# Extreme Programming

Aitor Corchero Rodríguez

eurecat
Centre Tecnològic de Catalunya

Universitat de Lleida

# Speaker

# Agenda

- Definition
- Pair Programmin
- Peer Review
- Continous Integration
- Test Driven Development

> This part of the subject is focused on learning about the techniques that compose the Extreme Programming (XP). The techniques that we will expose here will serve to generate robust programs and coding.

# !

XP

> A set of techniques to empowers user satisfaction.
>
> — XP

Extreme Programming (XP) is a set of techniques to empowers users satisfaction. For that, the techniques that we will see within this module are aimed at managing teamwork towards developing a software aligned with user needs and requirements.

# Pair Programming

Pair programming is defined as two developers working together in the same machine. The role of the developers are divided between:

- **Programmer**. The one who code or perform the code.
- **Observer/Reviewer**. The one who revise the code at same moment as the program is coded.

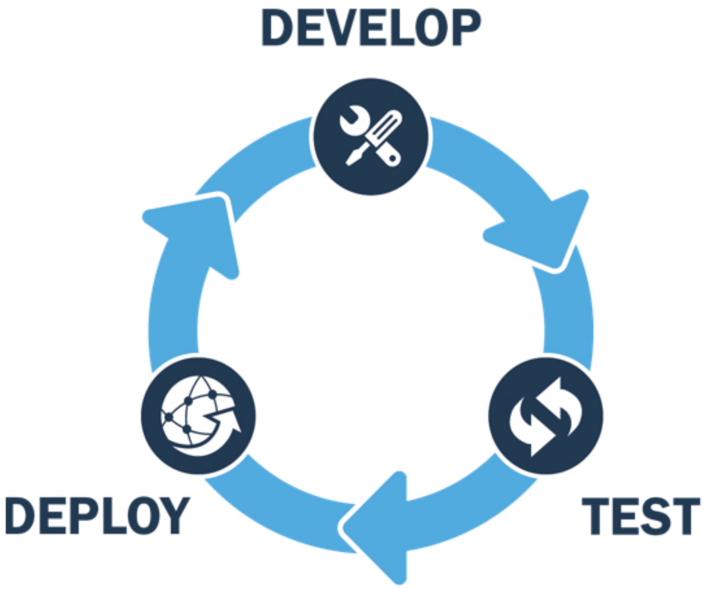
Comonly, these roles are exchanged during certain time to avoid fatigue or boring. The main benefit of using this technique is to code efficiently with less errors. It has been demonstrated that the code is developed faster too.

# Peer Review

Peer review consist on revising the code that other member of the team is developed. A person code more carefully if he knows that another team member is going to revise the code.

As a result, an improvement of code quality if it is reviewed by external person/people. Moreover, this technique also contributes to the fast correction of error/defects.

# Continous Integration

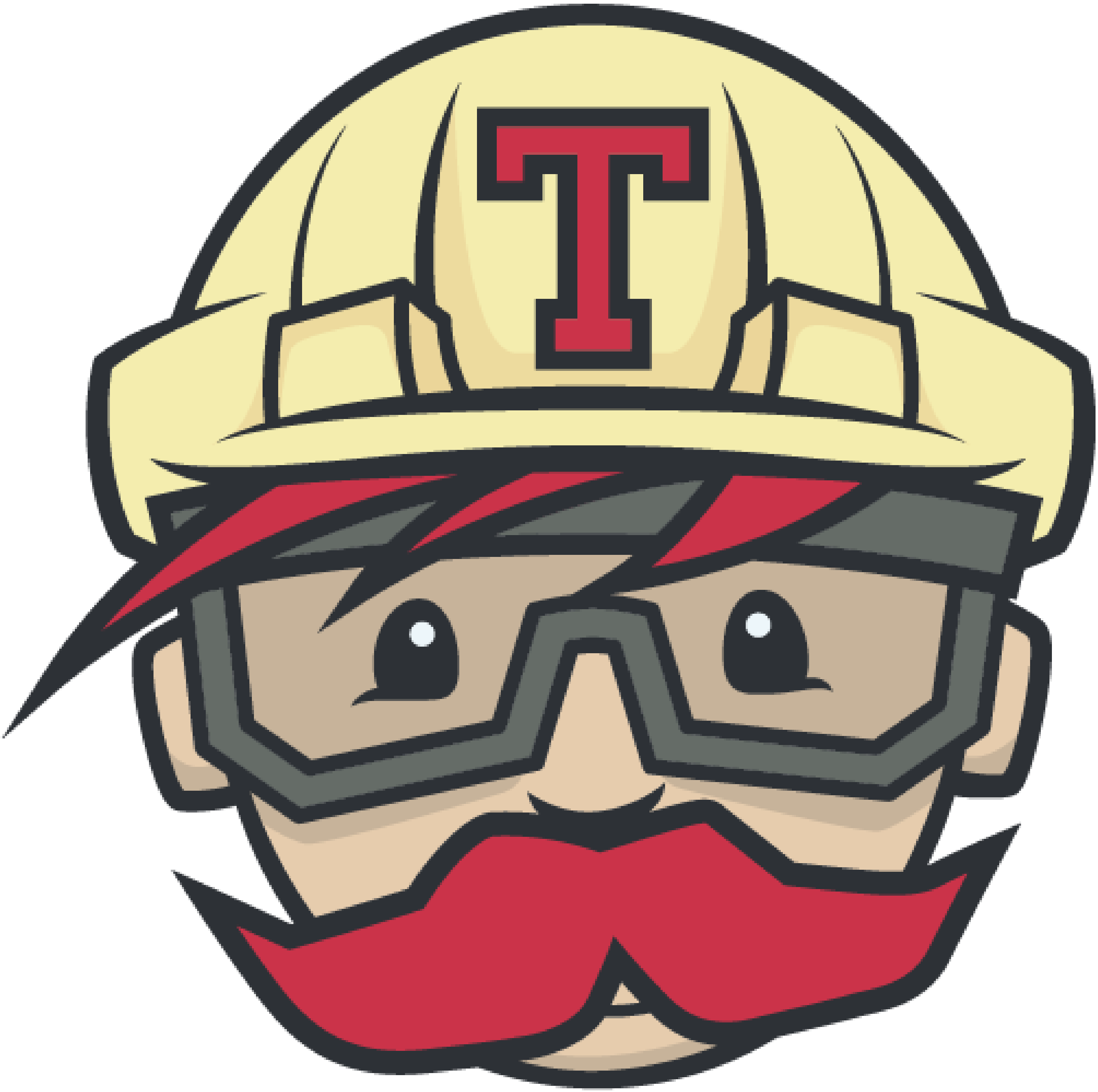


Continous integration is a cycle that permits to link the requirements (in form of user stories) with the development and testing. The concept of continous integration relies on integrating and launch the work performed by the team. Hence, the programmers/developers continously commit the work into for example github. Once the repository is updated, the CI automatically test the code and deploy the result into a web server.

Therefor, a CI helps to maintain a runnable version of the code during the development time.

# Jenkins CI

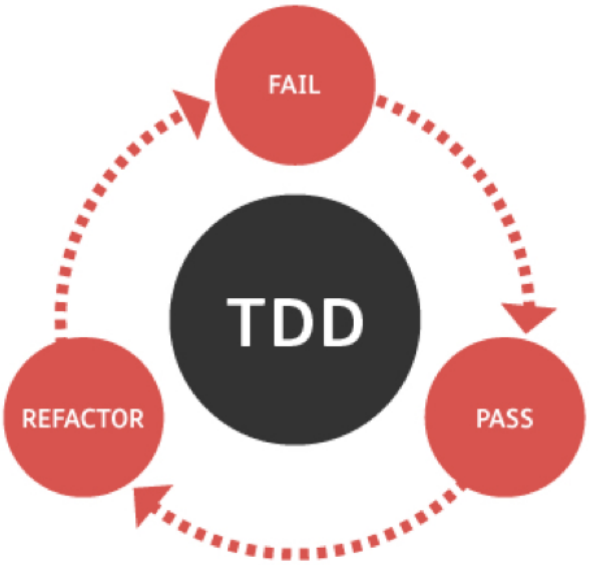
## Travis CI



# Circle CI

# Gitlab CI

# TDD

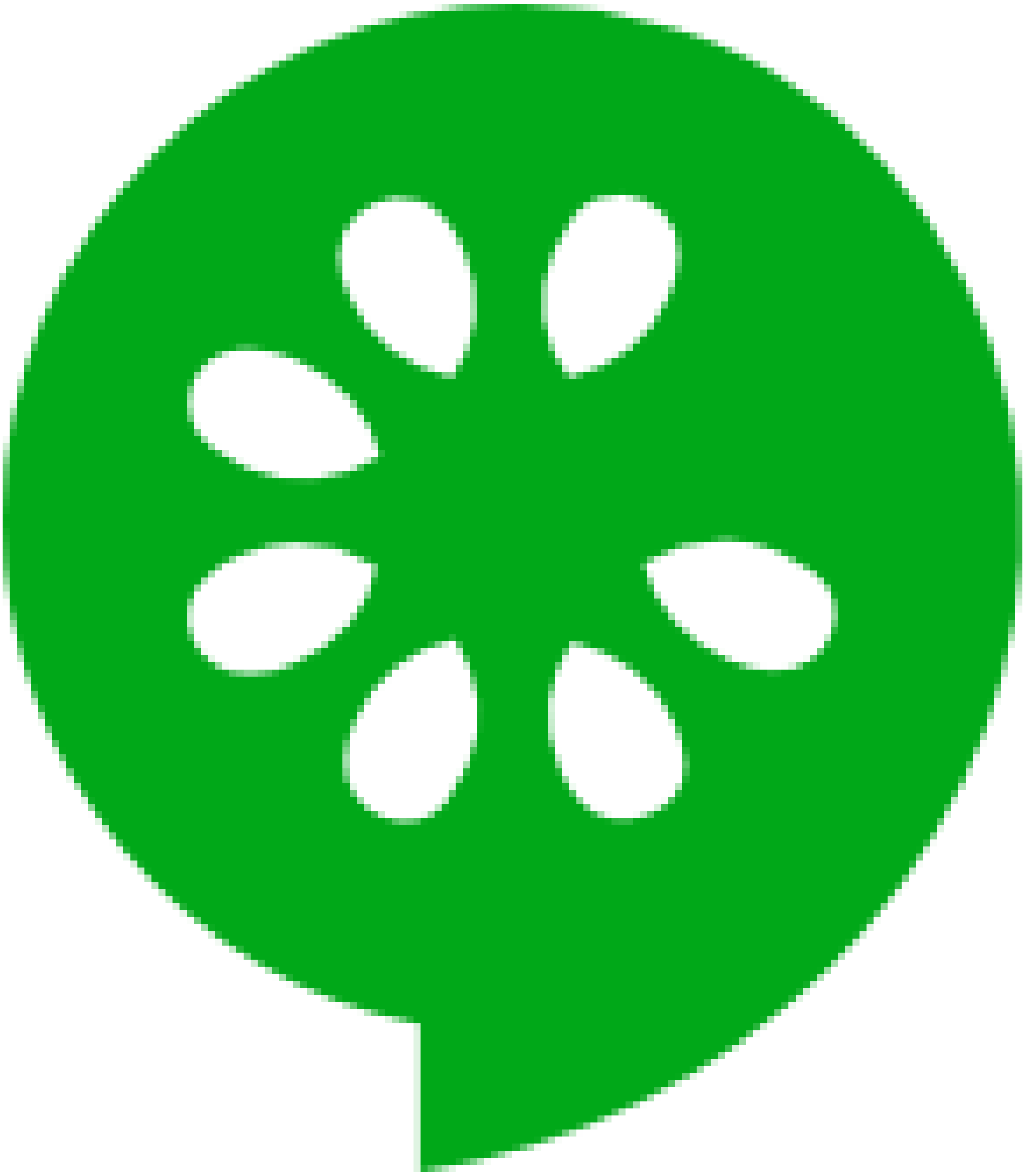


Test-Driven Development (TDD) is a cicle based on continous test. This technique is focusd on initially define a set of user stories to validate the code functionalities. Once the code is being developed the requirements and functionalities are constantly validated.

This technique could be integrated into the Continous Integration to validate the user requirements that will enable the deployment of the functional prototype. Definitelly, this technique will ensure code quality at same time as ensuring rapid detection of errors based on a set of test defined at the beginign of the project and aligned with user requirements.

## JUnit

## Cucumber



## Let’s go with SCRUM