Sustento del uso justo de Materiales Protegidos

derechos de autor para fines educativos

UCI

Universidad para la
Cooperación Internacional

Lean-Agile

The Definitive Guide

aCompiler

aCompiler.com

In this guide, I will show you everything you need to know about Lean Agile.

First, I will show you what lean agile is and why it's more important than ever.

Then, I'II help you with Lean Agile Software Development (LSD) in details, in this guide.

Sounds good? Let's dive in …

Contents

CHAPTER 1
Lean-Agile Fundamentals

CHAPTER 2
Lean-Agile Principles

CHAPTER 3
Lean vs Agile (Comparison)

CHAPTER 4
Lean-Agile Mindset

CHAPTER 5
Lean Software Development (LSD)

Chapter 1

Lean-Agile Fundamentals

In this chapter, I will help you with the basics.

So, If you are not sure what is lean-agile, this chapter will get you on the right track.

Then, in later chapters, I'II shows you a bunch of advanced concepts about lean-agile.

What is Lean-Agile

Lean-Agile is a set of principles and practices.
And primarily the purpose of lean-agile is

- Get maximizing value
- With minimizing waste

What is meant by Lean and Agile?

Lean and Agile are two different terms.

You can apply Lean and agile principles to different methods like SCRUM and Kanban.

What is the difference between Lean and Agile?

Agile is about building better products. The focus of agile is How to make a better product and not on the process.

Lean is about better value through continuous improvement of products and services.

Lean and Agile are mindset, or you can refer to them as philosophies.

In simple words, Lean and Agile are a set of principles and values. Scrum is a framework (not a methodology).

Many people consider Agile as a methodology or Framework, but Agile Manifesto does not talk about this.

What is the Difference Between Framework and Methodology?

Framework in about guidelines.

A Framework will not provide you with steps or processes to achieve a specific outcome. It is flexible, so enough room for creativity.

The methodology is all about procedure and steps of doing something.

A Methodology will provide you with a systematic procedure to achieve a definite output.

Why Is Lean-Agile Important?

In Short, Lean-Agile can help save lots of waste and maximize product value.

In fact, 80% of features never or rarely used.

Needless to say, Quality is the first and main function of a lean-agile environment.

Chapter 2

Lean-Agile Principles

To succeed with Lean Agile today, you should follow lean agile principles.

Fortunately, implementing lean agile principles isn't hard or complicated.

And in this chapter I'm going to lay out all the important principles that you can implement in your team.

(With a focus on maximizing value & minimizing waste)

7 Lean Principles for Software Development

1. Eliminate Waste

The first and important principle in Lean development is eliminating anything that does not add value.

There are many types of waste which you can avoid in software development. Here are a few which you try to avoid in your team.

a. Unnecessary functionality

If your team adds more and more functionality which is not required, it simply adds delay in delivery.

b. Unclear or constantly changing requirements

A single change in project requirement can impact project cost, its delivery and quality.

c. Task switching

Multitasking is a myth. And in software development task switching results in low team morale and lack of focus & poor quality.

d. Ineffective communication

Communication plays an important role in software development. And unclean & ineffective communication will add poor quality, delay and frustration.

2. Build Quality In

You should use best practices in your team. In Lean development, adding quality into the product is everyone's job and not for the quality analyst.

There are many important and popular tools of lean development for adding quality.

a. Test-Driven Development

Test-Driven Development or in short TDD is important.

In the TDD approach, you write the test first before the actual code. It adds quality to your product.

b. Pair Programming

You can combine the skills of two developers and avoid quality issues.

c. Code Review

Once you write code, then you share with other team members for their expert opinion.

It's a very common process to find defects or issues at a very early stage.

d. Refactor Code

After code review, you might do code refactoring to avoid any issues later in the product.

Your main focus is to enhance the product from the beginning.

3. Create Knowledge

In the Lean development principles, creating knowledge is another principle which seems simple.

But it requires discipline.

Creating knowledge principles encourages the team to document and retain valuable learning. You can achieve this by using one or more following tools.

a. Documentation

By writing the documentation.

b. By writing useful comments

This is one of the best practices especially when you commit your code in source control like Git - write useful commit messages.

c. Comments in the code

Developers use comments in the code and comment act as documentation for other developers.

d. Code Review

This process not only adds value to the product but also share knowledge.

d. Knowledge sharing sessions

You can also hold training or knowledge sharing sessions.

4. Defer Commitment

Defer Commitment Lean principle encourages the team members to accept the responsibility.

The team should NOT make decisions without necessary data.

On the other hand, the Development team should keep their options open and continuously collect information for better decisions.

In the area of uncertainty or complexity, this is especially important.

In other words - Defer commitment means

- Should not spend months in making a plan in excessive details in advance.
- Do not commit to a project without a full understanding of the business requirements.

Defer commitment favours irreversible decisions.

(More details on later chapters)

5. Deliver Fast

The customer wants fast delivery. In Lean software development team wants to deliver fast.

You should avoid the factors which slow you down. Here are a few of them.

- Do not think too far in advance about the future requirements.
- Do not over complicate solution and business requirements.

The lean way of delivering fast does not mean you should work long hours every day.

It simply means to build a simple solution and put it in front of your customers.

6. Respect People

People are the most important asset in your team.

They are the key drivers who build applications for your customers.

Without a doubt, people's respect is important to maintain a healthy culture and a productive environment.

In Lean development, you and your team encourage respect for people by

- Empowering each other to do their best work.
- Encouraging healthy conflicts.
- Effective communication is the key.

7. Optimize the whole

There is a need for a holistic view of the software development process in order to optimize the whole.
There are two important areas to improve.

- When the development team feels pressure, they release sloppy code. This increases the complexity of the code and adds more defects.

- The second issue is with testing. When testers are overloaded, they cannot give feedback to developers.

  It means the dev team may write or use the buggy codebase.

Value Stream Mapping can help with this lean principle.

Value Stream Mapping

Value Stream Mapping is a tool for visualizing and analyzing the steps taken to create a product or service.

There are four main parts to visualizing the delivery process in lean software development.

- Writing Code
- Testing Code
- Releasing
- Monitoring

Chapter 3

Lean vs Agile (Comparison)

aCompiler

Now that you understand the principles of Lean-Agile.

It's time to see the difference and similarities between Lean and Agile.

In this chapter, I will show you a detailed comparison and how Lean-Agile can help you to get Success in your project.

And in the end, I will explain what exactly is Lean-Agile.

What is Lean?

Lean comes from lean manufacturing (Toyota automobile manufacturing in Japan).

Lean is a set of principles for achieving three things

- Quality (maximizing value)
- Speed (minimizing waste)
- Customer Alignment

Lean-Mindset is the key

(more later in the next chapter)

What is Agile?

Agile is a set of values and principles used in software development.

There are six different phases in the agile software development model.

- Agile is iterative in nature
- Agile is about producing results in small iterations.

Agile Vs Lean: What are the similarities?

Here are the main three similarities between Agile and Lean.

1. Continuous Improvement

Lean and Agile, both focus on regular inspection on the product for possible improvements.

2. Customer value prioritization

Customer feedback is important. Agile actively involves customer feedback and lean’s focus on delivering quality and eliminating waste.

3.The continuous flow of results

Both focus on results.

Agile delivers value in increments (sprints).

Lean deliver value to customers by doing what is important and eliminating waste.

Agile Vs Lean: What are the Key Differences?

Here are five main differences between Agile and Lean.

1. Principle

Both have different sets of principles.

In Agile, there are 12 principles behind the Agile manifesto. It's about the scope and value of the product.

Lean offers seven principles of Software development. And these principles about product quality and its speed of development (by eliminating waste)

2. Approach

Agile and Lean both are excellent software development approaches. There is a slight difference in their approach.

In Agile, a project is developed in short incremental, iterative cycles (also known as a sprint). And this process is repeated until you achieve the desired outcome.

Lean aims to introduce small incremental change with the process. And Lean achieve this by increasing the efficiency in shorter development cycles.

4.Timeline

While both aim to deliver the product as early as possible, but their project timelines are different from each other.

In the agile, the team works in short cycles to deliver the product quickly. Each cycle typically lasts from 2 to 4 weeks.

In Lean there is no specific timeline as in agile. Lean shorten their timeline by optimizing process flow.

4. Development

Lean and Agile, both can be used in Software Development Model.

In Agile, there are six different phases in the agile software development model.

And there are different frameworks like Scrum, Kanban which helps to develop a product.

In Lean, it's derived from the Manufacturing model - Toyota Production System (TPS).

And in the Lean, the product can be developed using seven different principles.

5.Team

Lean and Agile follow different team structures.

The Agile team is self-organized. The team decides how to do the work by themselves.

And the team members have different areas of expertise, and they work for a common goal.

In the Lean, you form multiple Leam teams from relevant departments and each team headed by a team lead.

What is Lean-Agile?

Now you know the Lean and Agile, their similarities and key differences. Let's define the Lean-Agile.

- Lean-Agile is a hybrid approach; it uses both Lean principles and Agile practices.

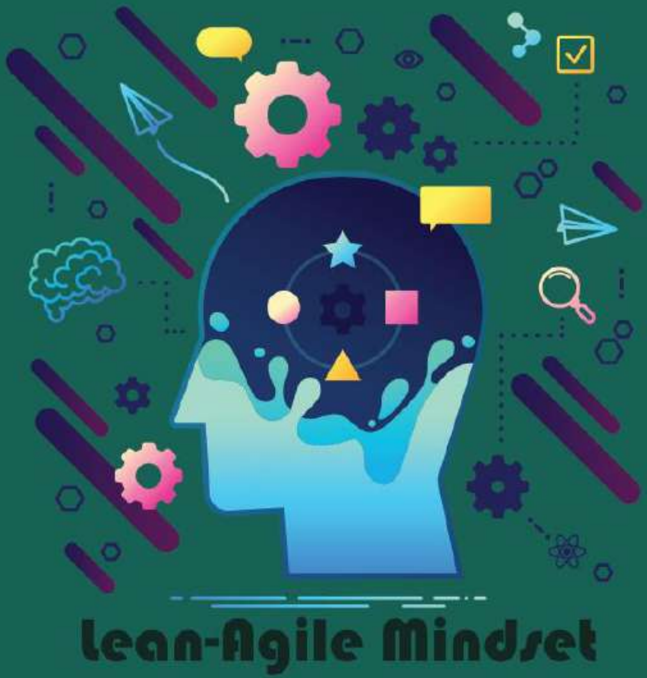
- Lean-Agile is a set of principles and practices for working that aims to minimize waste whilst maximizing value.

- A Lean-Agile process has both elements of continuous delivery and continuous improvement.

- Lean-Agile means doing less by adding the right feature in the product (and minimizing waste).

Chapter 4

Lean-Agile Mindset



As you know, Lean-Agile is about continuous delivery and continuous improvement (by minimizing waste).

And an efficient way to achieve this with a Lean-Agile Mindset.

In this chapter, I'll show you what a lean-agile mindset.

Also, I will show you can build a Lean-Agile Mindset.

(more on later)

So let's dive in …

Lean-Agile Mindset

Lets separate Lean-agile and Mindset first.

Mindset

It is a set of assumptions, methods, or notions held by one or more people or groups of people.

It's a person's way of thinking and opinions.

Lean-Agile Mindset

The Lean-Agile Mindset is a combination of beliefs, assumptions, methods, and principles.

There are two important aspects of the Lean-Agile Mindset.

- Lean Thinking
- Agile Manifesto concepts

Lean-Agile Mindset requires you to lean and embrace both Lean and Agile in behaviour.

Build a Lean-Agile Mindset

Lean-Agile Mindset is less visible but more powerful for an organization.

You can build a lean-agile mindset with practice.

Here are seven important factors that can help you and your team move toward a lean-agile mindset.

1. Respect for people

Respect for people is essential for a lean-agile mindset.

Do not make people wait and avoid all possible delays.

2. The Goal of Team Success

You should give your team a purpose and a vision to work.

Without a clear purpose and concept, it’s hard to imagine success

3. Flow

You can eliminate all those activities that have no value.

And limit the work that needs to be done.

4. Transparency

It would help if you were transparent about the work that needs to be complete and the work in progress with customers and stakeholders.

5. Innovation

Innovation adds ample room for creativity. Without innovation, your product and process with decline steadily.

Creating a space for innovation drives impressive results.

6. Positive Attitude

Lean and Agile is a journey. And in this journey, attitude is everything.

It would help if you looked at new situations as a challenge to excel and not an obstacle.

7. Improvement

Improvement encourages the learning and reflection of the process.

You can improve the whole and not some parts of the process and organization.

Chapter 5

Lean Software Development (LSD)

Now that you understand all the basic concepts of Lean-Agile, it's time to take things to the next level.

In the last chapter of this guide, we’re going to blast through Lean Software Development tips, strategies, and best practices.

Let’s dive in …

Lean Software Development (LSD)

Lean Software Development (LSD) is based on Agile and Lean Principles for optimizing development resources and eliminating waste.

It focuses only on what the product needs.

The Lean approach is also referred to as Minimum Viable Product (MVP) strategy.

In MVP, the development team only releases a bare-minimum version of the product to the market and then iterate based on the feedback.

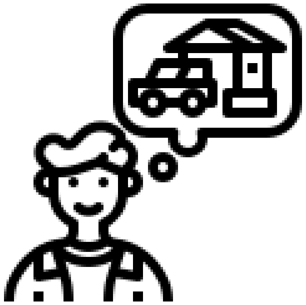
Strengths of Lean Software Development

- Deliver more in less.
- Eliminating waste
- LSD focus on market validation
- Work on Iterations and feedback
- Empower the development team to make better decisions

Weakness of Lean Software Development

- It requires flexibility in the team
- LSD requires high caliber people in your team

Five Pillars of Lean Software Development (LSD)



#1. User's need and Value

When it comes to user expectations, value is a vital link in the chain. It's about how it makes users feel.

Development teams find this value using quantitative and qualitative research.

Throughout the process, you will need to find qualitative and quantitative information that will help make better decisions on your product.

#2. Map the value stream

Once you define the value and what user's need, It's time to move to the next step.

Map all the current actions that are required to deliver a feature.

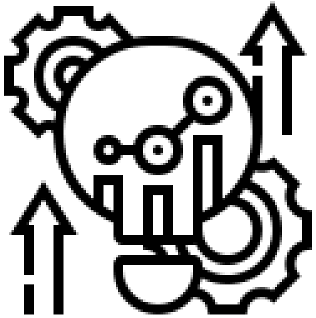
And an effective way to use flowcharts. You can analyse and improve the process by spotting complex points and delays.

#3. Flow

Software delivery is all about flow. And the process of finding and improving the paint points is called flow improvement.

Flow is the continuous delivery of finished, valuable features to our customers with short cycle times.

The faster we can deliver value, the quicker we can learn and improve our product.



#4. Product Development

After the flow you are ready to move on the product development.

As you have flow and precise customers' requirements.

The development team can start the work on the product.

#5. Continuous Improvement

To seek perfection means to repeat process multiple times until the best possible value is defined without any waste.

LSD is all about developing the right solution for the right problem. Lean Practices are about creating products that are required by users.

Lean helps to maintain a consistent value stream with users, and that helps create a sustainable product.

Want more actionable content like this?

Then make sure to stay subscribe to my twitter & website.

aCompiler