Continuous Delivery CI / CD

A presentation by Sabri Bouchlema

How we can deliver value ?

Before we start developing an application, it's important to plan how we can deliver it to the end user.

Without a clear deployment strategy, we will lose our customers who are not satisfied because the software was not delivered on time and can be full of bugs.

Customers must be satisfied with the product. Trust the team that builds it.

To gain confidence, we need to minimize human interaction.

We need to automate anything that can be automated.

Happy customer ⇒ Increase income.

Solution

- ➔ Continuous Integration
- ➔ Continuous Deployment
- ➔ Continuous Delivery

1. Continuous Integration

The practice of merging all developers' working copies to a shared mainline several times a day.

➔ **Reduce cost**

- Less developer time on issues from new developer code

➔ **Avoid Cost**

- Less bugs in production and less time in testing

- Prevent embarrassing or costly security holes

2. Continuous Deployment

A software engineering approach in which the value is delivered frequently through automated deployments.

➔ **Avoid Cost**

- Less human error, Faster deployments

➔ **Reduce Cost**

- Less infrastructure costs from unused resources

➔ **Increase Revenue**

- New value-generating features released more quickly
- Less time to market

➔ **Protect Revenue**

- Reduced downtime from a deploy-related crash or major bug
- Quick undo to return production to working state

3. Continuous Delivery

A mindset that informs and enhances the practices of Continuous Integration and Continuous Delivery.

➔ Benefits

- Ensure that the team produce and release value in short cycles.

- Enhance the quality of releases.

- It’s all about delivering value !

Sources

Udacity Program - Devops