Scrum vs. Kanban vs. Scrumban

Prelude

As Agile methodologies are becoming more popular, more companies try to adapt them. The most popular of them are Scrum and Kanban while Scrumban is mixed guideline set of Scrum and Kanban. Because of a different framework variety, there might be some confusion on which methodologies should a company adapt and how they differ. Each framework has its own drawbacks and benefits. In order to find out which method is best for your company we need to understand what purposes they have, as well as their advantages and disadvantages.

Scrum

Scrum is one of the dominant Agile frameworks which has a set of guidelines that manage the development of product. Methodology is iterative, measurable and incremental, since it focuses on really tightening up development cycles, which are based on scope that is divided into smaller ~~tasks~~ amounts of effort called sprints, rather than extensive planning, building, testing or deployment.

Advantages:

- **Iterative and incremental method:** allows tracking of a project workflow and provides intermediate results
- **Adaptability for product development:** scrum allows you to change priorities and requirements. Also, add modifications or features quickly
- **Participation and enhancing communication:** all team members are involved into process and motivated to express their opinion and contribute to all decisions. Team is able to easily communicate and eliminate obstacles as soon as possible
- **Cooperation:** enhanced customer and client relationships by daily communication
- **Increasing productivity:** it allows to deliver products more quickly by determining an estimation and comparing the performance of team productivity

Disadvantages:

- **Requires experienced team:** usually, scrum methodology is applied for small 5-8 people teams. Team members must be committed to the project, as this framework requires experienced team. If a team consists of novices in this area, there might be a risk of not to completing the project on time. Moreover, strict control over the team might put a lot of pressure on them which may also lead to failure
- **Time expenses:** after each sprint a new sprint planning needs to be done, which may consume a lot of time if a longer sprint is planned. Unexpected issues may also hinder the process of completing a sprint on time, thus more time will be needed to remove those issues

- **Scope creep:** scrum doesn't have a defined end date. For this reason, a released work may not have everything that the stakeholders want and new features are needed to be released
- **Iteration definition:** the scrum estimation is one of the hardest and wasteful parts, as tasks must be well defined, otherwise estimated project costs and time will not be precise

Kanban

Kanban is a Lean scheduling manufacture system developed in Japan. A Kanban system utilizes visual cues that calculate what to produce, when to produce and how much to produce. Kanban is intended to manage the workflow and increase performance which aims are to focus the efforts on the items that bring value to the end customer and remove waste.

Advantages:

- **Reduces waste and scrap:** products are manufactured only then when they are needed. In this way overproduction is eliminated
- **Inventory level management:** Kanban has great inventory practices which smoothens out inventory levels and eliminates carrying costs
- **Flexibility in production:** Kanban ensures you are not stuck with excess inventory if there is a sudden drop in demand for a product
- **Increases productivity and efficiency:** Kanban helps to eliminate the time waste and people are able to focus on current work. For this reason, making supplies is more accessible and productivity increases

Disadvantages:

- **Less effective in shared-resource situations:** suppose that upward production line is made of several parts. Downsteam line requires make more parts, but it requires a buffer to ensure that downsteam station does not run out. System becomes more complex because each part needs a separate signaling card
- **Inflexible in demand changes and product mix:** Kanban system assumes there is a stable production plan, where supplier always delivers components for production when it is needed. Therefore, the system is not suitable for industries where volumes and mixes of production fluctuate
- **Variability elimination:** system can be disrupted by unpredictable and lengthy down times and any variability may affect system's functions
- **Production flow:** because of the fact that Kanban requires planned weekly and monthly schedules linked with day-to-day flexibility, it may not be possible in an environment where multiple or short length product types are manufactured

Scrumban

Scrumban is a mixed methodology of both Scrum and Kanban which increases adaptability and universality for product manufacturing and support focused companies.

Advantages:

- **Saving time:** Scrumban uses planning on demand technique so there is no need to do estimating or sprint planning. Team plans only when there is a demand. For this reason team members get extra day of work
- **Quality:** saved time on planning allows to focus on quality control and to verify if work item is ill-formed. Saved time allows to control a manufacturing process and to inspect if work is promoted to the ready queue. If something ill-formed is found, then it gets bounced and troubles are eliminated, then process is repeated once again
- **Waste minimization:** Scrumban uses inter-process buffers and flow diagrams to show weaknesses and opportunities of the process. This gives an opportunity to eliminate everything that is not adding value to the customer

1. Applicability of Scrum

Practice 1.1 – Create a workspace

Team members need to know what is going on in a project. Use columns and rows for each sprint to represent project on the scrum board. Rows on the board depict user stories which are used as a product backlog. Product backlog is divided into multiple sprint backlogs. Columns on Scrumboard are used to depict each task card which starts with "To do" column. Generally, then follows other columns named "Work in process", "To verify" and "Done", they can be named differently. This allows you to follow the workflow of a project.
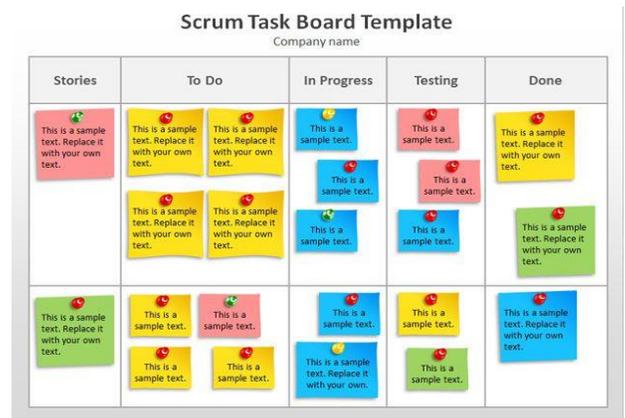
Practice 1.2 – Plan a sprint

All teams need to have a plan to govern and complete project successfully. Each sprint of a project starts with a planning meeting. The meeting is extremely important due to goals, quality and scope. During the meeting the main goals and priorities are set by product owner, scrum master and the whole team. The defined aims should be implemented until the end of the sprint.

Option No. 1 – one-two week sprint

For each project it is important to choose proper length of a sprint. Duration depends on company specifics. Two week duration is ideal because it challenges developers to complete tasks more quickly and reduces

procrastination of those tasks. Furthermore, short sprint reviews allow you to get feedback quicker and work on continuous improvement.

Option No. 2 – three-four week sprint

With longer sprints it is easier to plan and easier to start. But only disciplined teams will be able to maintain a sight of the sprint aims for the first couple weeks. Taking long term sprints there is always a risk that problems will be exposed too late. Anyway, longer sprints are recommended teams with less fluctuating environment and absence of support tasks. Also, product based manufacture can adapt longer sprints compared to short term project based.

Practice 1.3 – Create user stories

Select the work you will be doing for the next sprint in advance. Create task cards for your team members and write detailed information for each card. However, if a task card has lots of information, bigger columns should be made to contain several tasks at once, which may result in a chaotic and confusing board. To make your task board more understandable, choose different colors to group task by priorities and importance. Enter short yet informative titles, estimations and assignations. Bigger tasks should be split into small ones.
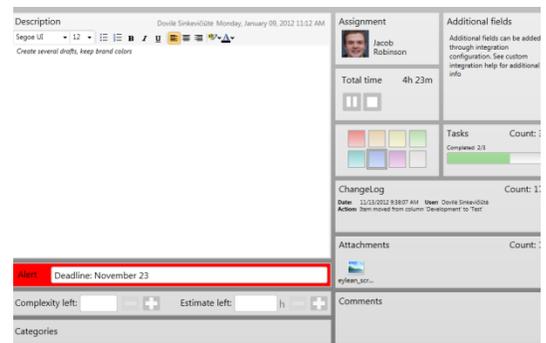
Practice 1.4 – Work during sprint

After the sprint is planned and tasks are set, only implementing them is left. Assign a task by yourself or assign them by drag and drop on the employee who should start working on a particular task. If you have problems while working with your own task, add alerts to your task card. Alerts provide the whole team with information that you need help. This allows you to get necessary help on time.

## Practice 1.5 – measure your workflow
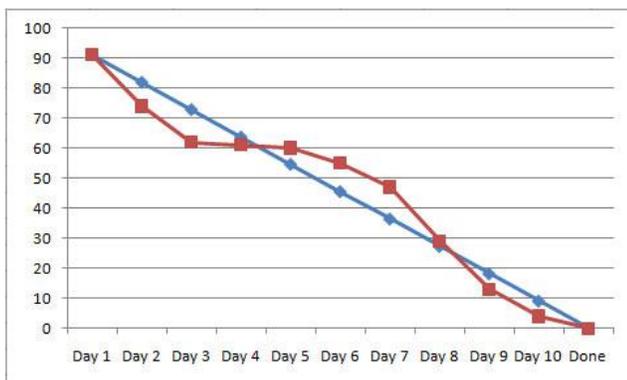
Use daily burndown charts to see if you are working according to the plan. The vertical axis shows the amount of work remaining at the start of each sprint while the horizontal axis demonstrates the sprints or one sprint timeline. The ideal blue line shows the sum of the estimates for all tasks that need to be completed. Red line depicts real situation of your team's workflow. The actual work line fluctuates above and below the ideal line depending on how effective according to timeline the team is. This allows you to follow whether your team will

complete the project on time.

2.   Applicability of Kanban

Practice 2.1– Visualize workflow

Create a Kanban board with "Input", "Work In Progress" and "Output" sections. Put task cards on the board and tag important tasks with short descriptions. Use different columns to show your team members which tasks are the most important and should be started first. Now you have an overview of the whole project that encourages your team to seek for the best result.
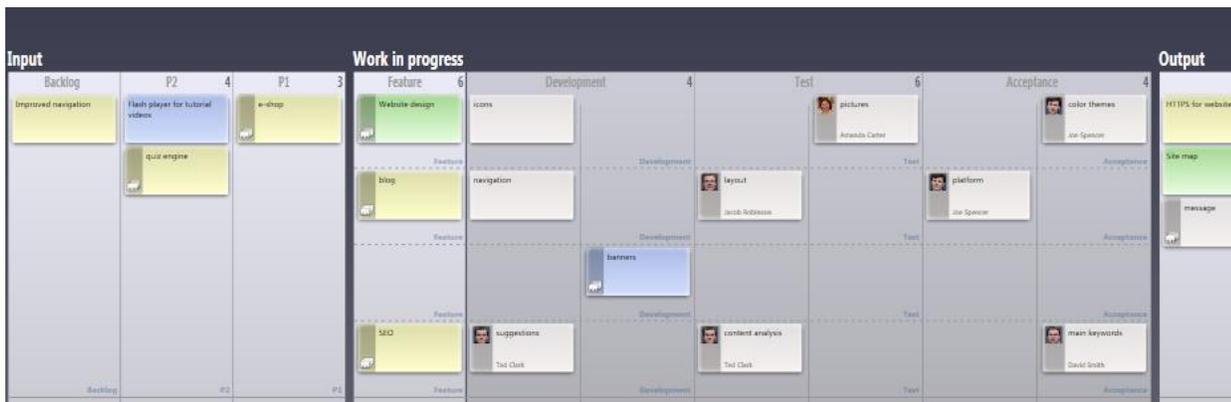
Practice 2.2–Create a workflow

Backlog can be split into two columns – backlog and backlog priorities. Now that a backlog is created, stories or tasks can be assigned to team members by the pull principle, where every team member chooses his own task according to themselves. Tasks that are in priorities column are pulled first. Work in progress column can also have a lot of other columns with steps like "plan", "development", "design", "draft", "test", "deploy", "integrate" or others. For an even clearer view of the workflow these columns may also have queue for done items.

Practice 2.3 – WIP limits
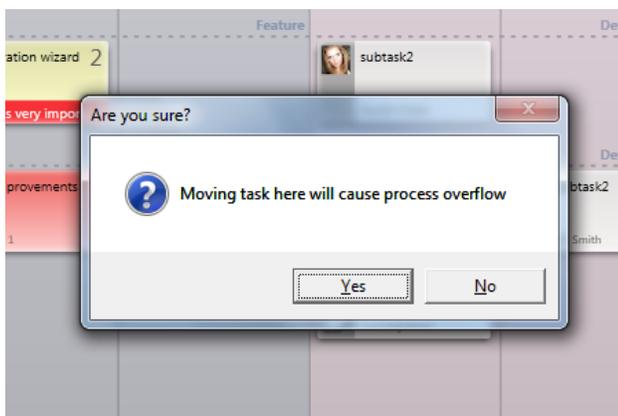
According to Kanban there should always be work limits per column to avoid multitasking, which may result in a wasted time. WIP limits helps to match team's development capacity. If your given column is set to four, then there should be only four tasks being worked on at one time. Those tasks that cannot be completed should be moved back to backlog and new task is selected from priority list. Also, there is possibility to establish limits for teams or each person. Work in progress limits are a great

to identify bottlenecks, as each smaller column of work in progress section can also have its own limits. If you are a product development enterprise which has large (lead time) duration, can considerably increase your company's profitability.

Practice 2.4 – Manage performance

Kanban uses lead and cycle times to measure performance. Lead time shows how long it takes to complete a task from its request until that request is done and delivered to the end consumer, while cycle time shows how long a task had spent in production or in work in progress section except queue. Cumulative flow diagram helps to monitor the progress of all tasks. It shows how many items are still in "to do" section, how many tasks are in "work in progress" and which of them are already finished.

Practice 2.5 – Planning routines

Kanban does not use a precise planning routine. Either release/iteration planning is used, when a new product or feature set is released, or it uses planning on demand technique, when backlog section runs out of items. Iterations are predefined timeframes in which some part of work or the goal needs to be completed. Kanban does not have predefined iterations, because different teams may choose different lengths. Some teams work continuously using short timeframes, usually shorter than one week, or choose bigger iterations like quarterly goals.

3.   Applicability of Scrumban
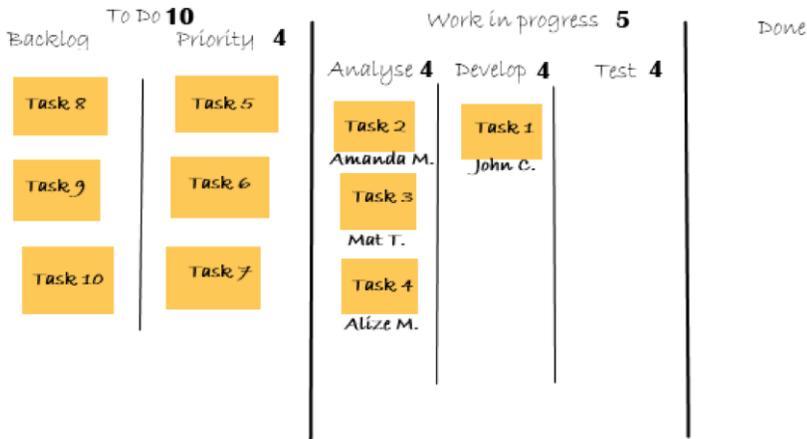
Practice 3.1 – Extend board

Create columns which consist of "To do", "Work in progress" and "Done". Column "Work in progress" can be divided into more sections, then new columns indicate the particular stage a task goes through, therefore everybody knows the current situation and tasks are completed as soon as possible. New tasks are put on the board without assigning them to a particular team member. For this reason team members are able to choose which task they would like to work with.

Practice 3.2 –Backlog limit

Make a list of tasks, put them into backlog and set work in progress limit for this column. Because Scrumban does not have regular planning meetings, limit is used to implement planning on demand technique. Team pulls items from the backlog into the process until it becomes empty, and empty backlog is a trigger which means that it is time to plan more tasks. It is better and easier to plan small, aiming only a few tasks per iteration. You can also use prioritization on demand technique which provides your team with information which task must be taken next.

Practice 3.3 – Find bottlenecks

Divide your work in progress section into smaller columns to implement separate WIP limits. This approach allows you to discover bottlenecks which block the process flow.

Backlog	To Do **10** Priority **4**	Work in progress **5**			Done
		Analyse **4**	Develop **4**	Test **4**	
Task 8	Task 5	Task 2 Amanda M.	Task 1 John C.		
Task 9	Task 6	Task 3 Mat T.			
Task 10	Task 7	Task 4 Alíze M.			

Practice 3.4 – Metrics performance

Scrumban uses average lead and cycle time as its key metrics for performance. If lead and cycle time is under control, then you can understand how long does it take for a task to reach the end consumer, how long it takes to develop and how long does it take to manage management. With these metrics you can predict how long it will take to provide a certain amount of value or earn some amount of money.

Comparison of Scrum, Kanban, Scrumban

	Scrum	Kanban	Scrumban
Iterations	1-4 week sprints	Continuous work alongside releases shorter than one week or bigger iterations like goals	Continuous work with short cycles for planning and longer cycles for release
Work routines	Push and pull principle mixed with early binding to team members	Pull principle with late binding to team members	Pull principle with late binding to team members
Scope limits	Sprint limits total work amount	Work in progress limits current work amount	Work in progress limits current work amount
Planning routines	Sprint planning	Release/iteration planning, demand planning	Planning on demand for new tasks
Estimation	Must be done before start of sprint	Optional	Optional
Performance metrics	Burndown	Cumulative flow diagram, lead time cycle time	Average cycle time
Continuous improvement	Sprint retrospective	Optional	Short Kaizen event as an option
Meetings	Sprint planning, daily scrum, retrospective	Can be avoided	Short Kaizen event
Roles	Product owner, Scrum master, team	Team and other work specific roles	Team and other work specific roles
Team members	Cross-functional team members	Cross-functional team members, specialization is allowed	Specialization or preference to tasks
Task size	The size that can be completed in sprint	Any size	Any size
New items in iteration	Forbidden	Allowed whenever queue allows it	Allowed whenever queue allows it
Ownership	Owned by a team	Supports multiple teams ownership	Supports multiple teams ownership
Board	Defined/reset each sprint	Persistent	Persistent
Prioritization	Through backlog	Optional	Recommended on each planning
Roles	Scrum master, product owner, team	Not defined, may vary	Not defined, may vary
Rules	Constrained process	Only a few constraints, flexible process	Slightly constrained process
Fit for	Enterprise maturity for teams working on product or especially project which is longer than a	Support and maintenance teams, continuous product manufacturing	Startups, fast-pace projects, continuous product manufacturing

	year		
Continuous improvement	Sprint retrospective	Optional	Short Kaizen event as an option
Meetings	Sprint planning, daily scrum, retrospective	Can be avoided	Short Kaizen event
Roles	Product owner, Scrum master, team	Team and other work specific roles	Team and other work specific roles
Team members	Cross-functional team members	Cross-functional team members, specialization is allowed	Specialization or preference to tasks
Task size	The size that can be completed in sprint	Any size	Any size
New items in iteration	Forbidden	Allowed whenever queue allows it	Allowed whenever queue allows it
Ownership	Owned by a team	Supports multiple teams ownership	Supports multiple teams ownership

Scrum

Scrum is the most restrictive method of scrum, kanban and scrumban. Scrum is a constrained process where tasks are assigned to each team member (all tasks might be bounded by deadlines), while Kanban and Scrumban uses pull technique. Scrum works well for large projects, because work is organized in sprints (usually 1-4 weeks) which are planned by all scrum team members. After one sprint ends, new sprint is planned, therefore the board is reset for each sprint. Unlike the other two methodologies Scrum has predefined roles like Scrum master, product owner and developer team. Scrum teams are cross-functional and tracks their workflow using burndown charts and have daily meetings where each member defines the current situation of his work. Larger enterprises, with larger projects usually use scrum to achieve effectiveness in a working process.

Kanban

Kanban is focused on continuous delivery of completed work or part of work. One of the most recognizable features of Kanban is WIP limits, which is used to reduce the amount of tasks that can be worked on at one time. As the work is continuous, the board stays persistent, only the tasks change. Workflow happens through a pull principle, where tasks are assigned after they went to the backlog or backlog priorities sections. New items can be added to the backlog whenever WIP limits allow it. Just like in Scrum, team members are working individually and may swarm a task if there is a need for it, however Kanban doesn't need daily meeting. There are no roles for team members, but they can choose specialization. Kanban uses cumulative flow diagrams, lead and cycle times to measure performance and the planning is done after each release/iteration or by planning on demand principle. Kanban doesn't impose strict constraints, process is more flexible. Therefore it fits best for support, maintenance teams or continuous product manufacturing.

Scrumban

Scrumban uses mixed techniques of both methodologies. It combines basic features of Scrum and flexibility of Kanban. Scrumban has a slightly constrained process where prioritization is optional, but it is recommended during each planning and planning is done by Kaizen events. Scrumban uses planning on demand principle to fill the backlog and tasks are assigned only by the pull system like in Kanban. Also, just like in Kanban, the board stays persistent, while only the tasks and their priorities change. In Scrumban the work is usually focused more on planning than releasing, while in Scrum planning is done after each sprint, Scrumbans planning is done only on demand. This method is mostly used for fast-paced process like startups or projects which require continuous product manufacturing, where the environment is dynamic.

Final Word

Taking everything into account, scrum, kanban and scrumban methods are quite distinct, as there are quite a few of them, one should understand the basic principles of all of them in order to find the best solution for their own specific needs. Agile methodologies will always benefit companies when they are applied in the right way. The most important thing is to choose a technique which fits your needs and requirements. These practices won't solve all of your problems, but definitely will make your working process more effective and productive.