



WHAT IS CI/CD

explained with
analogy



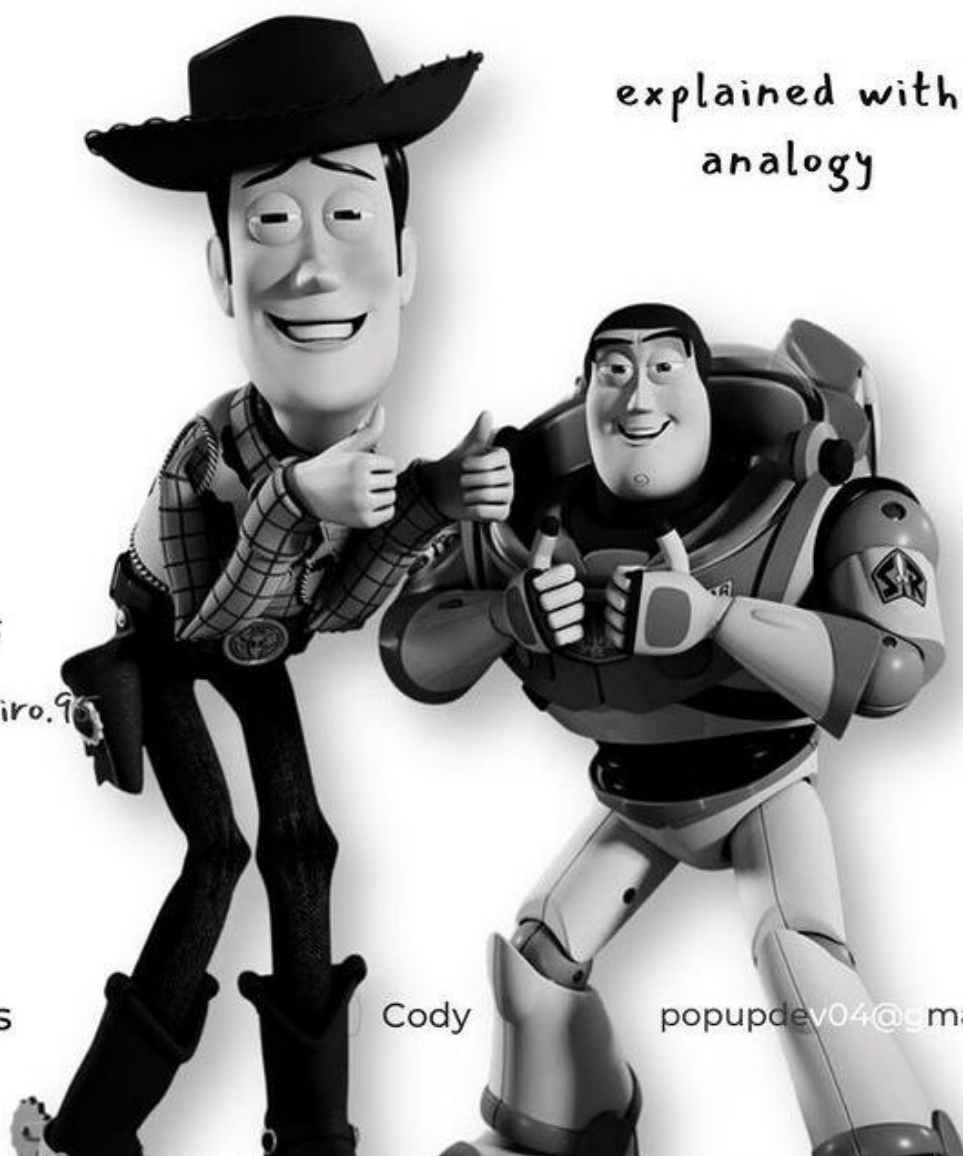
 @codechips

 @86.takahiro.95

@codechips

Cody

popupdev04@gmail.com





Lets say you are working as a
Toy Manufacturer

*I make the head
part of the toys*





Each person works on different parts of the Toy

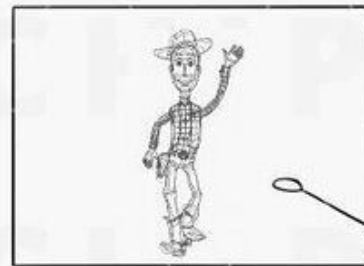


I make the clothes



I make the shoes

I do the designs



I make the hands



Before Combining all the parts they need to be
quality checked or else it will ruin the entire toy

Here are all the
work/parts done this week





And you need to combine them all perfectly
to be able to make it available to the
customers instantly

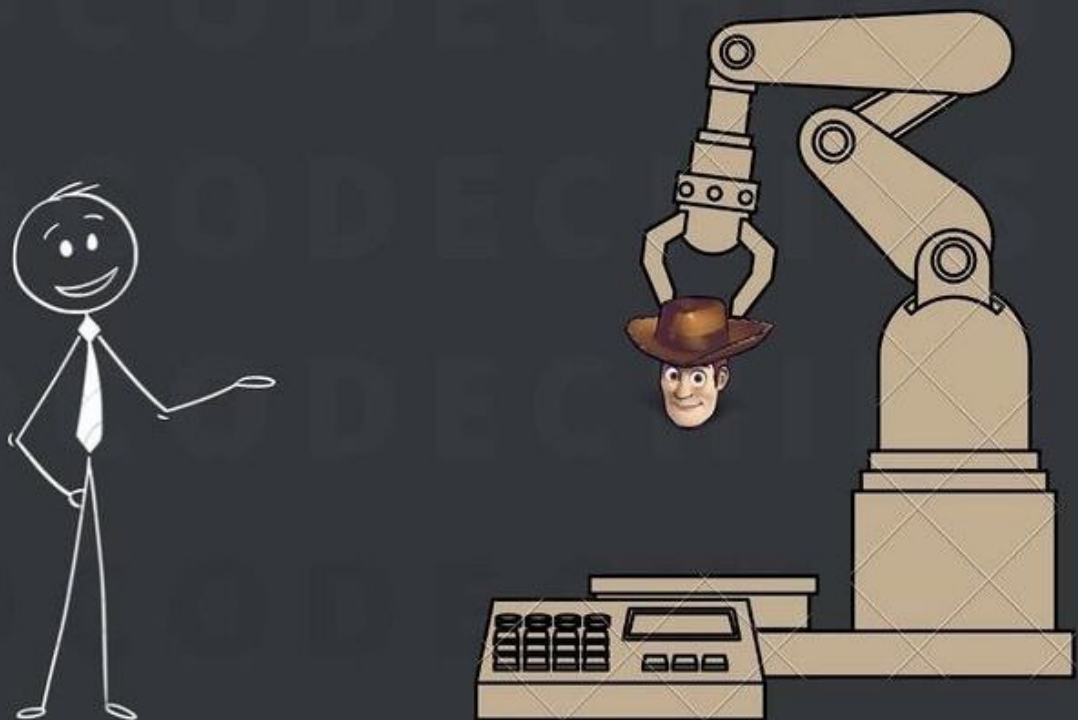
*I guess I messed up
something*





That is when your CEO brought an
Automated Machine called **CI/CD**

The employees are encouraged to input
their single finished product frequently
into the common automated machine

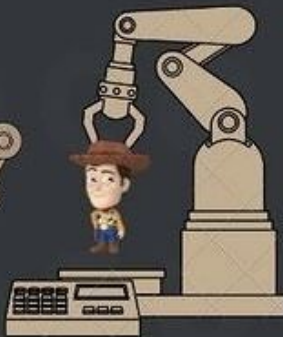




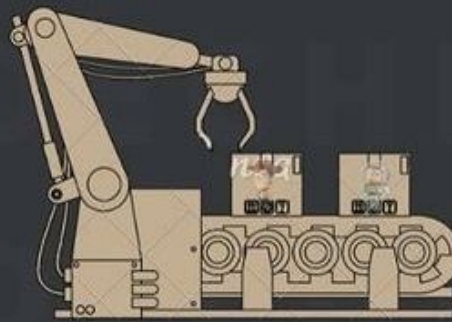
The machine automatically tests the parts, clears the bad parts, integrates all the good parts, seals the completed toy in ready to be delivered stage and also transports it to the consumers instantly



testing /
quality check



integrate /
assemble




production
ready / seal



deploy /
deliver

@codechips

 Cody

popupdev04@gmail.com



This visualizes the same
scenerio for developers



Developers and Code

CI/CD



When multiple developers work on different features & merge the entire code one day, it becomes exhausting with conflicts

With **Continuous Integration** practice, the code is merged frequently to the shared repository going through automated tests.

Continuous Delivery makes the code production ready and **Continuous Deployment** makes the working software available to the end-users instantly

CI/CD Pipeline

