IMPROVING DEVELOPMENT LIFECYCLE WITH CI/CD

Arturo Sánchez Palacio

CI / CD

CI/CD standing for Continuous Integration / Continuous Development is one of the most relevant parts of the DevOps philosophy. Only by using a commit you are capable of deploying in an automated way. This automation may include tests in order to verify the quality of your code and to ensure that all of the key functionalities work. In addition, CI/CD really eases upstaging from development to production.

CONTINUOUS INTEGRATION

Continuos Integration makes reference to the possibility of deploying and testing changes to our app in an automated way. This allows frequent building and testing in a common repository without incurring in compatibility issues. Indirectly this allows us to have a cleaner code and specially a cleaner structure without unnecessary branches.

CONTINUOUS DEVELOPMENT

Continuous Deployment allows us to pass our code from development to production in a more structured and automated way. It’s sort of a continuation of the Continuous Integration process; once the code is built and the tests are passed you can deploy this new code with new features into production in a more automated way without needing to perform extra safety checks.

MAIN ADVANTAGES

1. Better coordination between Development and Operations Teams.
2. A more robust code since every commit is automatically tested.
3. Faster release since we also automate the deployment into production environments.
4. Faster resolution of fails when they happen.
5. Easy way of returning to a previous working version in case of big problems.
6. One of the key levers for Agile Transformation.