกษา 55010711	กลุ่ม 2
นายพีรพล ตั้งขวัญพาณิชย์	รหัสนักศึกษา 55010877	กลุ่ม 2
นายองอาจ อรรถโสภณศักดิ์	รหัสนักศึกษา 55010386	กลุ่ม 2

เสนอ

ผศ. ดร.ชุตติเมษฐ์ ศรีนิลทา

รายงานนี้เป็นส่วนหนึ่งของวิชา วิศวกรรมซอฟต์แวร์ (Software Engineering) รหัสวิชา 01076256

ภาคเรียนที่ 1/2557

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สารบัญ

เรื่อง	หน้า
Kanban คืออะไร.....	1
KANBAN Process	2
CFD คืออะไร?.....	5
ข้อแตกต่างระหว่าง Scrum กับ Kanban.....	6

Kanban

Kanban แปลตรงตัว มาจากภาษา ญี่ปุ่น ว่า ป้าย แต่ในหลักของ Software Development แล้ว Kanban นั้นคือการที่เราสร้างตารางการทำงานขึ้นมานั่นเอง โดยจะเป็นการเน้นในเรื่องของการพัฒนา Software เป็นหลัก

Kanban มีวิธีการเรียงหลายวิธีขึ้นอยู่กับผู้ใช้ แต่จะแบ่งออกคร่าวๆ เป็น 5 ส่วน คือ Backlog / Start / In Progress / Validating / Done งานจะเริ่มเมื่อมีการส่งงานเข้าจาก Backlog และ จะไหลไปเรื่อยๆ จนถึง validated นอกจากนี้ในแต่ละหมวดนั้นต้องมีการกำหนดจำนวนงานสูงสุดที่จะอยู่ในส่วนนั้นได้ เช่น Start ได้ไม่เกิน 4 งาน In Progress ได้ไม่เกิน 3 งาน เป็นต้น การกำหนดมีไว้เพื่อให้การทำงานนั้นไม่มากเกินไป และ เมื่อมีงานที่เริ่มเร่ง ทีมที่ทำอยู่ก็สามารถทราบได้ทันทีว่า จะต้องเร่งงานส่วนไหนให้เสร็จเร็วขึ้น

Kanban คืออะไร

Kanban คือ วิธีการบริหารการเปลี่ยนแปลง โดยเน้นการจัดการงานให้อยู่บนหลักการตามข้างล่างนี้

- **Visualize Work** ทำให้เห็นแผนภาพของงานทุกๆขั้นตอน ในห่วงโซ่แห่งคุณค่า([Value Chain](#))จากแนวคิดที่คลุมเครือ ไปสู่การส่งมอบผลงานที่มีคุณค่า
- **Limit Work-In-Progress (WIP)** กำหนดขีดจำกัดที่ชัดเจนกับปริมาณของงานที่ได้รับอนุญาตทำได้ในแต่ละขั้นตอน
- **Make Policies Explicit** สร้างนโยบายที่ชัดเจน ที่คุณจะต้องปฏิบัติตาม
- **Measure and Manage Flow** การวัดผลและจัดการ การไหลของกระบวนการ นำไปสู่การใช้ข้อมูลเพื่อการตัดสินใจ และทำให้เห็นภาพผลลัพธ์ที่จะเกิดขึ้นตามมา
- **Identify Improvement Opportunities** สร้างวัฒนธรรมไคเซ็น([Kaizen](#))ให้เกิดขึ้น เพื่อการพัฒนางานให้ดีขึ้นเรื่อยๆขึ้นไป อย่างต่อเนื่องอยู่เสมอกับงานของทุกคน

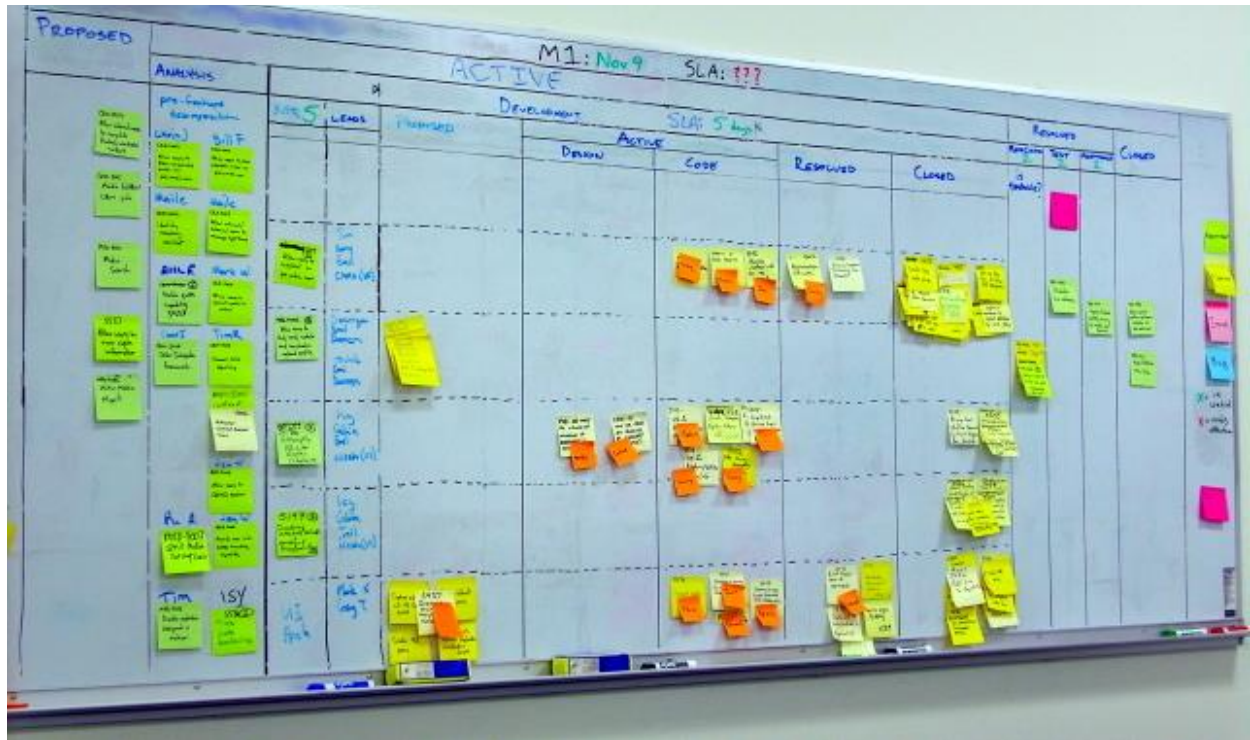
ข้างล่างนี้เป็นคำนิยาม Kanban จากผู้ริเริ่ม

*“...**Kanban** (capital K) is the evolutionary change method that utilizes a kanban (small k) pull system, visualization, and other tools to catalyze the introduction of Lean ideas into technology development and IT operations”*

David J. Anderson, Kanban 2010

KANBAN Process

Backlog (งานที่รอทำ) - นี่เป็นส่วนที่เรา List งานทั้งหมดที่ต้องทำเอาไว้ และเป็นจุดแรกในการเริ่มการทำงานโดยใช้ Kanban Backlog มีจุดเด่นอย่างหนึ่งคือ ไม่จำเป็นจะต้องมีจำนวนงานที่จำกัด แต่ควรมีการสื่อสารกับ Product Owner ตลอดเวลาว่า งานชิ้นใดที่อยู่ใน Product Backlog นั้นจำเป็นจะต้องทำจริงๆ เพื่อที่จะสามารถขยับเข้าสู่ขั้น Start ได้ ในส่วนนี้ทีมและ PO (Product Owner) สามารถที่จะเพิ่มจำนวนงานได้ตามที่เห็นชอบและเห็นควร



รูปร่างของกระดาน Kanban โดยคร่าวๆ

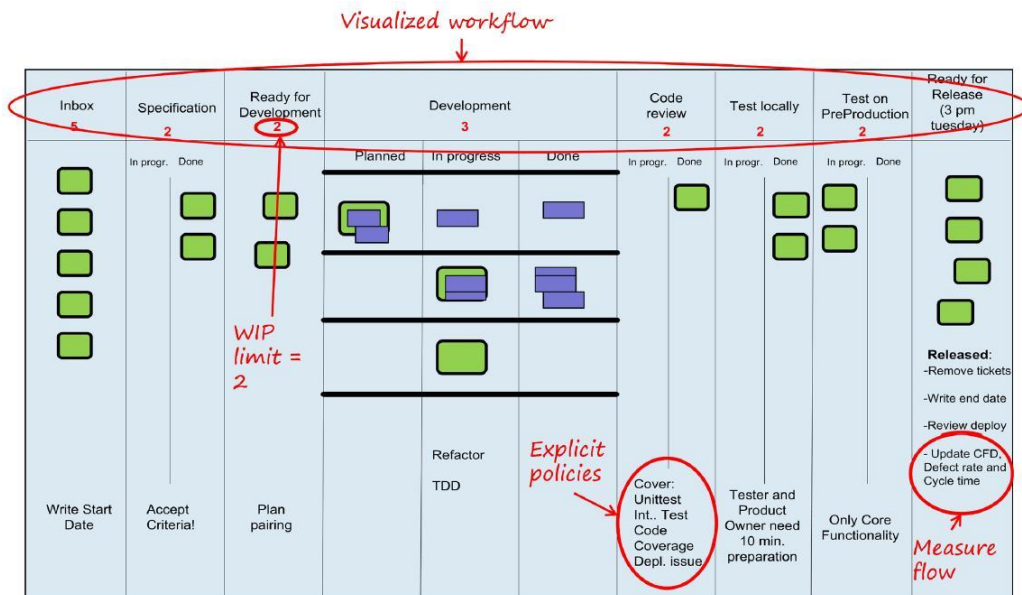
Start (งานที่รอเริ่ม) - หลังจากที่มีการเลือกงานเข้ามาอยู่ใน Start แล้ว ตรงนี้จะเป็นตัวบอกทีมที่กำลังทำงานอยู่ว่า งานใดเป็นงานที่ถูกเลือกให้ทำเป็นชิ้นต่อไป ทีมงานสามารถที่จะทำการเตรียมพร้อมทั้งในด้านทรัพยากร และ บุคคลที่จะเข้าร่วมทำงานเพื่อเริ่มที่จะทำงานในจุดนี้ การกำหนดส่วนนี้ควรอิงจากความสามารถในการส่งงานของทีมต่อสัปดาห์ เช่น Start ไม่เกิน 4 ชิ้นต่อสัปดาห์

In Progress (งานที่กำลังพัฒนา) - เมื่อทีมได้เลือกงานที่จะทำแล้ว งานนั้นก็จะเลื่อนเข้ามาอยู่ในช่อง In progress เพื่อเริ่มการทำงาน ในช่วงนี้ สามารถแบ่งเป็นช่วงย่อยๆ ได้อีก หากต้องการเช่น Design, Wireframe, Prototype, Full Mockup function เป็นต้น และควรมีการกำหนดจำนวนงานเช่นเดียวกัน

Validating (ทดสอบ) - เมื่องานสร้างเสร็จแล้ว งานนั้นๆก็จะเข้าสู่กระบวนการทดสอบ โดยจะทดสอบกันเองภายใน กับ Stakeholder / Product Owner ก็ได้ และจะต้องมีการ Review ผลการทดสอบกับ Stakeholder, Product Owner และ Development Team ด้วย

Done (เสร็จ) - เมื่องานทำเสร็จแล้ว ก็จะเข้ามาที่จุดนี้เพื่อเป็นการบอกว่างานนี้เสร็จแล้ว และได้ผ่านการทดสอบแล้ว จากนั้น ก็ขึ้นอยู่กับ Product Owner ว่าจะนำสิ่งที่ได้เรียนรู้ใหม่มาปรับปรุง แก้ไข เพื่อที่จะได้นำเข้าสู่ Product Backlog ใหม่อีกครั้ง

ตัวอย่างของ Kanban board ที่เป็นกระดานแสดงกระบวนการทำงานทั้งหมด ของการจัดการ projects ซอฟต์แวร์หนึ่ง โดยประยุกต์ใช้ Kanban



- คุณจะเห็นว่าบนกระดาน Kanban นั้น แสดงให้เห็นแผนภาพของกระบวนการพัฒนาซอฟต์แวร์ทั้งหมดของระบบ(Visualize Work)
- WIP limit ได้ถูกเขียนแสดงไว้ที่หัวกระดานในแต่ละคอลัมน์
- นโยบายถูกทำให้ชัดเจน(Policies Explicit) เพื่อให้เกิดคุณภาพของงานที่ยอมรับสำหรับ กระบวนการถัดไปดังต่อไปนี้ และที่คอลัมน์แรกมีการจัดลำดับของงาน(Prioritize) โดยงานที่อยู่บนสุดจะถูกดึงไปทำก่อน เพราะสำคัญสุด
- งานที่ไหลไปอยู่บนกระบวนการพัฒนาซอฟต์แวร์ใดๆในระบบ จะถูกคำนวณ(Measure flow) อยู่เสมอที่กระบวนการทำงานตรงคอลัมน์สุดท้าย
- ที่หัวของคอลัมน์สุดท้าย คุณจะเห็นว่า ไม่ได้กำหนดค่า WIP limit ไว้ แต่ได้เขียนเวลาจริงๆที่ทีมได้ตกลงร่วมกันแล้วไว้ว่าพร้อมส่งมอบผลงาน ทุกๆวันอังคาร เวลา บ่าย 3 โมง เป็นปกติเสมอ ซึ่งหมายความว่า ทุกๆชิ้นงานจะถูกทำให้เสร็จสิ้น และจะถูกส่งมอบในเวลา นั้น

Kanban คือเครื่องมือที่จะเข้ามาช่วยบริหารการเปลี่ยนแปลงของงานในกระบวนการพัฒนาซอฟต์แวร์เดิมของเรา ถ้านำ Kanban เข้ามาใช้ ก็จะต้องปรับกับกระบวนการพัฒนาซอฟต์แวร์ที่ทีมใช้อยู่แล้ว (เช่น XP, Scrum, Waterfall, UP) ให้เข้ากันกับหลักการของ Kanban หรือง่าย ๆ ก็คือทำกระบวนการพัฒนาซอฟต์แวร์ที่เราใช้อยู่ขณะนี้ ให้แสดงไปอยู่บนกระดาน Kanban ดังที่ตัวอย่างข้างต้นเท่านั้น โดยมีขั้นตอนดังนี้

1. Visualize your workflow: เริ่มต้นทำความเข้าใจการไหลของงานในกระบวนการพัฒนาซอฟต์แวร์ตั้งแต่ต้นจนถึงขั้นตอนส่งมอบซอฟต์แวร์ของคุณก่อน จากนั้นนำมาวาดเป็นแผนภาพการไหลของงานแล้วแบ่งคอลัมน์ บนกระดาน Kanban ออกเป็นจำนวนให้เท่ากับกระบวนการทำงานของคุณ แล้วนำชื่อของกระบวนการนั้นมาเขียนเป็น title บนกระดาน Kanban อีกที่เป็นอันเสร็จขั้นตอนนี้ ดังภาพตัวอย่างข้างล่างนี้

PO Inbox	Specification	Ready for development	Development			Code review	Test with PO and tester	Ready for release	Release
			Planned	In progress	Done				

Activity Stage

Buffer Stage

2. Limit Work in Progress (WIP): หลังจากได้กระดาน Kanban ที่มีคอลัมน์ของกระบวนการทำงานแล้ว ต่อจากนี้ก็ให้กำหนดข้อจำกัด (limit) ของ WIP ที่ยอมให้แต่ละกระบวนการดึงงาน (pull) เข้ามาทำได้ WIP ก็คือจำนวนงานที่กำลังทำอยู่ในระบบของเรา หรือจำนวนงานที่อยู่ระหว่างทำในแต่ละคอลัมน์บนกระดาน Kanban รวมกันนั่นเอง ดูตัวอย่างกำหนดขีดจำกัด WIP บนกระดาน Kanban ตามรูปตัวอย่างข้างล่างนี้ (รูปตัวอย่างนี้ ได้กำหนดขีดจำกัด WIP เป็นจำนวน slot ว่างไว้ในคอลัมน์เลย แทนที่จะเขียนไว้ที่หัวคอลัมน์เหมือนตัวอย่างกระดาน Kanban ภาพแรก)

Inbox	Specification	Breakdown	Development			Code review	Test locally	Test on PreProduction	Release
			Planned	In progress	Done				

WIP Limit = 2

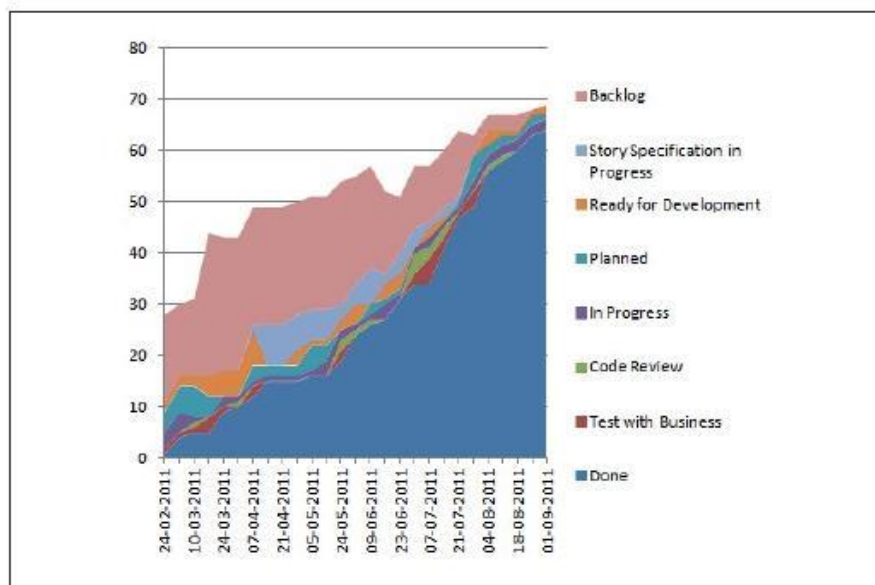
WIP Limit = 3

3. Make Policies Explicit: กำหนดนโยบาย QA และทำให้มันเห็นได้ชัดเจน เช่น ผลผลิตที่ออกมาในแต่ละคอลลัมน์ ที่จะเรียกว่าเสร็จแล้ว (Done) จะต้องทำอะไรบ้าง ขั้นตอนทำงานนี้กำหนดว่าต้องใช้เทคนิค TDD จะต้องเขียน unit test มี code coverage เท่าไรถึงยอมรับได้ ทำ code analysis ใหม่ มี document ใน code ก็เปอร์เซน ใช้ pattern อะไรออกแบบ แบบนี้เป็นต้น ดูภาพกระดาน Kanban ที่ได้เดินนโยบาย QA ลงไปแล้ว

[illegible]

4. Measure and Improve Flow: ที่นิยมใช้คำนวณเพื่อทำให้เห็นสุขภาพของกระบวนการทำงานทั้งระบบของทีมนั่นก็คือ Cumulative Flow Diagram(CFD) ซึ่งใช้แทน burn down charts ที่นิยมใช้ในกระบวนการแบบ Agile ทั่วๆไป Diagram นี้จะแสดงปริมาณงานสะสมของแต่ละกระบวนการ ในแต่ละวัน

ข้างล่างนี้เป็นตัวอย่าง CFD



CFD คืออะไร?

เมื่อไหร่ที่ระดับไปนพื้นที่สีน้ำเงินของเส้น Done จะแสดงให้เห็นถึง velocity ของงานที่เสร็จ หรือ ความเร็วของการผลิตในระบบนั่นเอง และความต่างของเส้นนี้ กับเส้น Backlog นั้น บางทีจะเรียกกันว่า WIP

- ถ้าความสูงที่เป็นส่วนหนึ่งใน WIP เพิ่มขึ้นแสดงว่าเกิด ปัญหาคอขวดขึ้นในระบบ
- ถ้าระดับพื้นที่ของเส้น Backlog ช้นกว่า ระดับพื้นที่ของเส้น Done แสดงว่าคุณได้ เพิ่มงานเข้าระบบมากเกินไป กว่าความสามารถทำงานได้ของระบบคุณ
- เราสามารถประมาณเวลาที่จะปิดโครงการนี้ง่ายๆ ได้โดยลากเส้นแนวโน้มบนเส้น Backlog และ Done ให้เส้นสองเส้นมาบรรจบกัน แล้วลากเส้นตั้งฉากตรงจุดตัดนั้นจนถึงแกนนอน จุดบนแกนนอนนั้น สามารถประมาณได้นั่นคือวันส่งงานครั้งสุดท้าย

ข้อแตกต่างระหว่าง Scrum กับ Kanban

Scrum	Kanban
fixed time boxes – มีการกำหนดช่วงเวลาใน iteration ที่แน่นอน และมีการกำหนด/แบ่งแยก planning meeting และ demo and review meeting เห็นชัดเจน	no fixed time boxes –ไม่มีการกำหนดช่วงเวลาในแต่ละ iteration ใช้วิธีการแบบ increment boundaries จึงไม่มี fixed planning meeting หรือ fixed demo and review meeting
tasks & estimates –ดูว่ามีงานอะไรบ้าง และ ประมาณเวลา ออกมาเพื่อดูสิ่งที่ต้องทำให้เสร็จภายใน เวลา ที่กำหนด	no tasks & estimates –หยิบงานมาเริ่มทำเลย
มีการ track velocity –เพื่อดูว่าในแต่ละ box เราควรจะได้ทำงานได้เยอะแค่ไหน	มีการ track flow –เพื่อดู queues (มีอะไรอยู่บ้าง), work in process(ทำอะไรอยู่) และ cycle time (ใช้เวลาทำนานเท่าไร)
scrum master เป็นเจ้าของ process –ทีมฟัง direction มาจาก scrum master เป็นหลัก	team เป็นเจ้าของ process –ไม่มีการกำหนด process ที่แน่นอน แต่ทีมมีการประเมินงานใน queue/WIP/cycle time แล้วค่อยดูว่าควรปรับปรุงและทำงานอย่างไรต่อไป

Kanban นี้เป็นเครื่องมือที่ยืดหยุ่น เข้าใจได้ไม่ยาก และชัดเจนดีมาก ๆ สามารถนำเอา Kanban เข้ามาประยุกต์ใช้กับกระบวนการพัฒนา ที่ทีมเป็นอยู่แล้วได้เลย แถมไม่ยาก เพราะมันจะไม่กระทบกระบวนการ และไม่กระทบกับบทบาทของทีมที่มีอยู่แล้ว

กระบวนการของ Agile จะกำหนดบทบาทใหม่ ๆ ขึ้นมาทำหน้าที่บางอย่าง เพื่อให้กระบวนการดำเนินไปได้ พอนำมาใช้ก็จะต้องฝึก หรือค้นหาบุคลากรเพื่อรับหน้าที่บทบาทนั้นๆเพิ่มเข้ามาในทีม ซึ่งมันก็เลยทำได้ยากที่จะเอาเข้ามาใช้ในทีมที่เป็นอยู่ในปัจจุบัน หากเราไม่ใช่ projects manager หรือเจ้าของบริษัทซะเอง

แต่ในกรณีที่เอา Kanban เข้ามาปรับใช้นี้ จะทำให้การทำงานง่ายกว่ากันเยอะเลย เพราะมันเป็นแค่เครื่องมือ ที่จะเข้ามาช่วยให้ทีมประสานงานกันผ่านกระดาน Kanban ที่ทำให้ทั้งทีมเห็นภาพขั้นตอนการทำงานได้ชัดเจนขึ้นเพื่อ จัดการวางแผนงานร่วมกัน รู้เข้าใจ และต้องปฏิบัติตามนโยบายของ QA ตลอดทุกๆ

กระบวนการพัฒนา สามารถประเมินผลงานได้ ซึ่งนั่นจะนำไปสู่การปรับปรุงกระบวนการทำงานอย่างต่อเนื่อง (continuous improvement) ได้ต่อไปเลยทีเดียว

และสุดท้าย อย่าลืมหลักการหนึ่ง ที่สำคัญมากๆ ของ Kanban เลย นั่นก็คือ ทุกคนในทีมจะต้องติดตั้งหัวใจแห่ง Kaizen ลงไป หรือไม่ก็ติดตั้งหัวใจแห่ง Agile ก็ได้เหมือนกัน เพราะทั้งสองหลักการนี้นั้น มีสิ่งที่คล้ายๆกันอยู่แล้ว

เพิ่มเติม:

- สำหรับซอฟต์แวร์ที่จะนำมาใช้เป็นกระดาน Kanban แนะนำ [kanbanery](#)
- หนังสือที่จะทำให้เข้าใจ Kanban ต้องอ่าน [Kanban: Successful Evolutionary Change for Your Technology Business](#)
- อ่านน้อยๆหน่อย แต่เข้าใจแล้ว Kanban มาลองใช้ได้เลย [Priming Kanban](#)