

ENHANCING IT STUDENT PROJECTS WITH LANBAN

Case study: Lahti University of Applied
Sciences (Lahti UAS)

LAHTI UNIVERSITY OF APPLIED
SCIENCES
Degree programme in Business
Information Technology
Thesis
12.04.2016
Minh Son Nguyen

Lahti University of Applied Sciences
Degree Programme in Business Information Technology

NGUYEN, MINH SON

Enhancing IT-student projects with
Kanban

Case study: Lahti University of
Applied Sciences (Lahti UAS)

Bachelor's Thesis in Business Information Technology, 25 pages, 1 page
of appendix

Spring 2016

ABSTRACT

Group work is considered as the biggest challenge of project-based learning in colleges and universities. Lahti University of Applied Sciences (Lahti UAS) is not an exception. IT students at Lahti UAS are facing this challenge in their projects due to the fact that there is no available project management tool and practical methodologies for them to use. Lacking of the transparency of workflow and assessment in project work creates collaborative overload and uneven shares of workload for students.

The purpose of this study is to investigate feasible and promising Agile practices which can be applied in school projects. This thesis discusses IT students' project work and proposes a project management method and tool to improve project work among students at Lahti UAS.

A group of students doing a project and their lecturer was interviewed. All volunteered students and the lecturer agreed that Kanban does help to improve the performance of group work by visualizing workflow and creating a transparent assessment framework to measure each member's contribution to the project.

Keywords: project management, agile methodologies, Kanban

CONTENTS

1	INTRODUCTION	1
2	RESEARCH BACKGROUND	2
2.1	Key concepts	2
2.2	Motivation	2
2.3	Research questions, objectives and scope	2
2.4	Research methodology, framework and structure	3
3	BACKGROUND THEORY	5
3.1	Software development life cycle	5
3.2	Traditional methodology and agile methodologies in IT project	6
3.3	Collaboration and Kanban for students in digital era	8
3.4	Challenges to project-based learning of IT student	11
4	LANBAN THE TOOL – KANBAN FOR LAHTI UAS	12
5	CASE STUDY: IT-STUDENT PROJECT AT LAHTI UAS	15
5.1	Analysis of collected data	15
5.1.1	Before using Lanban	15
5.1.2	After using Lanban	16
5.2	Findings	19
6	DISCUSSION	22
6.1	Conclusion and limitations of the study	22
6.2	Future research	22
	REFERENCES	24
	APPENDIX	25

ABBREVIATIONS

Lahti UAS	Lahti University of Applied Sciences
BIT	Business Information Technology
IT	Information Technology
SDLC	Software Development Life Cycle

1 INTRODUCTION

In ICT professions, skills and experience can only be gained by doing projects or to be more accurate, by effectively doing projects. Since 1960s, there have been many methodologies that have been adopted widely in the information technology (IT) sector. Such methodologies help to organize work and manage a project in an effective way. These methodologies can be categorized into groups.

There is a group known as “Agile methodology” or Agile Software Development which has become more popular in the recent years. This is one of the reasons for students to understand and practice agile methodologies before they graduate and find a job. This study will firstly introduce the core concepts of agile methodologies. It will then focus mainly on Kanban which is useful and easy to use for students.

Additionally, working on projects at a university is also known as project-based learning. This type of learning brings some challenges to students. Group work is usually seen as the biggest challenge that students have to face when doing a project (Harmer and Stokes 2014, 21). One of the causes of the this challenge is the lack of an assessment framework to ensure equal contribution and workload from students which leads to resentment between students (Butler and Christofili 2014, according to Harmer and Stokes, 2014, 21)

Thus, by proposing Lanban (a code name for a Kanban board web-based application) to a group of volunteered students to visualize their project work, the study will determine whether the efficiency of project work can be enhanced. In addition, the thesis discusses how Lanban can provide transparency for supervisors (lecturers) who are assessing an individual’s contribution to a project.

2 RESEARCH BACKGROUND

2.1 Key concepts

Agile Alliance (2015) defines Agile Software Development as a set of methods and practices for software development based on the values and principles expressed in the Agile Manifesto.

Kanban in IT is an agile software development method that derives from a similar-named technique used by Toyota line-workers to signal steps in the manufacturing process (Hefley 2014, 5)

2.2 Motivation

The motivation of this study is based on two key points:

- Agile methodologies are only officially taught at Lahti UAS during the final year which is not enough for students to get familiar with project management methods and tools. Students usually use the enhanced water-fall methodology which is less efficient than agile methodologies.
- There is no available project management tool for Lahti UAS's students at the moment. Other project management tools used widely in IT companies are usually not free.

2.3 Research questions, objectives and scope

In this study, the objectives are to observe and determine the efficiency of a visualization tool for IT project management. The author hopes that, as a result of this thesis, students at Lahti UAS could start using Kanban and Lanban in their projects. Therefore, there is one main research question: ***“How can Lanban improve students' project work at Lahti UAS?”***

This study will be carried out with the help of a group of volunteered students who will use Lanban in their project.

2.4 Research methodology, framework and structure

As a key part of this study, a group of students participating in a project course is asked to apply particular project management methodologies and a specific tool during their project. Before starting the project, the students and their supervisors are interviewed about their thoughts and ideas of project work.

The study is based on three hypotheses:

- Hypothesis#1: Visualizing workload enhances project work performance.
- Hypothesis#2: Students are more committed to work when they know their share of workload.
- Hypothesis#3: Lanban creates transparency in evaluation for supervisors (lecturers) to examine individual contribution to the project.

After the project is finished, additional interviews will be carried out with the same respondents. The project outcome is evaluated by the supervisors. Depending on the interviews and the project outcome, the above hypotheses will either be accepted or rejected.

In addition to the qualitative method, the interviews, a quantitative method is also used to provide data to answer the research question. Because quantitative method is a complementary method in this research, the data is collected randomly from BIT (Business Information Technology) students from three groups at Lahti UAS: BIT11, BIT12, and BIT13. Moreover, it should be noted that the selected students come from both Asian countries and European countries which ensures the validity and objectivity of data sampling.

The students are asked with three questions:

1. Which communication tool do you usually use for project work?
2. Do you like online work or face-to-face work?

3. How do you usually divide project work? For example: Do you allow team members to choose their role (e.g. a programmer, a tester, a designer etc.), or does each team member choose a specific feature they deliver and do all development work from designing to coding and testing?

3 BACKGROUND THEORY

3.1 Software development life cycle

Software development life cycle (SDLC) also known as software development process, is a structured sequence of stages in software development (Tutorials Point, Software Development Life Cycle).

SDLC contains a specific plan addressing maintenance, replacement or enhancement. The typical stages of SDLC can be illustrated in the graph below.

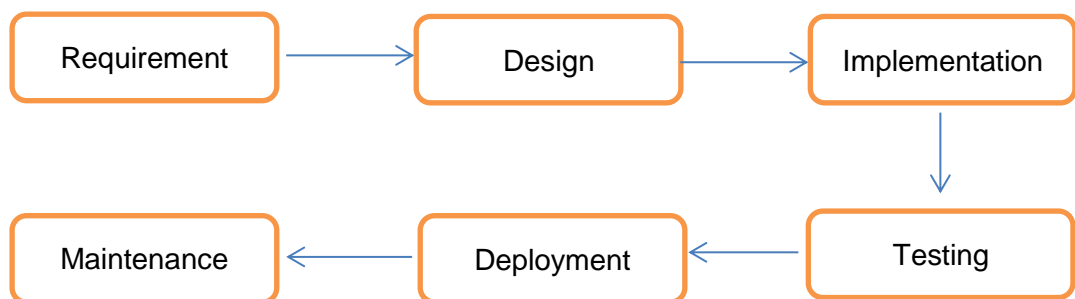


Figure 1 Software development life cycle stages

It should be noted that there are various SDLC models with different characteristics and use-cases that are based on stages above. Tutorials Point (SDLC Overview, 2016) mentions some of the most important and popular SDLC models in IT: waterfall model, iterative model, spiral, v-model, agile model, rapid application development model and prototyping model.

This thesis focuses on the waterfall model and agile model. The term “waterfall model” is used interchangeably with the term “traditional methodology” due to the fact that it is the first and the earliest SDLC model for software development. Likewise, the agile model is known as “agile methodology”.

3.2 Traditional methodology and agile methodologies in IT project

Based on SLDC Overview (Tutorials Point) and Agile discussion guide (LeanDog, 2012) and Agile unleashed (Cooke, 2010), the author summarizes the comparison between traditional methodology and agile methodologies in the table below.

Table 1 Comparison between traditional method and agile methods

Traditional method – Waterfall	Agile methods
<p>Developed in the 1960s, the first software development approach</p> <p>Based on physical process models</p> <p>IS development is “process flow” so it is hard or unable to halt or go backward</p>	<p>Not trying to accomplish everything at once but slicing the big goal into little iterations</p> <p>Agile methods drive for reasonable ratio between documentation design and implementation</p> <p>Focus on customer satisfaction and a system that works</p>
Advantages	
<p>A clear and ideal model – giving information on what tasks are needed in an IT project</p> <p>Good for relatively small projects</p>	<p>Easier to control the process of work</p> <p>Easier for customers to see an actual outcome which means offering possibilities to customers to evaluate each improvement</p>

	High value and quality by giving possibility to customers to alter the solution even near the end of the project
Disadvantages	
Customers cannot see the final product before the second half of the project	Not suitable for handling complex dependencies in complex project.
Changes in requirements are difficult	More risk of maintainability and extensibility due to lacking of detailed planning
Does not give flexibility at the end of the project	Heavily depends on the customers; thus the team can be out of track if the customers do not have clear requirements.
Toward the end of the project the number of defects increase	
Aims	
Detailed planning and complete forecasts of the tasks during the product life cycle	Verify the customers' wishes really early in the project
Design and development is emphasized in the 1 st half the project. Testing is emphasized in the 2 nd half of the project	Put focus on delivering fully working features and keep the current version as ready as possible
	Drive for continuous integration and

	development and tackling defects early
--	---

According to Agile Alliance (2015) and Cooke (2010, 117), agile methodologies value:

- ✓ **Individuals and interactions** over processes and tools
- ✓ **Working software** over comprehensive documentation
- ✓ **Customer collaboration** over contract negotiation
- ✓ **Responding to change** over following a plan

A list of the twelve principles of the Agile Manifesto can be found in Appendix 1.

3.3 Collaboration and Kanban for students in digital era

“Collaboration is working with others to achieve shared and explicit goals” (Laudon & Laudon 2014, 88). Collaboration is the main factor of agile methodology which is implied by the Agile Manifesto (see Appendix 1) and a collaboration tool in IT also is a part of project management tool.

It is necessary to remember that Kanban is an agile methodology for project management. However, the main characteristic of Kanban is a visual board, which is used to “visualize workflow and picture of the product in each stage – from concept to deployment” (LeanDog, 2012), is a tool. Because of this characteristic, Kanban can be easily integrated with other agile methodologies, for example with Scrum (another agile methodology), to create a new methodology named ScrumBan.

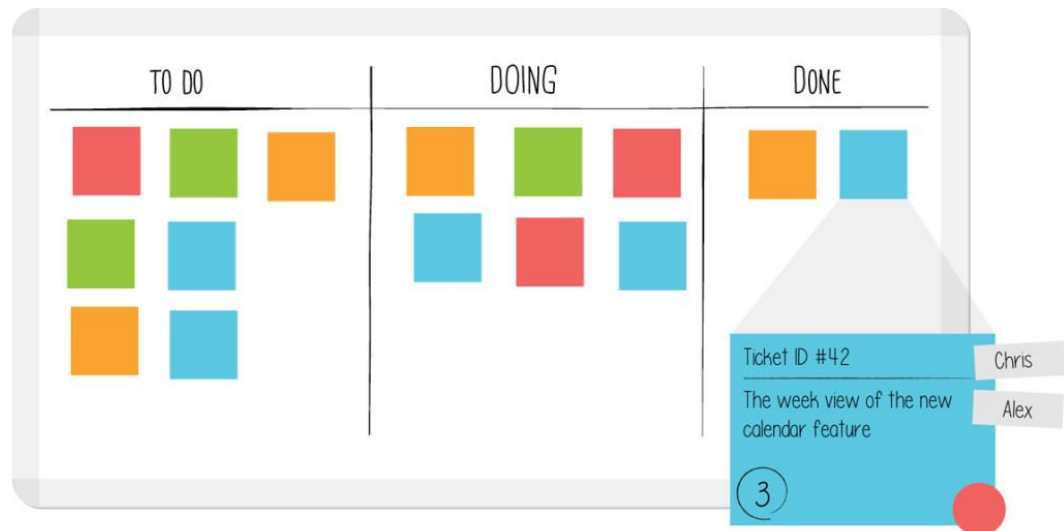


Figure 2 Kanban visual board (Source: LeanKit - <http://leankit.com/learn/kanban/kanban-board/>)

Kanban also helps to limit work in progress (WIP). This helps to prevent collaborative overload and manage work flow to allow and prepare for changes during iteration. Moreover, Kanban helps in making team policies transparent so that each team member can follow these policies. The project, then, has smoother iteration (LeanDog, 2012). The minified version of Kanban board can be as in Figure 2.

However, visual boards such as the one in Figure 2 are missing features and do not therefore meet the requirements for collaboration in the digital era. Based on the evaluation checklist for collaboration tools (Laudon & Laudon, 2014), those features are:

- Profiles: describe who the user is
- Content sharing: share, store and manage content including documents, presentations and other files.
- Feed and notifications: real-time information streams, status updates and announcements
- Tagging and social bookmarking
- Permissions and privacy

These missing features lead to the birth of Lanban which is described in more detailed in the next chapter. Although the development of Lanban was stopped at the prototype phase, it implemented content sharing, feed

and notifications feature; profiles, permissions and privacy features were partially developed; tagging and social bookmarking is what need to be done in the future.

Another tool which can help to evaluate collaboration tools is the time/space collaboration matrix (Laudon & Laudon 2014, 97). This matrix addresses collaboration problem: time and space. From a student's perspective, a space-related problem may arise, for instance, if a group of students can only spend a limited amount of time in a specific classroom because the room is then reserved for other students. Moreover, it is not certain that the room is available at all. Secondly, time can be a problem because students' timetables may differ, and it is therefore difficult to find common time for group work.

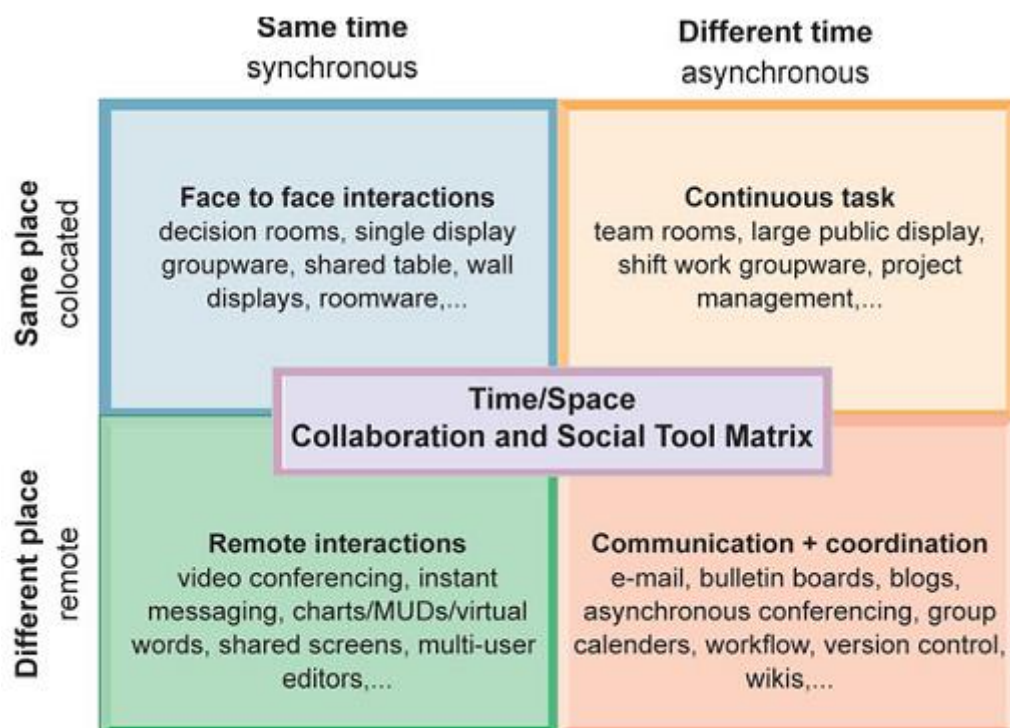


Figure 3 The time/space collaboration and social tool matrix (Laudon & Laudon, 2014)

Based on the time/space matrix, the collaboration tool for student needs to support group work in different time and different place. That is the reason why Lanban – Kanban – is a real-time web-based application.

3.4 Challenges to project-based learning of IT student

According to Harmer and Stokes (2014, 21), the project-based learning approach hinges on a philosophy that learning is effective when students apply theory into practice. The student role changes from “learning by listening to learning by doing” (Stauffacher, Walter, Lang, Wiek, & Scholz 2006, 255).

However, students often face challenges in project-based learning. Harmer and Stokes (2014) mention that group work have been usually seen as the biggest challenge that students have to face when doing a project. Stauffacher et al. (2006) also discovered that group work is the major source of other challenges. One of the causes of this challenge is the lack of an assessment framework to ensure equal contribution and workload from students which leads to the resentment between students (Butler and Christofili 2014, according to Harmer and Stokes, 2014).

In addition, another concern regarding project-based learning is that lecturers use their own criteria for assessing project outcomes. As a result, the question about transparency and equity of this assessment framework emerged (Van den Berg et al., 2006 according to Harmer and Stokes, 2014). Blumenfeld et al. (1991) suggested that the lack of assessment clarity creates uncertainty for students (Harmer & Stokes, 2014)

The following chapter discusses one of the main features of Lanban, graphical charts, which can help to create transparency to examine an individual’s contribution to a project. It also means that Lanban can mitigate the challenges of project-based learning.

4 LANBAN THE TOOL – KANBAN FOR LAHTI UAS

In the autumn semester 2014, under the supervision of two senior lecturers, Heikki Paananen and Antti Salopuro, an ultra-light project management Kanban tool was created as an output of web application development project. The outcome was published in GitHub (2014) as an open-source application. It was named Lanban, the code can be found at <https://github.com/nguymin4/Lanban>

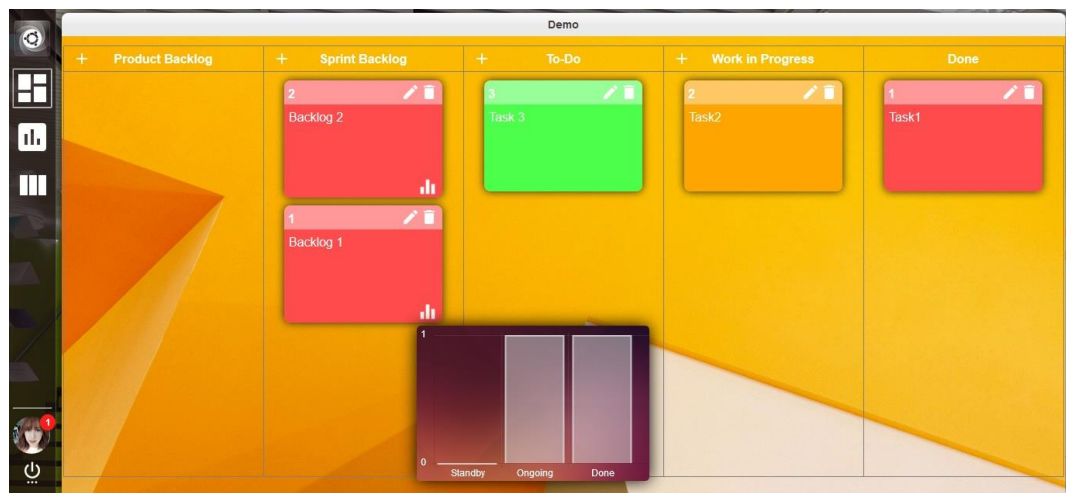


Figure 4 The main window of Lanban and the real-time notification icon in the lower left corner

This web application is a visual management tool which was developed as open-source software (OSS – License CC BY-NC) and dedicated to Lahti UAS. It can be used to control the flow of teamwork. With the help of this tool, a team is able to see who is working on which task. Thus, the tool enables better visual communication over team’s overall work in a time box. Furthermore, the tool allows users see how their work is progressing over time and illustrates the received business valued created in a project.

Value of the application: Project teams at Lahti UAS can use agile methodology with a lean approach. Electronic Kanban board would bring visibility to not only current work at team level but also at individual level: who is doing what and how much “points” have been earned within a

sprint. Team leaders or “scrum masters” could see directly how efficiently the team is working and where to find the obstacles or barriers for working. The board also works as a basis for everyday stand-up meeting content (Show-and-tell). This also means that everyone in the team can see the contribution of each team member. In other words, work becomes transparent.

Main Features

1) Working Kanban board (default 5 swim-lanes). It is possible to add custom swim-lanes and modify them.

2) Graphical charts and tracking data to sprint

- Illustrate tasks done by persons in a pie chart
- Illustrate points earned by persons
- Illustrate team estimation factor
- Illustrate a burn up data chart (Business value cumulated)

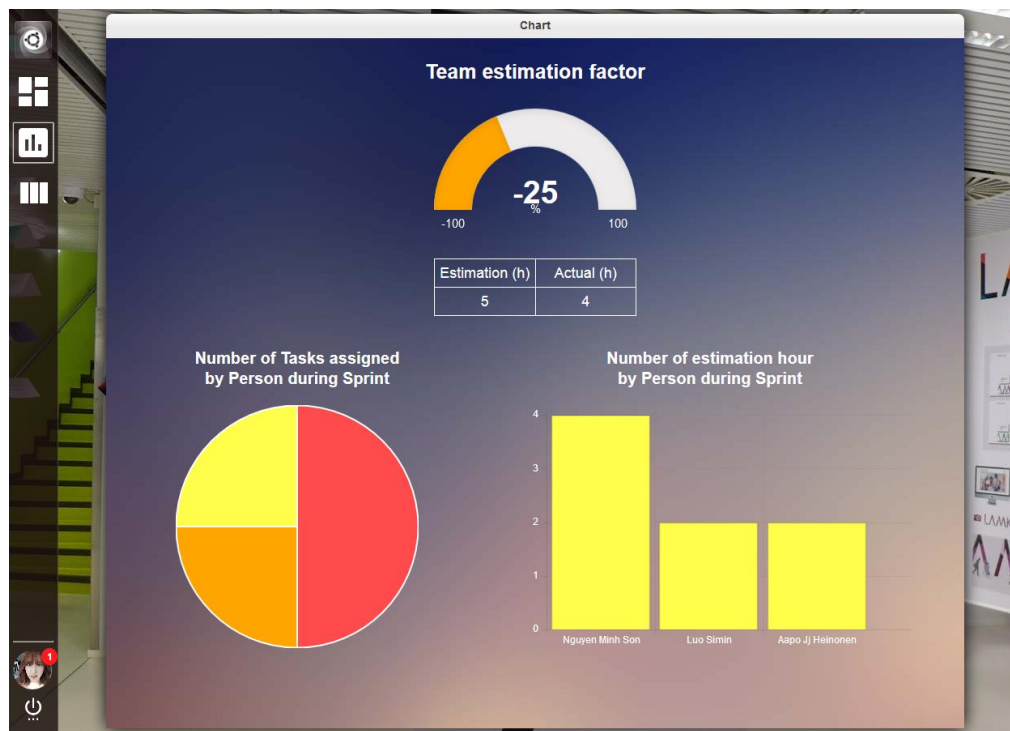


Figure 5 Graphical charts and tracking data to sprint

3) Attaching documents and comments to tasks in Kanban board

- Multiple uploading files with various file types
- Real-time chatting in the comments section

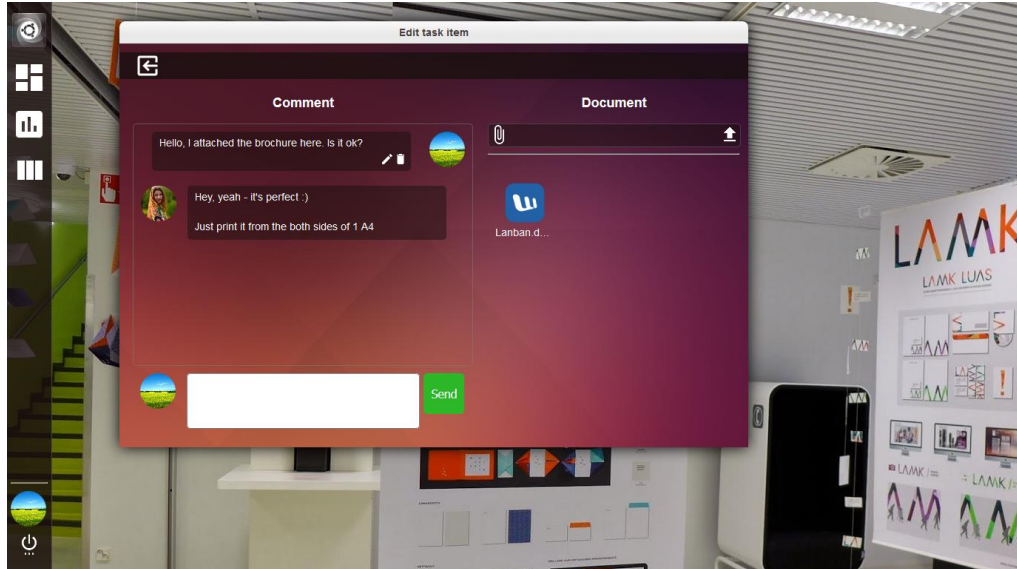


Figure 6 Comments and documents for a task

4) Supports multiple projects with multiple users

5) Notifications: Real-time status updates and announcements

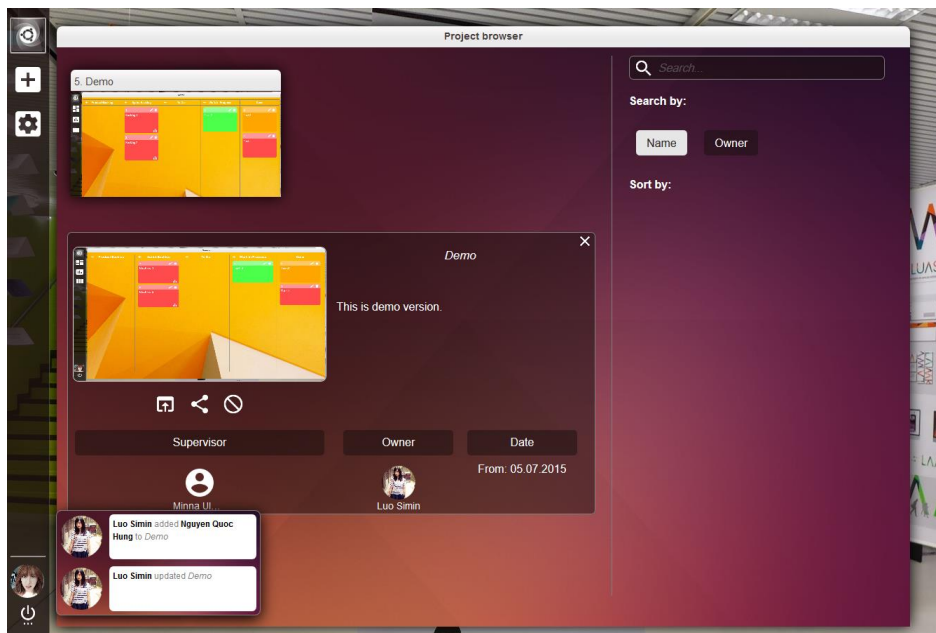


Figure 7 Multiple projects management and an opened notification center

5 CASE STUDY: IT STUDENT PROJECT AT LAHTI UAS

5.1 Analysis of collected data

5.1.1 Before using Lanban

To gather data, IT students with different national and cultural backgrounds from different study groups were asked three questions. The following presents an analysis of their answers.

a. Which communication tool do you usually use at the moment for project work?

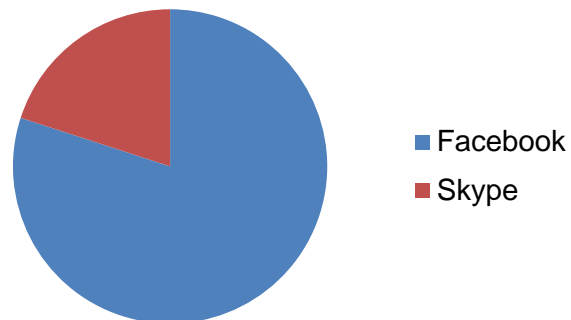


Figure 8 Current communication tools in project

It can be easily seen that Facebook is the main tool used for communication.

b. Do you like online work or face-to-face work?

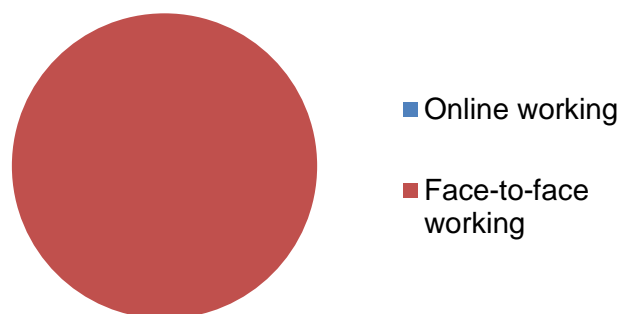


Figure 9 Preferred working style: Online working and Face-to-face working

Every respondent prefers face-to-face working to online working.

c. How do you usually divide project work? For example:

First way: Do you allow team members to choose their role (e.g. a programmer, a tester, a designer etc.)

Second way: Does each team member choose a specific feature they deliver and do all development work from designing to coding and testing?

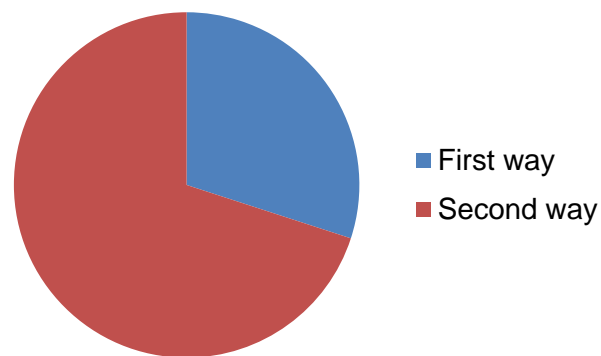


Figure 10 Dividing project work based on: member roles vs features of the project

Most of students prefer dividing project work based on features of the project they are delivering.

5.1.2 After using Lanban

a. Opinions and ideas about the tool from students

This is a summary of the feedback from students who used Lanban in their project.

Table 2 Advantages and disadvantages of Lanban based on students' feedback

Advantages	Disadvantages
<p>Friendly user interface (UI)</p> <p>Nice useful features: charts and graphs, notifications</p> <p>Support multiple projects with multiple users</p>	<p>Missing integrating to GitHub (software version control)</p> <p>Notification features are not too visible</p> <p>Limited number of character when writing description of tasks</p>

Question: By using Lanban, are members more committed to work when knowing their own share of workload?

Answer: “During the whole process of conducting the project, team members were divided the tasks fairly and with the high self-discipline, they were trying to finish the assigned task within the given amount of time and with the help of the Lanban, they know his or her share of workload better, so more or less, this tool partially helps them knowing his or her contribution for the whole project, leads to the their commit to work.”

Question: Does it improve project performance and what features of Lanban are useful for you?

Answer: All students agreed that Lanban does improve project performance of their project. The list below contains all the Lanban features which are most favored by students:

- Swim-lanes and visual board
It is now easy to check active tasks given to members. “No work overload anymore”.
- Burndown data chart illustrates remaining hours in the schedule.
This feature mitigates the risk of running out of time during a project.

- Task-done chart shows the number of tasks accomplished by team members

Each team member can see his/her tasks and other members' tasks, and the work can therefore be divided more evenly.

- Point chart illustrates contribution points per members.

Number of tasks is not the only factor should be accounted for. There are difficult tasks which requires big amount of time to complete as well as easy tasks which can be done very quickly. To measure a member contribution, it is a must to take both number of tasks and tasks difficulty into consideration.

One student gave a more insightful comment about Lanban:

"I am not really good in programming comparing to other students. So, I used to have harsh arguments with other team members about dividing tasks. With Lanban, the problem is gone. We together decide the difficulty of a task first by giving it a point from 1 – 10. Then each member chooses tasks based on the point. If you choose a hard task then you just need to pick one or two task like that. It's really convenient."

b. Reviews and comments from supervisors:

The whole interview with supervisors focused on three main points.

In general, Kanban can help a teacher to see a "slice-through" to the current progress inside a project or bigger task. Kanban helps in two ways:

a) helping a team to break down the topic/task, and b) visualize the progress. Breakdown can illustrate complex project topic as a bunch of smaller tasks. These tasks as a whole are equal to whole topic. By spending time in breakdown team members are able to understand the topic better, thus, enabling better estimating and preparation. Second part, visualizing the progress, it helps radically the teacher to spot groups in vein or problem. If a project progress is not happening, teacher can go to

talk to the students and keep a follow-up / support session. Also students can see that what sub tasks are done and what is still left to do. This helps seeing the big picture. Seeing the whole project in board will push the team to progress every week, every day.

There are couples of points worth understanding: a) willingness in the team to do proper breakdown and b) obeying the rules of Kanban flow. Teacher should be closely supporting the teams to follow the flow. This can be done through daily/weekly standup meetings. By using Kanban a team can really put the essential to the table; through visualizing the tasks everyone can see how work is progressing and value can be delivered. Moreover, light weight tool Kanban is fun and easy to learn and use but difficult to master.

For evaluating, Lanban can provide the usage of structured data. Each action done for tickets can be used to see how have done what. Metrics help to see how students have participated to the projects. By attaching values / work amount estimations, it is possible to see generated value per student. This requires teacher involvement in the planning/estimations session e.g. a review meeting at the beginning of project after the planning part.

5.2 Findings

a. Hypothesis#1: Visualizing workload enhances project work performance.

Based on chapter 3, it can be noted that Kanban helps to enhance group work by supporting better workflow management. The features that support workflow management are visualizing card between Kanban swim-lanes and limiting working in progress to mitigate the work overload issue of students.

In this chapter, after summarizing and analyzing interview with students and supervisors, there are several key points from practice of Kanban and Lanban which help to improve project performance. From students' point of view, features - visual board and chart - are the most favorable of students and help them to organize, manage and monitor their project work. Those features reduce students' work overload because the students can track active tasks in their project so that they can allocate their resources (time resource, human resource) to meet the schedule. On the other hand, from supervisors' point of view, visualizing the progress helps the lecturers to spot the groups' problems and then support them if necessary. In addition, with the help of visualizing the workload, students can see the big picture of the whole project and have a situational awareness of the current progress. It is essential and directly affects the outcome of the project.

Therefore, it can be concluded that hypothesis#1 is accepted.

b. Hypothesis#2: Students are more committed to work when they know their share of workload.

In section 3.4, one challenge of project-based learning which students have to face is group work. Unequal contribution and an unevenly divided workload lead to resentment between the students (Butler and Christofili 2014, according to Harmer and Stokes, 2014) and thus degrade the outcome of the project.

When the students were interviewed and asked what features of Lanban they consider helpful, they mentioned contribution issue. Breaking project features into small tasks does not ensure that every task has the same difficulty level. The hard tasks will require more time to complete. Importantly, the level of difficulty can be various which depend on the skills of the member(s) who do the task. A skillful member takes less time to complete a task than a member who is lack of necessary skills. Task-done chart and contribution-point chart help to solve the problem of contribution

based on the answer of students. Based on student interview, the team together decide the difficulty of a task first by giving it a point from 1 – 10 and then each member chooses tasks based on the point. The contribution point will be illustrated on contribution point chart and number of task done by a member will be displayed on task-done bar chart. Every member in the team knows his/her own contribution point by looking at the charts. The supervisor base on that to evaluate each students thus make students put more effort and be more committed to the project.

To conclude, hypothesis#2 is accepted.

c. Hypothesis#3: Lanban creates transparency in evaluation for supervisors (lecturers) to examine individual contribution to the project.

Project-based learning creates a concern for students that lecturers use their own criteria for assessing project outcome. The transparency and equity of this assessment framework are questioned by students (Van den Berg et al., 2006 according to Harmer and Stokes, 2014).

In the proof of hypothesis#2, it can be seen that with task-done chart and contribution point chart, the team and supervisor have a clear view of the contribution of each team member during the project. This is the base of a transparent framework to evaluate students that the thesis is addressing.

Furthermore, from the interview with supervisor (lecturer), one point needs to be emphasized is Lanban can provide the usage of structured data and metrics which help to indicate how active and committed a student is when working on the project. By attaching values / work amount estimations, it is possible to see generated value per student. Therefore, supervisors (lecturers) can evaluate a student's contribution objectively without the concern of students about the arbitrary and bias assessment.

Hypothesis#3 is also accepted.

6 DISCUSSION

6.1 Conclusion and limitations of the study

All the hypotheses are accepted. The interviewed students and supervisors (lecturers) agree that Kanban in general and especially Lanban do help improve the project performance at Lahti UAS. However, the study has some limitations.

The biggest limitation of the study is the small sample size. Two groups with 10 members used Lanban in their projects. This leads to the situation that the interviews with the students lacked diversity of opinion, feedback and ideas about Lanban.

Secondly, all students who used Lanban in their projects come from Vietnam so the communication in the team is already better than in a team with students from different cultures. Thus project performance is supposed to be better than in the case when the team has members come from different cultures.

As a prototype, Lanban itself contains several bugs. They are minor bugs based on students' reports. Additionally, Lanban also misses some features which can greatly support the project namely, GitHub integration.

6.2 Future research

The evaluation checklist for collaboration tools (Laudon & Laudon, 2014) was used to evaluate the Lanban itself. Despite the fact that Lanban has content sharing feature, built-in communication tool (chat) and feed and notification, these features were not highly favored by students, they still use for example, Facebook chat and group page as a main means of communicating, sharing files and getting notification. Based on author's

opinion, this can be explained that students spend most of their free time on Facebook e.g. chatting with their friends, following interesting news from community. Therefore, they find Facebook is convenient for team work because they can “work and play” with same tool at the same time. This topic need to be further studied.

About Lanban, for IT students, one of the desired features which need to be developed is the integration with a version control system, typically GitHub. The feed and notifications should be refactored with new functionalities and better user experience (UX). The source code is available on GitHub (<https://github.com/nguymin4/Lanban>) developers or students are welcomed to join and make it better based on the motto of open-source movement.

REFERENCES

Agile Alliance. 2015. What is Agile Software Development. *Agile Alliance Website*. [Online] 2015. [Cited: February 24, 2016.]

<https://www.agilealliance.org/agile101/what-is-agile/>.

Cooke, Jamie Lynn. 2010. *Agile Principles Unleashed*. IT Governance Publishing, 2010. pp. 27-130

Harmer, Nichola and Stokes, Alison. 2014. *The benefits and challenges of project-based learning*. Plymouth : Pedagogic Research Institute and Observatory (PedRIO) - Plymouth University, 2014. pp. 21-23

Hefley, Chris. 2014. *Kanban Roadmap*. LeanKit, 2014.

Laudon, Kenneth C. and Laudon, Jane Price. 2014. *Management Information System*. Pearson Education Limited, 2014. pp. 88-98

LeanDog. 2012. *Agile Discussion Guide*. LeanDog, Inc., 2012.

Stauffacher, Michael; Walter, Alexander I.; Lang, D.J; Wiek, A and Scholz, R.W 2006. Learning to research environmental problems from a functional socio-cultural constructivism perspective: the transdisciplinary case study approach. *International Journal of Sustainability in Higher Education* 2006, Vol. 7, 3. pp. 252-275

Tutorials Point. SDLC - Overview. *Tutorials Point Website*. [Online] [Cited: 27 2 2016.] http://www.tutorialspoint.com/sdlc/sdlc_overview.htm.

— . Software Development Life Cycle. *Tutorials Point Website*. [Online] [Cited: 27 February 2016.] http://www.tutorialspoint.com/software_engineering/software_development_life_cycle.htmhttp://www.tutorialspoint.com/software_engineering/software_development_life_cycle.htm.

APPENDIX

Appendix 1: The twelve principles of the Agile Manifesto

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.