Continuous Delivery and Deployment – Part 2

This lesson continues discussing continuous delivery and deployment.

We'll cover the following

Continuing the discussion from the previous lesson, the third upside of adopting a continuous delivery approach is:

Enables the business to stay ahead of the competition

By delivering new features to the customers more often, the business can get customer feedback earlier, subsequently making the service better than the competition. This enables the business to continually innovate and stay ahead of the curve.

We are all aware of the importance of moving fast in the industry. Competition is brutal. Evolution is happening at an exponential rate. This made the software development process evolve from the *waterfall approach* to the *agile methodology* and from quarterly code releases to *continuous delivery* and *deployment*.

If you are an early mover in the market, you kind of get a bit of leeway in perfecting your product, overcoming the limitations and issues if it has any. This is due to the fact you are the first to bring the product or the feature to the market.

Imagine releasing a buggy software or software with limited features to the end-users when you already have a competition in the same niche that is offering top-notch service.

notch service.

Here are some of the real-world examples that provide insight into the need for *continuous delivery*.

Need for continuous delivery – Real-world examples

Browser war

Browsers are our portal to the web, making them data mines to the companies. The competition to attract users to use their products is fierce. Some of the big names in the game are *Chrome, Firefox, UC, Opera,* and so on.

If you don't move fast, there is no survival as companies are innovating at a quick rate. To move fast, we need to cut down the deployment time as much as possible.

Chrome always has so many frequent updates. Whenever I happen to look at the top right corner of the browser, I always see that green update icon.

UC browser offers so many features out of the box, including messenger integrations, VPN, social features, news etc. It already has a firm grip on the Asian market.

Do you think we stand a chance if we launch a product without continuously improving it?

> ***Recommended read***: The Story Of Firefox OS

Banking

Banking systems have ideally operated on legacy systems and archaic tech, which in my opinion is a major hurdle in innovation and bringing out new features to the market. Lately, they seem to have realized this, and they have started rapidly launching new features for their customers.

I am always sort of wary of using credit cards online, especially on servers hosted outside of my country, because the government has no control over them. If a miscreant sitting in any other country gets hold of my credit card number, I am screwed. Just the card number will suffice; no security code or anything of that sort is required to use a card internationally.

Recently, a popular global bank came up with a pretty amazing feature that enables the customer to turn on and off transactions on their card via their mobile app whenever they want. It's a toggle switch. Just turn it on when you need the card, and turn it off when done. Not only does this add an extra layer of security, but it also attracts potential new customers to the business.

I always want that feature in my banking app. The bank that continually comes up with new features will undoubtedly have the maximum market share.

Gaming

In the present state of the gaming business, there is a battle between the gaming companies. You might have heard of the *Fortnite* and the *PUBG* tussle.

If you are even remotely into gaming, I am sure you would have heard of the multiplayer mobile game *Clash Of Clans*. The sole reason why *Clash Of Clans* and *PUBG* exploded was that they brought something innovative to the market, and they kept innovating. They did not give an inch of ground to their competitors. The only way to stay ahead of the curve is to deploy patches and fixing bugs in the system asap.

Another instance of fierce competition between game streaming platforms is *YouTube Gaming* and *Twitch*. *YouTube* is spending a lot of money on their gaming platform to compete with *Twitch*. Both the platforms keep experimenting on launching new monetization strategies like donation buttons, subscription-based models, etc. to attract more creators.

Faster deployments give businesses a competitive edge. *Facebook Gaming* and *Microsoft's Mixer* are other big players in the same niche.

Internet of things

This is the use case where continuous deployment really shines. Manually updating IoT devices, such as clusters of UNIX servers, home automation systems, and security devices installed in the offices, every now and then is not feasible. We need to have an efficient way of detecting and automatically updating the software that runs on these devices. This is where continual deployments come in handy.

The entire continuous deployment workflow is facilitated via deployment pipelines. In the next lesson, let’s have an insight into how they work.