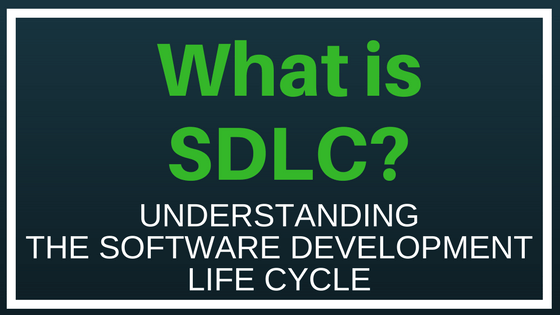
**DevOps:**

Digitalization is sweeping across many industries, creating a huge need for innovation. This innovation forces companies to be more agile and deliver faster. However, increasing the speed of your development team doesn’t happen by magic. It’s a metric that isn’t easy to change.

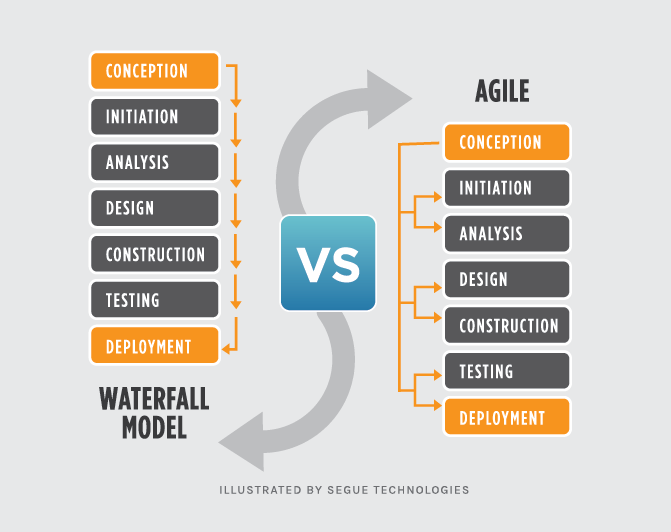


The Software Development Life Cycle (SDLC) refers to a methodology with clearly defined processes for creating high-quality software.

In detail, the SDLC methodology focuses on the following phases of software development:

* Requirement analysis
* Planning
* Software design such as architectural design
* Software development
* Testing
* Deployment

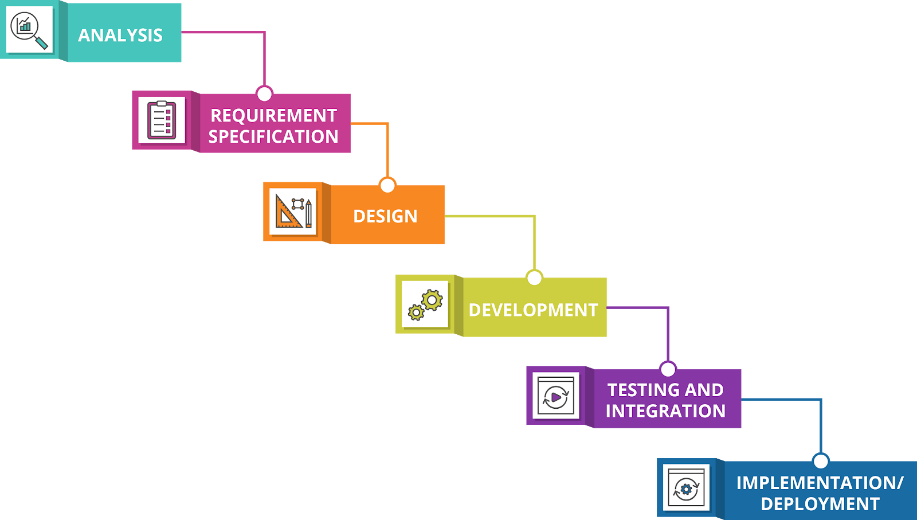
Popular SDLC models include the [waterfall model](https://economictimes.indiatimes.com/definition/waterfall-model), [spiral model](http://searchsoftwarequality.techtarget.com/definition/spiral-model), and [Agile model](http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/).



1. *Waterfall:* (ugh, terrible name!), which might be more properly called the “traditional” approach, and
2. *Agile:*a specific type of Rapid Application Development and newer than Waterfall, but not that new, which is often implemented using Scrum.

## The Waterfall Methodology

**Waterfall**is a linear approach to software development. In this methodology, the sequence of events is something like:



## The Agile Methodology

[**Agile**](https://www.seguetech.com/blog/2013/04/05/what-is-agile-software-development) is an iterative, team-based approach to development. This approach emphasizes the rapid delivery of an application in complete functional components. Rather than creating tasks and schedules, all time is “time-boxed” into phases called “sprints.” Each sprint has a defined duration (usually in weeks) with a running list of deliverables, planned at the start of the sprint.

REFERENCE : <https://www.seguetech.com/waterfall-vs-agile-methodology/>

**How can you adjust to this fast-paced digitalization???**

**DevOps** is a software development methodology that combines software development (**Dev**) with information technology operations (**Ops**) participating together in the entire service lifecycle, from design through the development process to production support.

**Devops isn't any single person's job. It's everyone's job.**

## History of DevOps

The DevOps movement started to coalesce some time between 2007 and 2008, when IT operations and software development communities got vocal about what they felt was a fatal level of dysfunction in the industry.

What is Dysfunction ?

### **Dysfunctional culture**

Actually, IT is usually divided into 2 parts. The ‘operations’ part which aims to maintain stable, working IT infrastructure to support the business. The ‘development’ part which aims to develop new solutions quickly which respond to the needs of a changing business environment. Although these 2 parts are part of IT, at best they have often worked in isolation, and at worst have viewed one another with suspicion and hostility.

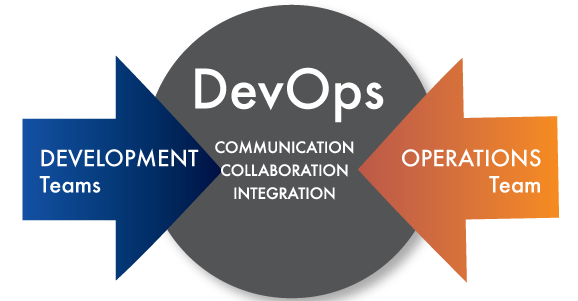
One part – ‘Ops’ – is inherently risk-averse. Procedures typically prohibit actions which would increase risk. The other part – ‘Dev’ – typically embraces risk. After all, ‘Dev’s’ job is to develop new services for a rapidly changing business. ‘Ops’ loves stability, whereas ‘Dev’ loves change. ‘Ops’ can slow down the pace of change to maintain stability, which infuriates ‘Dev’ and their business sponsors. This leads to a dysfunctional culture and missed business goals.

DevOps is a recognition that cultural improvements are required which entail a sharing of goals aligned with business goals.

They railed against the traditional software development model, which called for those who write code to be organizationally and functionally apart from those who deploy and support that code.

Developers and [IT/Ops professionals](https://www.atlassian.com/it-unplugged/devops) had separate (and often competing) objectives, separate department leadership, separate key performance indicators by which they were judged, and often worked0 on separate floors or even separate buildings.

DevOps is not a framework or a workflow. It’s a culture that is overtaking the business world.



DevOps is a concept with different interpretations and definitions, but when you get down to it, it’s all about developers and operations teams breaking down silos and working together to innovate faster. For many companies, the ability to innovate at a rapid pace — responding to market conditions and customer feedback is a key factor for success.

<https://www.knowledgetrain.co.uk/res/ebooks/what-is-devops-ebook.pdf>