Swedbank

CI/CD

Mantas Boronilščikovas
Software Engineer
mantas.boronilscikovas@swedbank.lt

Agenda

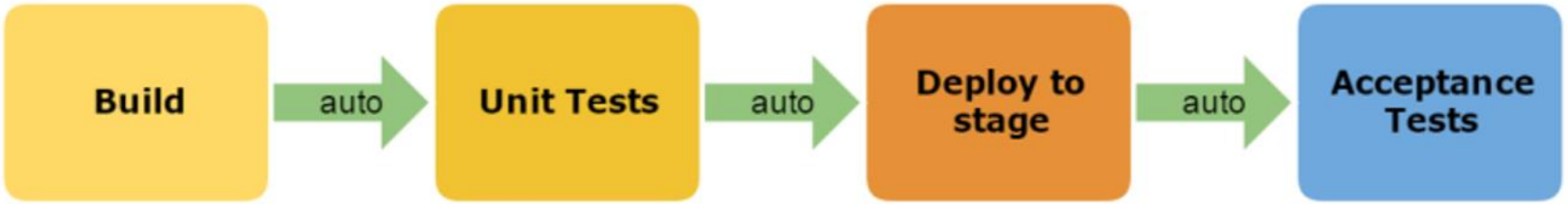
1. CI/CD explanation
2. How it looks in practice?
3. Tools
4. Practice

© Swedbank

1. What is CI/CD?

© Ankesh K,
www.linuxnix.com

Continuous Integration

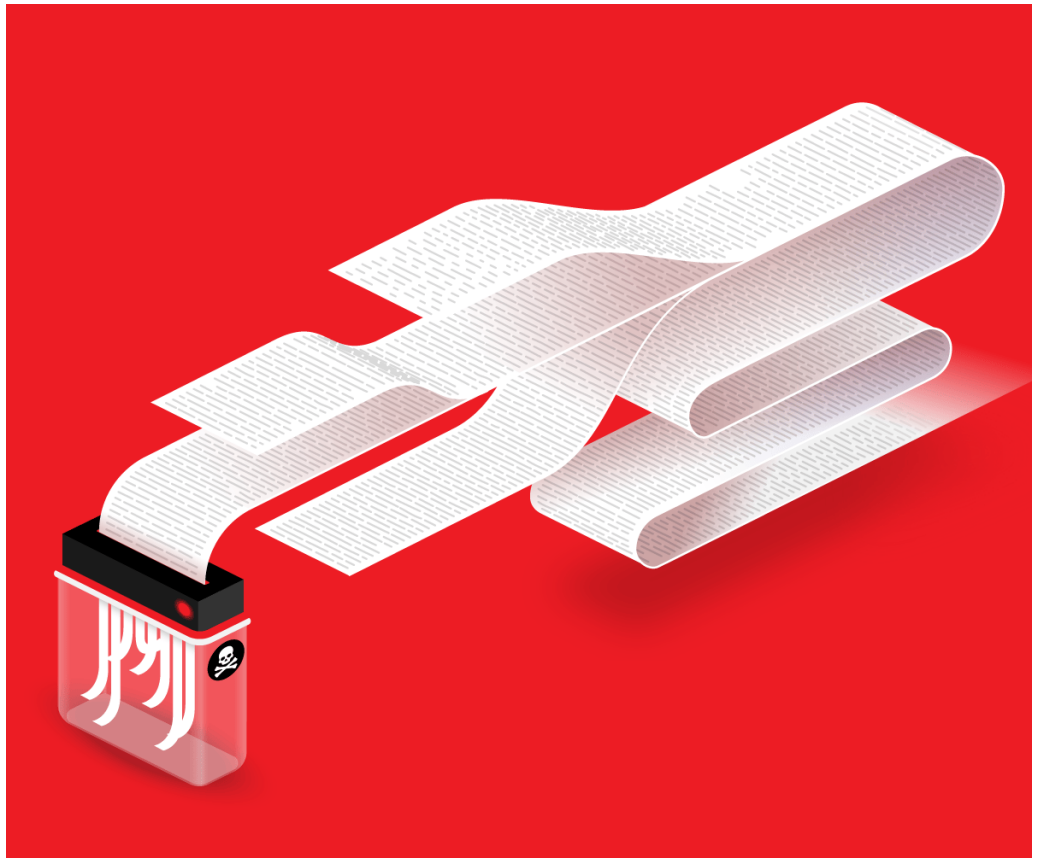


Continuous Delivery

Continuous Deployment

Why we need Continuous Integration?

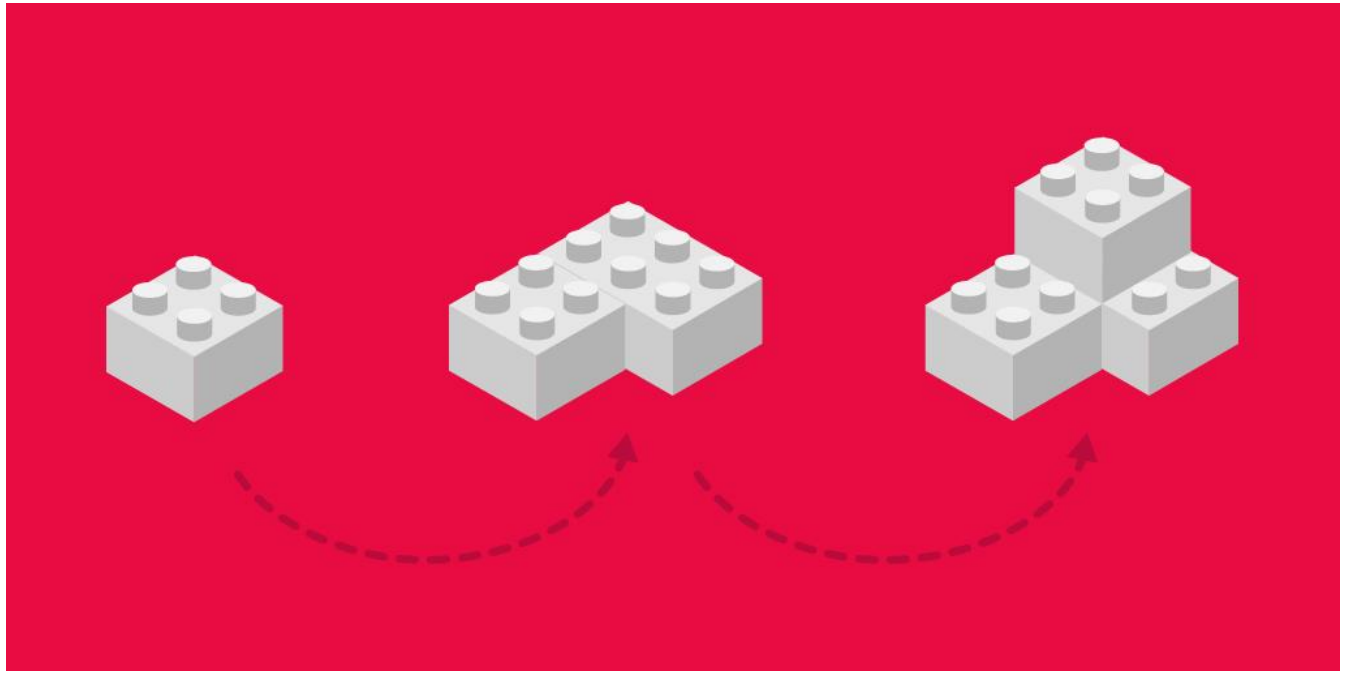
- Many developers
- Shared repository
- Frequent integration of code



© Swedbank

How Continuous Integration work?

- As often as you want
- Automated test cases
- Automated builds



© Swedbank

Benefits of Continuous Integration

- Reduces bug detection
- Automating the process
- The process becomes transparent

- Cos-effective process
- Confidency

© Swedbank

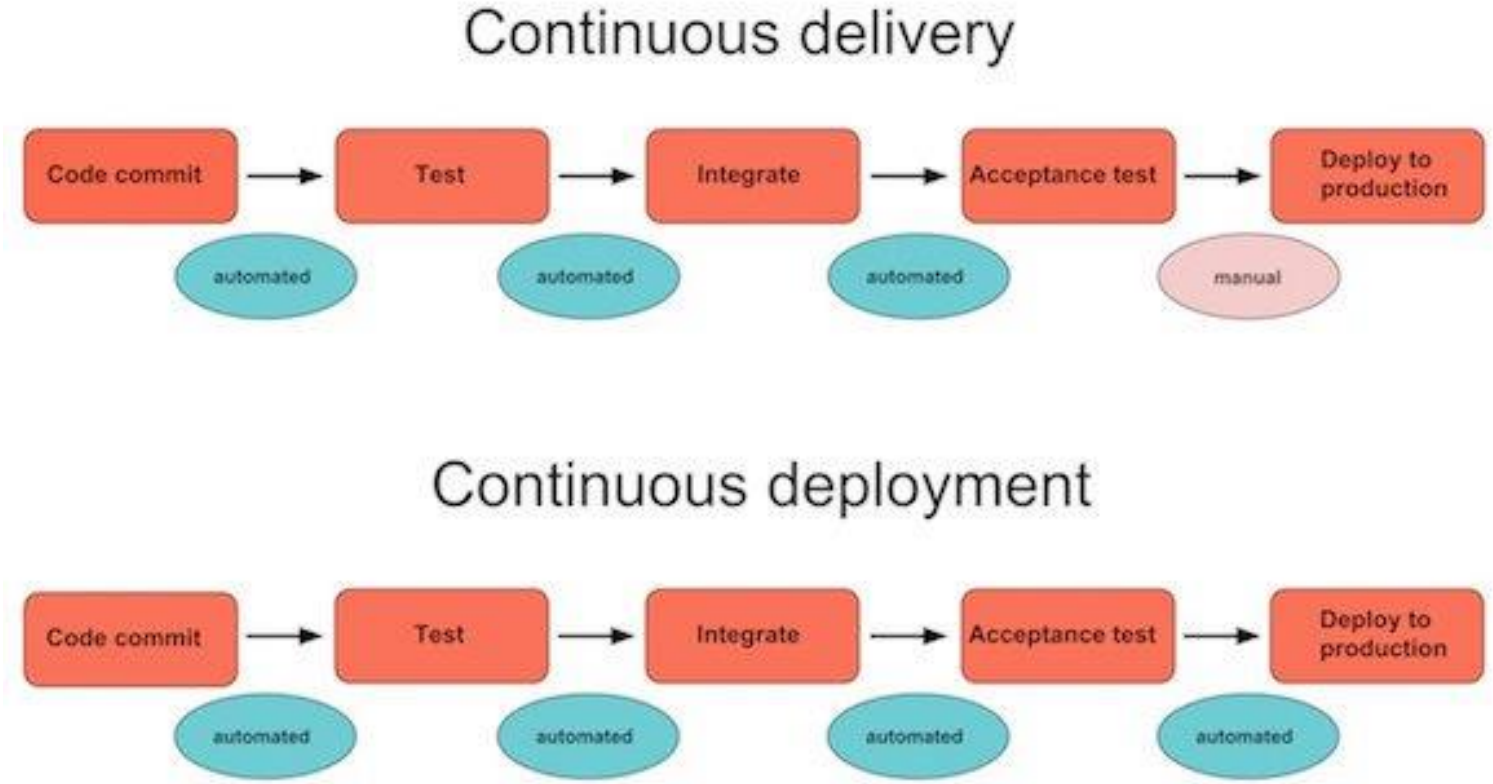
Why Continuous Deployment is needed?

- Deliver software with lower risk and money

© Swedbank

How Continuous Deployment works?

- Software build will automatically deploy if it passes all tests



© Swedbank

Benefits of Continuous Deployment

- Time to market
- Reducing the risk
- High quality application
- Reduced cost
- Happier team and better product

© Swedbank

2. How it looks in practice?

How To Perform Continuous Delivery (1)

- The developer builds their code on the local system that has all the new changes or new requirements.
- Once coding is completed, the developer needs to write automated unit testing scripts that will test the code. This process is optional, however, and can be done by the testing team as well.
- A local build is executed which ensures that no breakage is occurring in the application because of the code.

© Swedbank

How To Perform Continuous Delivery (2)

- After a successful build, the developer checks if any of his team members or peers have checked-in anything new. If there are any incoming changes, they should be accepted by the developer to make sure that the copy he is uploading is the most recent one.
- Because of the newly merged copies, syncing the code with the main branch may cause certain conflicts.
- In case there is any conflict, they should be fixed to make sure the changes made are in sync with the main branch.

© Swedbank

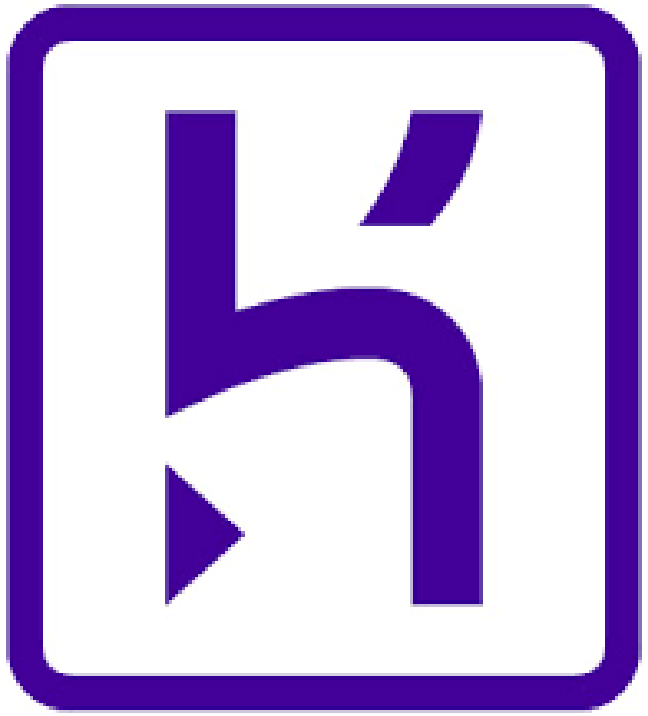
How To Perform Continuous Delivery (3)

- The changes are now ready to be checked in. This process is known as a “code commit.”
- After the code is committed, another build of the source code is run on the integration system.
- The new and updated code is finally ready for the next stage, i.e. testing or deployment. In the next section, we shall discuss some basic checklist for continuous delivery.

© Swedbank

Demo of Swedbank CI/CD

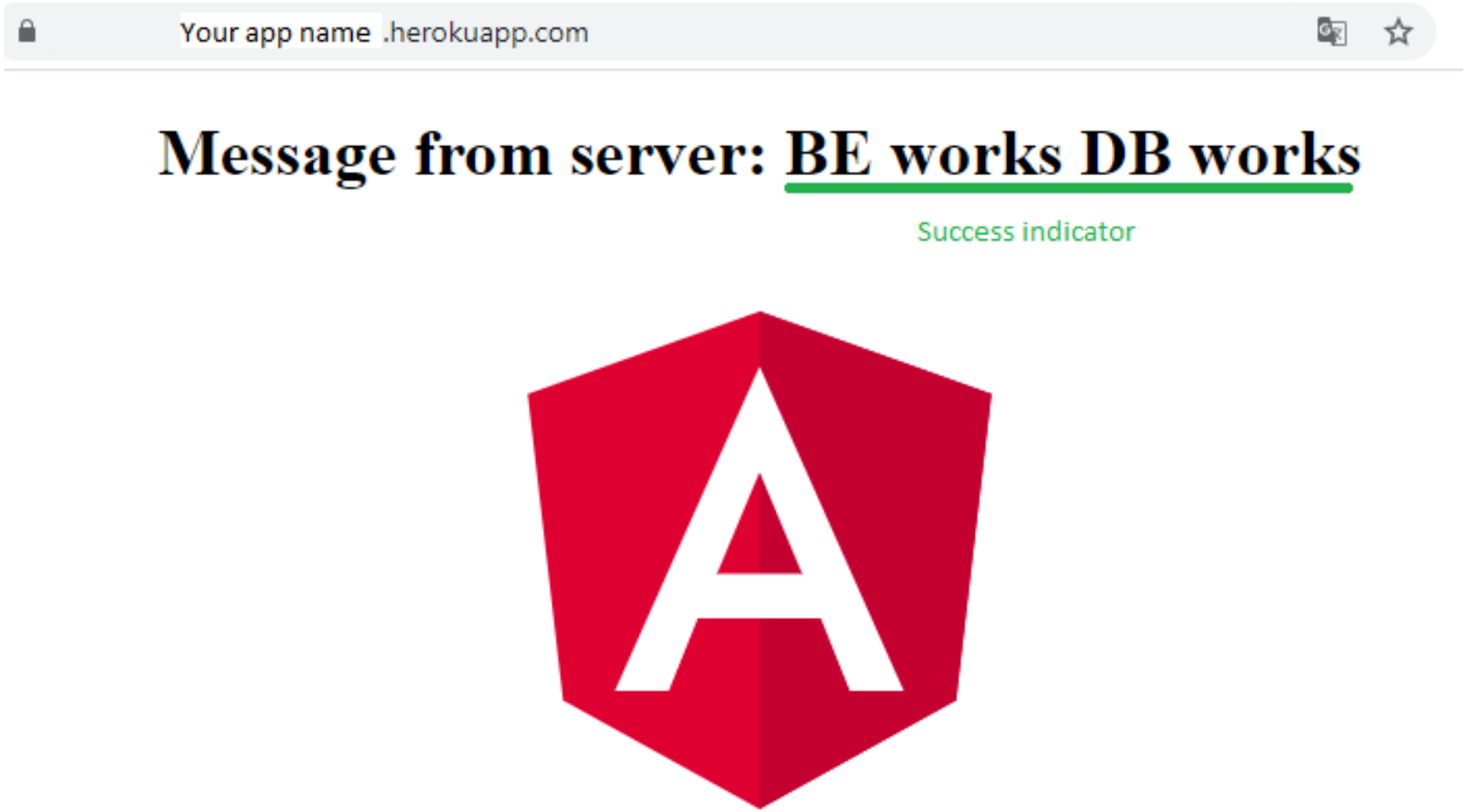
3. Tools



© Swedbank

Practice!!!

Expected result



© Swedbank