**Report**

# **A) Scrum Approach and Critical Evaluation**

The agile approach we are going to use is a scrum; we have chosen to use a scrum because it provides clear structure from the product backlog which allows us to use our time more efficiently throughout creating the system. Also structure is maintained through the sprint backlog as well which will help us manage our time better when in a sprint. After the sprint having a potentially shippable product increment will allow us to cut any features that are not needed if time is an issue.

Democratically deciding for Ryan to be our scrum master will help Luke and Lloyd to be free from distractions, also Ryan is the best candidate for exchanging information between the team. In the scenario of our team being too small the scrum master will step in and will assist with iterations.

## **Lean Development**

Lean manufacturing was adapted by Mary and Tom Poppendieck to fit software development, using this approach they were capable of reducing transition efforts from design to production from 6 months to 6 weeks. Lean development is a set of principles for achieving quality, speed and customer alignment.

The principles of lean development are: eliminate waste, build quality, create knowledge, defer commitment, deliver fast, respect people and optimize the whole. You can see from these ideas how similar they are to agile development and explains why agile works the way it does today.

The whole process of lean development is to be as efficient as possible, this is done by eliminating waste for example, useless meetings, tasks and documentation. The system in this principle is to be delivered as quickly as possible without wasting time, usually time is wasted working on parts of the system that might not even need to be used in the future or by multitasking too much.

In this development technique you should always look at how the team operates as a whole. Instead of looking at workflow and productivity of an individual, the principle emphasis’s on looking at the team’s progress as a whole meaning no time is wasted on writing counterproductive code for example. This means you should expect your employees best know how to complete the task at hand, so just give them what they need to be effective and then trust them to do it.

In the lean development process workers should be continuously learning and should make decisions at the last responsible moment because at this time you’ll have the most knowledge about the topic. Also in the development process each step should produce quality into the product to prevent having to go back and remanufacture a part of the system.

## **Agile Development**

The agile manifesto was created as a reaction to the larger methodologies at the time such as the waterfall model and incremental development. All these bigger methods never keyed into the main point of making a system which is helping the customer. Within the agile manifesto you’ll be able to see a lot of similarities compared to lean development.

Agile manifesto refers to a set of values which are: **Individuals and interactions** over processes and tools, **Working software** over comprehensive documentation, **Customer collaboration** over contract negotiations and **responding to change** over following a plan.

The manifesto also has a set of principles which are: Highest priority is customer satisfaction, Welcome changing requirements, Frequent delivery of software, Business people & developers cooperating daily, Build projects around motivated people, Face-to-face conversation is best, Progress measured by working software, Sustainable development pace, Continuous attention to technical excellence, Simplicity, Self-organizing teams and Regular reflection & adaptation.

If a project follows these values and principles then they could definitely be considered as an agile methodology. But there are preferred practices that are used in order to achieve agility more commonly such as, scrum or Kanban which uses management practices or extreme programming which is more of an iterative process.

These practices also overlap each other in some ways as they all follow the main core agile processes. An agile team should choose the management and technical practices that best work for them.

## **Lean and Agile Development Comparison**

My initial comparison between the two developments is the customer involvement in both (more so agile) they stress the important of customers and their opinions in developing the software. This can be backed up in the lean approach where one of its principles is to respect people and also when making decisions at the last possible moment. Making the decision at the last possible moment allows developers to take customers opinions on board for that final decision. The agile manifesto has adopted these principles from lean into one of its values, which is customer collaboration over contract negotiations. Customer involvement is also imbedded into agile’s principles as the first one says highest priority is customer satisfaction.

Another key point that links the two developments together is quality. Lean looks at building quality into the system as its being developed and optimizing the team as a whole. This is very similar to one of agile’s principles which is continuous attention to technical excellence and regular reflection and adaptation. The one difference in this case between the two is that agile is looking for quality in the technical side of things like coding, whereas lean is suggesting quality needs to be considered across all stages in making the system.

Another topic both developments cover is speed of delivery of the system, lean from its principles aims to eliminate waste and defer commitment to allow delivery to be faster and more efficient. Whereas agile tries using frequent delivery and a lot of planning prior, this allows the whole process to be fast. The two approaches from lean and agile are different but the target of maintaining speed is the same.

A main difference between the two development methodologies is efficiency, this is because lean development looks at eliminating anything in the system that doesn’t provide value and it also defers commitment to optimize efficiency. But if you look at agile’s principles you can tell efficiency is still considered but isn’t as fundamental, this is because if you look at a scrum a backlog is used where features are prioritised, but this still means in the end all features are meant to be added whereas in lean these features are just cut.

# **Reference page**

## **Part 1 (A)**

<https://hackerchick.com/agile-vs-lean-yeah-yeah-whats-the-difference/>

<http://www.poppendieck.com/people.htm>