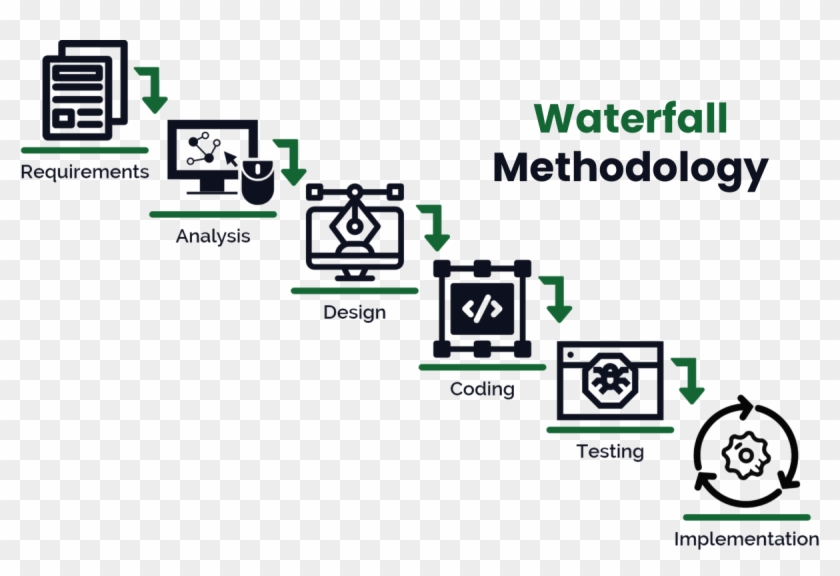
Today’s world is all about the automation technologies. Human nature is now completely dependent on these technologies.  Artificial Intelligent, IoT, Machine learning are the main Automation technology.

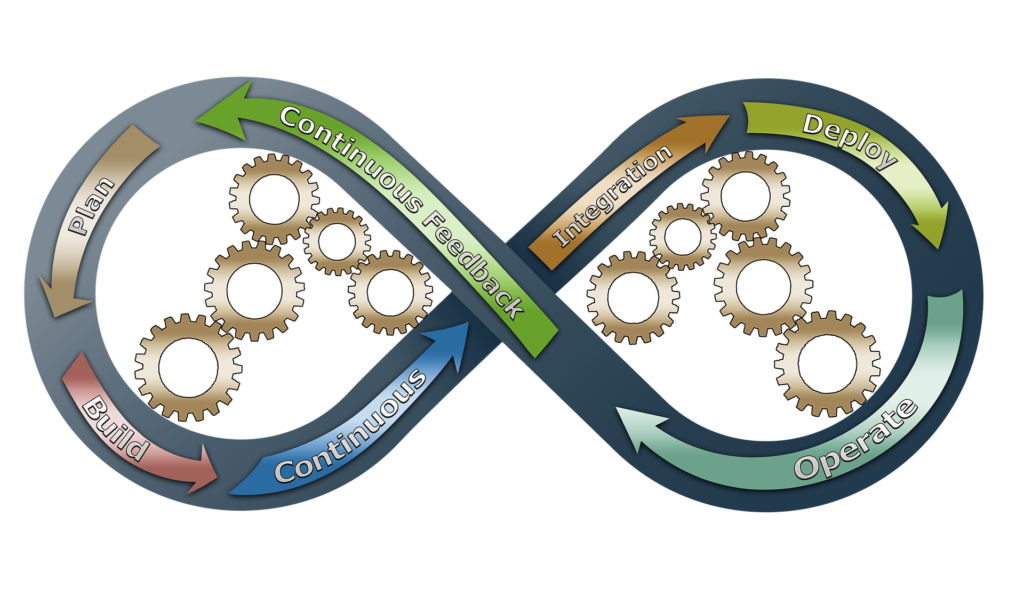
 This era consists of technology like ML, AI, Block Chain, IoT, and Cyber-Physical World. Every day numerous amount of software’s, applications, web portals are built. This Fast growing industry requires fast technologies. High Demanding Markets require more strong Software Development Lifecycles(SDLC) to build software’s with faster and efficient results.

When SDLC came in to the picture, it all begin with waterfall model. Waterfall model is very basic and easy to understand as it had simple steps to follow. Let us understand waterfall model.



Collect user requirement   > Analyse > plan > design > Code/Implement > Deployment/ release > monitoring / maintenance.

The waterfall model has its limitation. It is a linear and one-way development cycle; it takes large amount of time for its development. If any change occurs after completion of the designing and planning phase it becomes difficult to handle them. To overcome the problems of waterfall model, IT industry adopted agile methodology. The Agile method had its own limitation as it was not suitable for large and heavy projects.



To resolve all this problem software development lifecycles DevOps is more suitable for all this condition. It gives more chance to update the plan and also provide a function to adapt new things.

DevOps has more features. It is a Bridge between Developer and Operator. Many of time developers have set their own environment for developing the code on their server and after completing it they forward the code to operator or tester. The code might not run there, as it’s not a faulty code but the configuration on the other side is not same as the developer side, which causes such inconvenience.

There is another issue in old SDLC is, after the Development of the completion of code there is wait for the conformation of the tester which happens manually. meanwhile the developer starts working on new project or module. During this phase the tester confirms a bug in the code, as the development of other module has already begun its very time consuming process to correct this error. DevOps has features to overcome this issue.

Continue Development: -  Developed code with a Version Control system

Continue Integration: - Integrate all the codes as soon as the developers commits them.

Continue Deployment: - Deploy code on production or test server

Continue Testing: - Check code using software automation

Continue Monitoring: - Monitor production server and notify using mail of notification

Let’s understand this term using the example

Take an Example of my website https://1111darsh.github.io

Here is static web but for it, we need coder1(HTML coder) and coder2(CSS coder).

They use git as a version control tool. Coder 1 write code in HTML and another coder write code of website design in CSS.

As both are working on the same project they need code of each other. Without using VCT they share their code via emails or flash drives. It is a time-consuming process, and it takes a long time to understand. For understanding the Tags, they require constant communication, if there are no comments. But using the version control tool(VCT) they can share their code on GitHub. So there is no need to share code manually and there is a commit message on GitHub so you and the other person can understand what changes have been made in the code so it is faster and continuous development done using a version control tool.

After developing the site user need to deploy it on a live server for this user need to integrate all the code together. Without using CI/CD tools person fetch the code from GitHub and manually deploy it on server. If there is a version update, then do this same task manually, so for the automation there is a tool Jenkins to use this automation it new commit push by the developer then Jenkins automatically deploy on test server after test case the app is automatically deployed on a Production server.

But for a large portal there are so many servers and for conjugation of server manually is also very hectic task. Consider there are 1000 server and the company moves to python from java server then for each and every server we need to install python base library it will be done manually. we also need automation in this time. So there is a tool Ansible and Puppet. They configure server using only one master. We only need to configure one master system after that all the client are automatically configured.

In a small company, there are not many funds to use the server for testing and development. The developer code is on a desktop and given to tester but tester checks the code on server. The code is not functioning properly because configuration of the developer and tester is different. To overcome this problem, they try to match all the configuration manually which is a time consuming process. For this we have Docker.

Docker working on images and Containers. Images is blue print of container. Container is a like a virtual machine. Container is the small size of OS. So it can manage each and every server with the same configuration. Now a day Some company use this container for production. In this technology user interact with containers which are deployed on server instead of numbers of servers. Containers are also used to overcomes the monolith technology. It provides a separate container for each service. So if one service is down the other service are not affected.

A container is immutable, if any changes are to be made in the image of the container then the container will be crashed. To overcome this problem there is a Docker swarm and kubernetes available to container orchestration. It solves the problem of load balance automatically.

The above information clarifies the concept of DevOps Cycle but if the production on server. If something happens it becomes hard to find the problem so there is Nagios which is a monitoring tool with it, we can monitor our portal and resolve our problem faster.