Kanban Teaching Examples

A Master's Project in Software Engineering

By

Karen Remick

For the

University of Alaska – Fairbanks

Computer Science department

April 20, 2017

Graduate committee:
Dr Genetti (chair)
Dr Lawler
Dr Chappell

Introduction

Computers entered the every-day life of the man-on-the-street in the late 1970s. Since then the use and capability of computers has exploded. We now casually carry around more computing power in our pockets than was used to land a man on the moon. With that proliferation of hardware has come the requirement for software to run it.

When we first started developing software, we followed the same practices as we'd used for developing hardware: Gather the requirements once, design it and build it (a.k.a. the waterfall method). This soon proved to be unworkable as the requirements changed faster than the software could be developed. As a result of pressures to develop software to fit these changing requirements, the methods used to develop software became more agile, sacrificing extensive planning for the ability to incorporate new requirements as they arose.

Software developers are problem solvers, and determining the best way to handle rapidly changing requirements is just another type of problem for them to solve. Since there are tens of thousands of developers, we end up with a multitude of solutions. One common thread has become obvious from all of these solutions: agile software development produces the best results as the requirements change on timescales less than the development period. The faster the changes, the more agile the development process has to be.

One of the most agile methods currently used is Kanban. Kanban was initially developed by Toyota to mimic the inventory control used by grocery stores, in order to make sure that they weren't keeping any more stock on hand than necessary. Instead of using the system to track inventory, software developers use it to track work in progress. The basic idea is that limiting the work in progress reduces the chaos in the system and smooths out the workflow, making it more efficient.

Kanban is an extreme when it comes to reduction of planning. The first step in this reduction is to reduce the project requirements to the minimum it takes to have a working program (the minimum viable product). Following that, we further reduce the planning by only addressing a couple of the features at a time. This reduces chaos and smooths the workflow because the developer is focused on only one or two things, and doesn't have the disruption of changing what they are doing frequently, as happens when multi-tasking is required.

The result of this reduction in planning is that small bits of work get done quickly. Thus, changing requirements have very little impact on the work in progress, as changes usually effect things that haven't been addressed yet. Therefore, there is no previous work that needs to be thrown out. When changes do affect work that's been done, only the relevant part of the code needs to be changed, rather than having to have a whole section re-planned and reworked.

Because of these advantages, Kanban has become a popular development method. Unfortunately, there are very few resources available to learn Kanban from. It's more passed by word of mouth than by any standardized teaching methods. This project seeks to address this issue.

The Problem

While there are people, both professors and developers, willing and able to teach Kanban as a software development method, teaching doesn’t happen in a vacuum. Materials and concrete examples are needed to show how the process works. While there are several websites that will explain Kanban, there are no websites that supply this necessary learning material. The vast majority of videos touted as tutorials cover the basic ideas in a couple minutes (81,300 Kanban videos that last less than 4 minutes on Youtube as of 4/20/2017), but don’t go into any detail or show the project management capabilities of Kanban. There are also longer, more detailed videos that last more than 20 minutes, but they tend to focus on more technical aspects such as Kanban vs. Scrum, or incorporating Excel into Kanban, rather than focusing on Kanban.

The lack of good teaching materials is due to two primary factors: first, software developers who use Kanban are too busy actually developing software to create learning tools. Developing teaching materials is not part of their job description. Second, even if they are willing to spend personal time making examples, most of the software developed today is proprietary, so they can’t show actual examples. A final consideration is that developers don’t necessarily have any teaching experience or skill, and may not know how to present information to learners.

Another possible source for teaching materials is for them to be created by computer science professors. However, they have similar time constraints to the software developers, and unlike the software developers, they don’t have ready access to a development team working on a project, so lack the details to put into the example.

The Project

The solution is to find someone who has the time and teaching experience, and can research enough to understand how Kanban works: a graduate student. Since I have experience both in teaching and developing educational presentations , I only needed to research Kanban and think about how to apply the ideas behind Kanban in various situations.

Dr Genetti and I discussed what was needed, and hammered out the basic examples to use: Fast food to demonstrate the method, and a student registration program for the specific example. By picking a software project, but not actually developing it, we could give the needed specific examples.

I determined that there should be four videos; the demonstration video at layman level and the other three at a higher, more professional level. Here are the videos I created:

1) Kanban: A Fast Food Example

One way to teach someone a new method is to show it in terms of a process they already know. This video does just that; it takes the well understood process of delivering fast food and puts it in Kanban terms. The video shows the development of the Kanban board based on what will be needed for the process, and then work through its use, both for a single order, and for multiple orders all happening at once. It also addresses problems like running out of fries and fine tuning of the work in progress limits. This video is produced at layman level so it can be used in a survey class or to help someone get an over-all understanding.

2) Preparing to Kanban

When we think of software development, coding comes to mind, however there's a lot of work that needs to be done before the coding starts. This video addresses how we need to handle a project where we'll be using Kanban. The video has a fictional development team and university customer. The customer has a wish list and narrows it down to the minimum viable product (MVP). The Kanban board is developed including how to find the starting work in progress limits and done conditions. We determine priorities and dependencies among the MVP features and determine which features should be developed first.

3) Running Kanban

This video picks up exactly where the previous one left off and shows the fictional development team working through the development of the software. Specific functions are broken down into their programing tasks. These programming tasks are developed, tested and implemented into the system. Common problems, such as changing requirements and unavailable information, occur and the video shows how they are addressed when using Kanban.

4) Managing Kanban

This video is for project managers. Kanban is primarily a process management tool, which we happen to be using for software development. This video focuses on project management issues and performance metrics. It covers how Kanban requires a different type of thinking than most project management methods, how and when to adjust the work in progress limits, what kind of data Kanban generates, the use of statistics on that data, and the use of other project management tools.

The videos are each around 20 minutes. This makes them easy to schedule into a busy day and doesn't overburden the watcher with too much new information. They can be watched sequentially, or videos 1, 2 & 3, and 4 can all be watched independently.

Since pictures and motion hold attention better than text, the videos are illustrations with a voice-over narration. Voice and visuals were used to complement each other, with images and figures in the video illustrating the story told by the narration. One of the primary goals when creating the videos was to keep slides with bullet lists, specifically, and text in general, to an absolute minimum, and to add animation whenever it would aid in understanding.

Who Would Use These Videos?

These videos have different audiences; with ”Kanban: A Fast Food Example” being for laymen, “Preparing to Kanban” and “Running Kanban” for developers, and “Managing Kanban” for project managers or people managing developers using Kanban.

In the professional world, “Preparing to Kanban” and “Running Kanban” can give developers a taste of how Kanban would work for them so they could decide if it’s appropriate for their team. “Managing Kanban” would be helpful to software project managers whose teams use Kanban, showing them performance metrics and what to look for to identify problems. Finally, “Kanban: A Fast Food Example” could be used to illustrate the method to upper management, who want to know how the system works, but not really get into the details.

Another audience for these videos are software development instructors, and their students. “Kanban: A Fast Food Example” is suitable for a survey class covering multiple types of software development methods. “Preparing to Kanban” and “Running Kanban” is more suitable to students working on a project of their own, while “Managing Kanban” could be incorporated into a project management class.

Dissemination

At 124 – 236 megabytes, these videos are far too large to email around, so making them available to watch on the web is optimal for both distribution and convenience of the viewer. Currently, the best way to make videos available to the general public is through Youtube. The videos have been uploaded and their URLs are:

- Kanban: A Fast Food Example: https://youtu.be/RsOUkXC5TaA
- Preparing to Kanban: https://youtu.be/pRxxlZaiX6w
- Running Kanban: https://youtu.be/Z_c4Zo_TvM4
- Managing Kanban: https://youtu.be/WmA_1bG-yUU

The videos are in the public domain and given keywords so they can be searched. In addition to simply making them searchable, I’ll be posting the URLs on my Linked-In page, where they can be seen and forwarded. I’ll also be putting the links on my personal web page and giving a copy to the University of Alaska – Fairbanks for dissemination through the Computer Sciences department.

References used in research for this project

Brechner, Eric “Agile Project Management with Kanban”, Microsoft Press © 2015

Brechner, Eric “Agile Project Management with Kanban”, Presentation https://www.youtube.com/watch?v=CKWvmiY7f_g&t=1248s

KanbanFlow “Lean Project Management Simplified” https://kanbanflow.com/

Kanbanize “Kanban 101 – Applying WIP limits” https://kanbanize.com/blog/kanban-101-applying-wip-limits/

Kanban Tool “Kanban WIP Limits” http://kanbantool.com/kanban-wip-limits

Schreiber, Dan “Kanban 101: How to use Kanban Boards to manage your next project” https://zapier.com/learn/project-management/kanban-board/

StackExchange: Project Management “What information should be stored on Kanban Cards?” https://pm.stackexchange.com/questions/4794/what-information-should-be-stored-on-kanban-cards