# Work Approach & Practices

Our team will be applying a mixture of the Scrum and Kanban approaches in order to complete our products. Scrum will provide the structure in roles, meetings and artefacts, while Kanban principles will assist in our work flow, ensuring that we have a consistent flow of completed tasks.

**Definition of Scrum**

The Agile approach is known to be more effective than traditional waterfall model due to its flexibility, individual empowerment and quick adaptation to changes (Beck, Beedle, Grenning & Fowler, 2001). One of the most popular methodologies used for implementing agile approach is Scrum. Scrum is an iterative and incremental process that produces working product after each time-boxed iteration called a Sprint.

**Rationale for Choosing Scrum**

As a student team, each member has different priorities and schedules, thus, there is a risk of negligence and a chance of failing to meet deadline. By implementing the following practices used in Scrum, we can optimize our performance.

* Sprint: Time boxing will reduce procrastination and allow the team to produce “good enough” deliverables in short time. It also provides clear targets for the team to work towards.
* Reviews and Sprint retrospective: There will be regular meeting in lab sessions with the lecturer Jim Buchan, to help the team to make right assumptions and analyse requirements for each deliverable. Also, active discussion in sprint retrospective meetings on the performance of our team following each sprint, could help us identify problems and increase productivity for upcoming sprints.
* Stand-up meetings: The team will have stand-up meetings twice a week to review each other’s progress. This will ensure that the team knows exactly what the other team members are doing, as well as provide a chance to identify any blockages that people are facing.
* Scrum Roles: Having specific roles in the group project often leads to success.
  + Scrum master is responsible for guiding team members and helping them find solutions to impediments or obstacles.

**Definition of Kanban**

Kanban is an agile principle that focuses on the practices of minimizing work that is currently in progress (WIP), continuously delivering completed tasks, and visualisation of workflow (Kniberg as cited in Ahmed, Markkula, Oivo, 2013). It aligns the amount of WIP to the team’s capacity, in order to reduce task switching and ensure individual tasks are completed. In other words, the number of WIP tasks is limited.

**Rationale for Choosing Kanban.**

As a team with many other tasks and assignments, we feel that Kanban’s minimization of WIP, will allow us to focus on the tasks which are the most critical, at that point in time, rather than doing many tasks in parallel, at an unproductive level. Concurrently it will provide confidence that each task has been entirely completed. It will also provide a visual aid which will clearly present bottlenecks and from there impediments, which we can then solve as a team.

* Kanban board: We will use Trello to implement the Kanban board. Our board will contain phases of, “Backlog”, “To-do”, “In-Progress”, “In-Review”, and “Completed”. The phases in-progress, and in review, will be limited to 4 tasks in total.
* Tasks: we will implement tasks instead of user stories, as requirements are not as necessary because we have set tasks/goals allocated to us.

**Why Scrum goes well with Kanban**

Scrum has flaws in that, in each sprint, tasks will be completed in batches. This does not ensure each task has been fully completed. Hence, in each sprint multiple tasks switching may occur, which may lead to less productive work. As such, we have decided to also implement Kanban in order to improve our work efficiency, and ensure each task is completed thoroughly.

As a team, we will experience many interruptions due to external assignments, thus having clear set tasks will be greatly beneficial.

Scrum will however, provide the structure and time boxing that Kanban does not define.

**Quality Assurance Practices**

Assuring the quality of deliverables is most important part of the project. Therefore, we will be conducting several practices to ensure the quality of each deliverable.

* Version control: If there is any change to the deliverables, these documents will have specific versions and it will be managed using GitHub.
* Proof-read/peer review: In order to ensure that there is no logical flaw or grammatical error, project deliverables will always get proof-read by another team member. This will occur in the “In-Review” phase.
* Investigate the credibility of sources: To guarantee credibility of our work, we will thoroughly investigate references and sources that back our opinions and claims.

**Definition of Done:**

A task will be considered complete when:

* The task covers all content requirements put forward by the school (marking scheme).
* All quality assurance activities (such as proofreading) have been completed.
* All team members have read and confirmed wording and content coverage.
* The overall quality is up to the standard of the team.

**Reference**

Ahmad, M. O., Markkula, J., & Oivo, M. (2013). Kanban in software development: A systematic literature review. *Software Engineering and Advanced Applications (SEAA), 2013 39th EUROMICRO Conference on*. IEEE.

Beck, Kent; et al. (2001). *"Manifesto for Agile Software Development*". Agile Alliance. Retrieved 14 April 2016.