**Kanban**

Intro

Principles de base

3 règles du kanban

« In general, Kanban has three rules: (1) visualize the workflow,

(2) limit work in progress (WIP) at each workflow state,

and (3) measure the lead time (i.e. average time to complete

one item) [8]. » (Ikonen et al. 2010)

principles (ley 2017)

\_ Limiting WorkinProcess(WIP).

\_ Pulling valuethroughthedevelopmentprocess.

\_ Making thedevelopmentprocessvisible.

\_ Increasing throughput.

\_ Using a fixedbacklog.

\_ Embedding quality.

Kanban methodology focuses on having the right work done at the right time, given the skill sets of the developers.

Inaddition,projectdevelopersdonotimplement unnecessary features,donotwritemorespecifications thanthey can code,donotwritemorecodethantheycantest,anddonot test morecodethantheycandeploy.Therefore,Kanbanmetho- dology eliminateswasteineverystep,andissuitableforsoftware engineering projects [20].

“Themain focus ofKanbanistoaccuratelystatewhatworkneedstobedone, and whenitneedstobedone.Itdoessobyprioritizingtasks,and defining workflow aswellaslead-timetodelivery [20]. TheKan- ban processexplicitlypresentsthemostimportanttasksthatneed the mostattentioninordertoreducetheriskoftheirincomple- tion, andalsotoincrease flexibilityamongstothertasksinthe project. . Lei 2017



Exemple de kanban au niveau de la transparence Lei et al. 2017



**Reconnaissance des pertes / wastes**

« «In lean thinking, *waste* is basically everything that does

not add to the customer value of the products. In general,

three basic categories of waste-related elements can be identified:

(1) Muda: non-value-adding activities (NVA), (2)

Mura: variations (in process quality, cost, delivery), and (3)

Muri: unreasonableness (overburden). Consequently, one

of the most important principles of lean software development

is to learn to recognize waste. In software development,

such elements can be interpreted as follows [12]: partially

done work (inventory); extra processes (NVA); extra

features (overproduction); task switching, waiting, motion

(NVA); and defects. » (Ikonen et al., 2010)

Ex de pertes: code partiel, procédés excédentaires, extra features, changements de tâches/projets, attentes, bug découvert tard dans le développement, …

Fonctionnement

Avantage

« Advantages of the Kanban-driven operations

are that the inventories (simultaneous WIP) are reduced

and the overall production flow can be balanced easier.” (Ikonen et al. 2010)

“Another member commented that: *“Because the tasks*

*were partitioned into small pieces, excluding a couple of*

*exceptions, people mainly did not have to wait for the accomplishment*

*of other tasks.”*

Waiting is an inseparable part of projects. While some

waiting can be tolerated, damages for a project caused by

waiting must be eliminated to have more efficient progress.” Ikonen et al 2010

Inconvénients

-comme les tâches en cours sont limitées en nombre, si une tâche est plus longue que prévue, elle bloque tout le projet

« *“One task was*

*estimated to be ready within half an hour. However, it took*

*over two days, which blocked a couple of other tasks until*

*the delayed task was accomplished.” (Ikonen et al. 2010)*

étapes de revision/évaluation parfois superflues, changement de taches parfois nécessaire pour éviter les blocages (ikonen 2010)

Conclusion

« By revealing waste in projects, significant actionable opportunities

in terms of saving resources and accelerating

lead-time can be reached for practical use.”

“The study demonstrates that zero waste is not a requirement

for a successful project. » Ikonen et al. 2010

“Kanban methodology focuses on having the right work done at the right time, given the skill sets of the developers”. Ley 2017

While theresultsimplythatthereisnostatisticallysignificant difference atthe95%confidence levelbetweenKanbanandScrum for thefactors,theresultsdosuggestthatKanbanperformsbetter than Scrumintermsofmanagingprojectschedule(i.e.thesche- dule factor).ResultsalsosuggestthatprojectsusingtheKanban methodology canexperiencegreaterconsistencyintermsofthe projectmanagementfactors.Resultsalsosuggestthatoverall,both Scrum andKanbanleadtosuccessfulsoftwareprojects,whereon average,thesurveyrespondentsrespondedwiththe “Agree” re- sponse toquestionspertainingtothequalityfactor.Companies should beawareofthedifferencesinthepracticalimplementation of thesemethodologies,andchooseoneortheotherbasedonthe context,practicalneeds,andresourcesoftheproject. Ley 2017

References

Ikonen, M. & Kettunen, P. & Nilay, O. & Abrahamsson, P. in Proc. of SEAA Euromicro 2010 (IEEE).

**Exploring the Sources ofWaste in Kanban Software Development Projects 6 pages**

H. Lei et al. 2017 A statistical analysis of the effects of Scrum and Kanban on software development projects Robotics and Computer-Integrated Manufacturing 43 (2017) 59–67