# NOTE WEEK 10

## NOTES

### 1. Lecturer introduction

- Matt Richardson: Creating software since 1998, Graduated css in 2000
- Solution architext at Octopus Deploy
- Github: GITHUB.COM/MATT-RICHARDSON

### 2. Why do he care about CI/CD/Devops? [3:10]

Dev just wanna write code but there is so much more than writting code

⇒ Devops is the area that needs the most love

Area with potential to make a big difference: gap in devlopment cycle: complete code ⇒ production

Dev face lengthy delay, result in wasted opportunities and costs

He also care abt this b'cause he love to help other dev to ship faster better with higher quality

### 3. Why should we care? [5:00]

Many companies, high functioning and not good one.

Goog one give you a sense of pride, achievement, quality of life.

The talk will give a bunch of tools that can affect positive changes in your organization

### 4. How do we publish code? [5:45]

Need to think abt this from the first time u start writing code, from day 1

**Bad for longevity:** directly publish from VStudio, only great for home projects.

**Why not build locally:** different OS, version, dll, dependencies, other random software. malware, those are not repeatable ⇒ need to build on a build server that is repeatable

**List out some tools:** Jenkins, teamcity, GOCD etc

**Build server:** scripted the whole process, enable build automation, enable consistent, repeatable builds, extensible, increase visibility,

# 5. CI [10:10]

Book suggestion

**Build server ≠ CI:** CI integrate code to repo several times, verified by automated build ⇒ detect problems early, faster detect → quicker, easier, cheaper to fix

# 6. CI Principles? [11:56]

Always use source control

Automate the build

Make the build self-testing

Everyone commits to mainline every day

Every commit should builds on the build server

Fix the build immediately

Keep the build fast

Make it Visible

# 7. CDelivery [19:28]

**CD:** produce software in short cycles, ensuring it can be released at any time, aim at building, testing, releasing faster and more frequently

CI all abt building and testing

CD all abt bring ready to release

# 8. CD principles [21:40]

Build Quality into your process ⇒ fast feedback loops

Small batch size ⇒ release more often, smaller changes, shorter time to fix bug, easier to make changes to fit client reqs

computers perform repetitive tasks, people solve problems

Kaizen - continuous improvement: always opportunities to improve things,

Everyone is responsible

# 9. The old world [29:30]

Waterfall

Over the fence mentality

Quality Gates

Low trust

# 10. Devops [31:40]

**C.A.L.M.S:**

- Culture: the way we work together (trust, communication, shared goals)
- Automation: automate to make things faster more reliable
- Automation: automate to make things faster more reliable
- Lean: eliminate the works that provides no value within the manufacturing process.
- Measure: gives we the way of knowing what to do and why and whether it success or not
- Share: share sucess, failure, learn from other mistakes. Code, toolsets, folders, languages, ownership, problems, feedback, pain.

**The three ways: [40:40]**

- The firstway: systems thinking: Devs → Ops, one direction.
- The secondway: amplify feedback loops; Devs → Ops → Devs
- The thirdway: culture of continual experimentation and learning: Devs → testing 1 loops, if ok → testing loops, if ok→ Devs

**DevSecOps: [44:20]**

Extension of DevOps, looking at things like:

- Security unit test
- Static analysis
- Security Scanning

- Continuous testing
- Bug bounty

**State of Devops survey 2019 [46:10]**

- Show strong correlation between devops practises and elite performance

link page: https://devops.com/the-state-of-devops-report-2019-is-out/

link chart: https://devops.com/wp-content/uploads/2019/09/Choudhuri_fig3.png

# 11. Book suggestions: [47:30]

# 12. Demo [48:30]

- Tool: Octopus deploy
- Go over the basic features

**How Octopus works:**

- Remote first
- High trust
- Engineer heavy
- Product Focused
- By engineers for engineers
- Do stuff in the open
- iterative approach to how we work
- evolved to the basecamp “shape up” approach
- quick response force

**How does it actually work: demo deploy using octopus [1:02:30] → [1:22:57]**

# 13. Devops in the real: [1:22:58]

- Unit tests
- Peer review
- Automation

- Fast build
- Security testing
- Fast deployment

## 14. QnA section

- nghe k rõ lắm câu hỏi gì
- Question about bug bounties

# QNA

1. **Where has Matt worked and what other things has he done?**
   Creating software since 1998,
   Graduated css in 2000
   Solution architext at Octopus Deploy
2. **What are his projects on Github?**
   Water Tank Sensor, Packer Plugin Azure, oktokit.Extensions
3. **What is a “High Functioning Company”?**
   Those ones that they will give you a sense of pride, achievement, quality of life.
4. **What is the process of publishing code? For a PoC? For real?**

   **PoC:** build locally and copy to prod server, VStudio and directly publish

   **For real:** build on a build server
5. **What can go wrong with building binary with computer?**
   Environment Differences, Inconsistent Builds, Security Risks
6. **What are the benefits of using a build server?**
   Scripted the whole process, enable build automation, enable consistent, repeatable builds, extensible, increase visibility,
7. **what are benefits of CI?**
   Detect problems early, faster detect → quicker, easier, cheaper to fix
8. **what are the CI principles (according to Martin Fowler)?**

   Always use source control

Automate the build

Make the build self-testing

Everyone commits to mainline every day

Every commit should builds on the build server

Fix the build immediately

Keep the build fast

Make it Visible

9. **How are L.E.A.N manufacturing principles interpreted for Continuos Delivery?**

   Lean: eliminate the works that provides no value within the manufacturing process.

10. **What are the CD principles (according to continuosdelivery.com)**

    Build Quality into your process ⇒ fast feedback loops

    Small batch size ⇒ release more often, smaller changes, shorter time to fix bug, easier to make changes to fit client reqs

    Computers perform repetitive tasks, people solve problems.

    Kaizen - continuous improvement: always opportunities to improve things,

    Everyone is responsible

11. **What are the "The Three Ways" in term of software development? e.g: how is your software working in practice.**

    The firstway: systems thinking: Devs → Ops, one direction.

    The secondway: amplify feedback loops; Devs → Ops → Devs

    The thirdway: culture of continual experimentation and learning: Devs → testing 1 loops, if ok → testing loops, if ok→ Devs

12. **How is DevSecOps different to DevOps testing?**

    Extension of DevOps, looking at things like:

    - Security unit test
    - Static analysis

- Security Scanning
- Continuous testing
- Bug bounty

13. **What aspects of DevOps are implemented by Octopus Deploy?**
Continuous Deployment, Release Management, Infrastructure as Code, Integration with CI Tools, Monitoring and Logging, ...etc