Desmistificando Microsserviços e DevOps: Projetando Arquiteturas Efetivamente Escaláveis

Prof. Vinicius Cardoso Garcia
vcg@cin.ufpe.br :: @vinicius3w :: assertlab.com

[IF1004] - Seminários em SI 3
https://github.com/vinicius3w/if1004-DevOps

Licença do material

Este Trabalho foi licenciado com uma Licença

Creative Commons - Atribuição-NãoComercial-CompartilhaIgual 3.0 Não Adaptada

Mais informações visite

http://creativecommons.org/licenses/by-nc-sa/3.0/deed.pt

AssertLab
Advanced Software and Systems Engineering Research Technologies

Outline

- Modern software development organizations require entire teams of DevOps to automate and maintain software engineering processes and infrastructure vital to the organization. In this course, you will gain practical exposure to the skills, tools, and knowledge needed in automating software engineering processes and infrastructure. Students will have the chance to build new or extend existing software engineering tools and design a DevOps pipeline.

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática 40 años

Objectives

- Students are expected to gain practical exposure to tools, processes, and principles of software engineering through hands-on projects while understanding models and research ideas behind the tools and processes. Lectures will include workshop style learning experiences, where students get to work on a problemset and receive feedback from the instructor and other classmates. When possible, guest lectures from industry will help illustrate examples of how the technology is deployed in practice.

Topics

Welcome, course introduction and methodology
Challenges in software Delivery in modern software development
Processes, Agile software development and DevOps
*Unit Tests, TDD, BDD
Microservices Vs Monolith: Distributed Systems concepts and common arch. Patterns
Designing software to be run in containers
Containers: Docker
Orchestrating containers
Continuous Integration/Delivery
Deploy on Cloud
Software configuration Management
Source code Management: GIT
Software configuration System: Ansible
monitoring: Elastic Search, Kibana, Logstash
Presentation of projects

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Resources

- The following books are not required but may provide useful references
  - DevOps: A Software Architect's Perspective (SEI Series in Software Engineering)
  - Ansible: Up and Running
  - Continous Delivery
  - Continous Integration
  - Designing Data-Intensive Applications
  - Systems Performance: Enterprise and the Cloud
  - The Practice of Cloud System Administration

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Warm up

Software Process Models

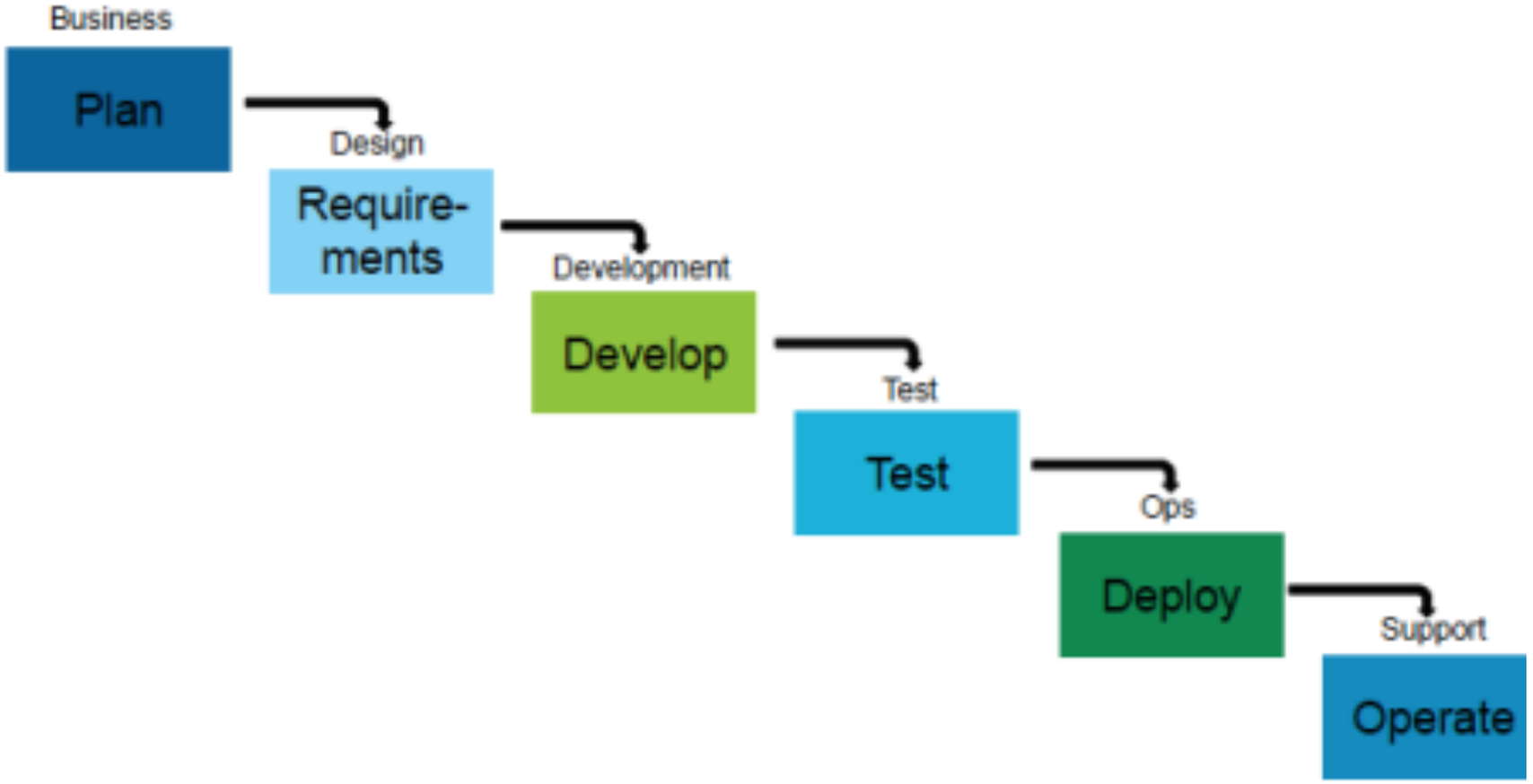
- What is a process model?
- What are some examples?
- What are some activities performed as part of a software process?

Process Model Activities

- Software project management
- Formal technical reviews
- Software quality assurance
- Software configuration management
- Document preparation and production
- Reusability management
- Measurement
- Risk management

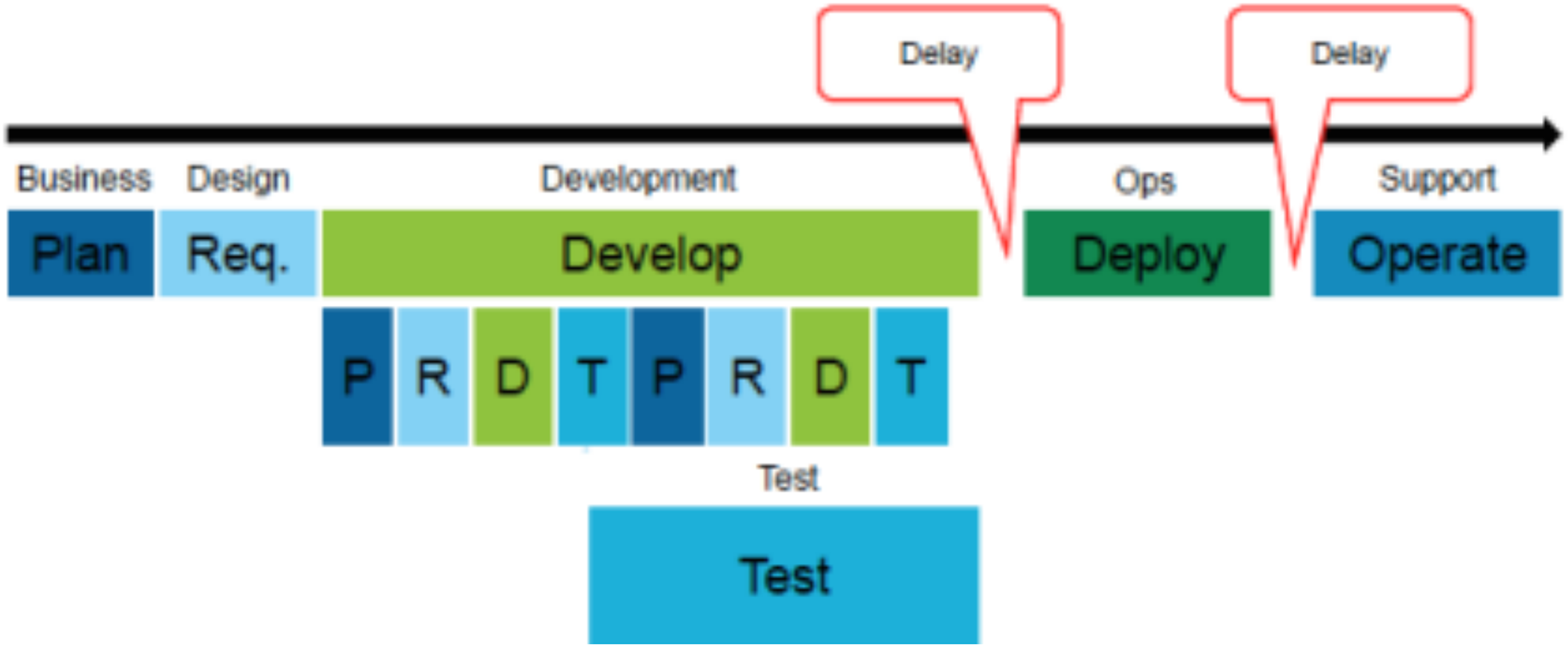
AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Waterfall model



- Problems?
- Benefits?
- Still in use?

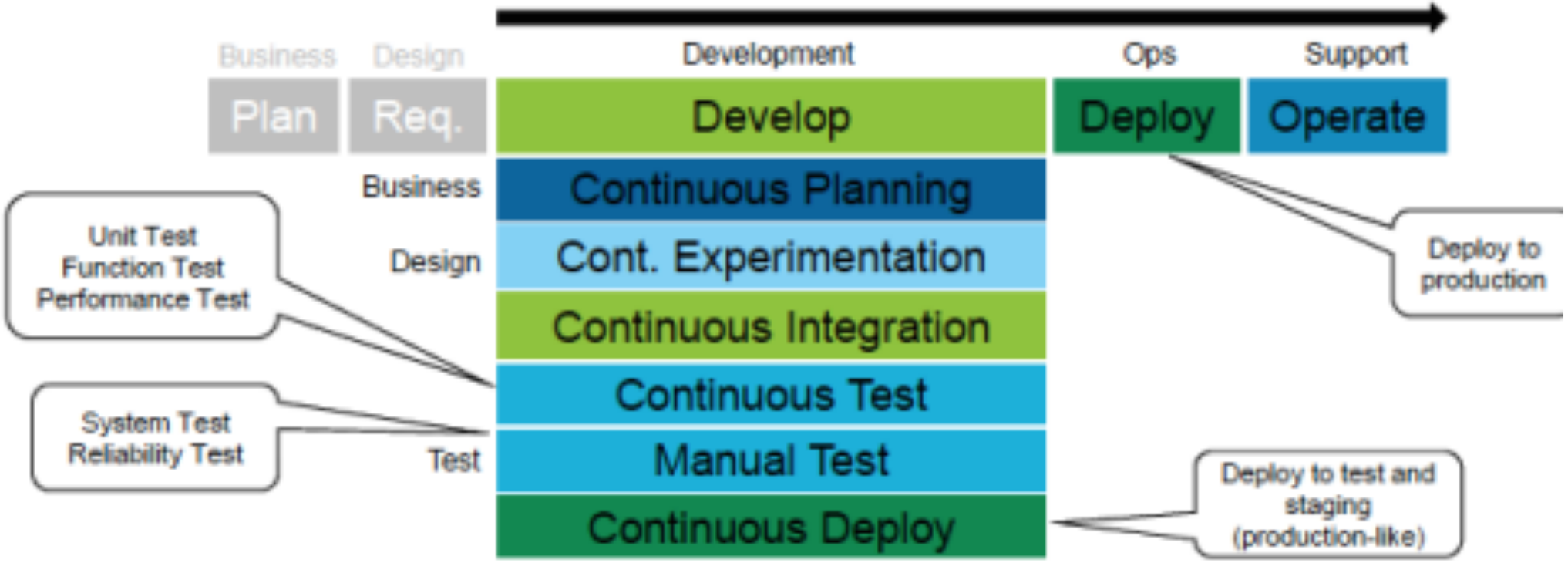
Agile Processes



- Problems?
- Benefits?
- Where not in use?

Iteration: Plan, Requirements, Development, Testing.

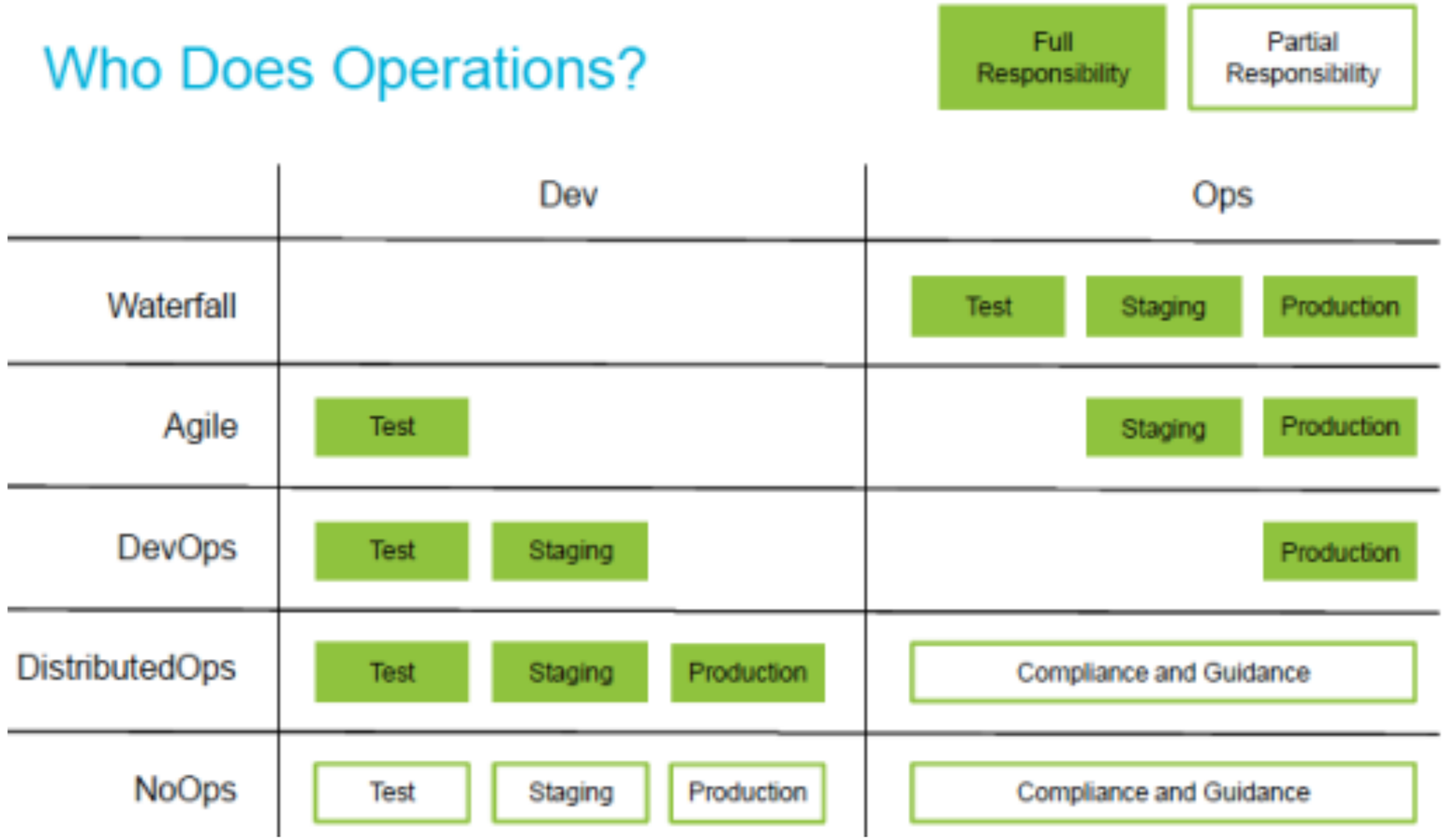
Continuous * (Perpetual Development)



AssertLab
Advanced Software and Systems Engineering Research Technologies
40 años
Centro de Informática

Operations Responsibility

Who Does Operations?



AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Exercise

- What are some tradeoffs of having a DevOps vs NoOps team model?

- What process model did you experience at internship/past company?

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Values

- No silos, no walls -- no responsibility pipelines.
- One team, owning changes, “cradle to grave”.
- You are the support person.

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Nightly Build

- Build code and run smoke test (Microsoft 1995)

- Benefits
  - It minimizes integration risk.
  - It reduces the risk of low quality
  - It supports easier defect diagnosis
  - It improves morale

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Continuous Integration

- A practice where developers automatically build, test, and analyze a software change in response to every software change committed to the source repository.

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Continuous Delivery

- A practice that ensures that a software change can be delivered and ready for use by a customer by testing in production-like environments.

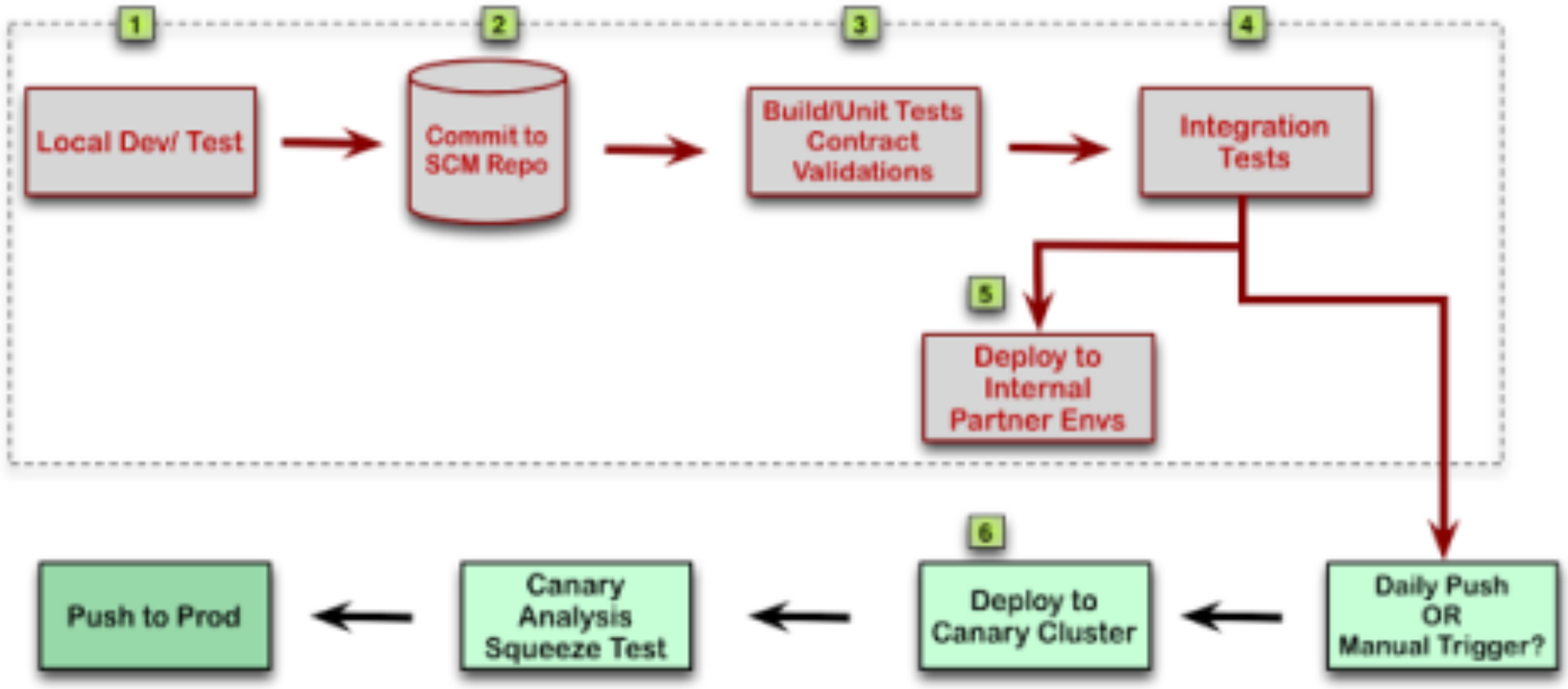
AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Continuous Deployment

- A practice where incremental software changes are automatically tested, vetted, and deployed to production environments.

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Example Deployment Pipeline



AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Exercise

- How would you design a deployment pipeline for your project or a company you might work for?

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Skills

- http://pcottle.github.io/learnGitBranching/

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Homework 1.1

- Basic course setup
  - Slack team: http://if1004-2017-2.slack.com (@cin.ufpe.br)
  - Properly setting up your Slack profile by providing a picture will help the teaching staff learn your name. Upload a current headshot picture of you (not anyone else, not a cartoon picture of you, etc.) to your profile. Besides, make sure you have your first and last name as part of your profile.

Homework 1.2

- Learning Git
  - Solve the first four levels in: http://pcottle.github.io/learnGitBranching/
    - Introduction Sequence (5%)
    - Ramping Up (5%)
    - Moving Work Around (5%)
    - A Mixed Bag (5%)
    - For extra credit, complete "Advanced Topics". (5%)
  - For submission, you only need to demonstrate completing the levels, which can be done taking a screenshot. However, you should keep track of your solutions to help you remember how to solve these types of issues in the future, or recover if your progress gets lost.

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Homework 1.3

- Hooks
  - Create a local git repository (using git init) in a new directory. Create a "post-commit" file in .git/hooks/. Inside the file, create a command that will open a web page immediately after a commit is performed to that repo.
  - Some hints:
    - http://stackoverflow.com/questions/8967902/why-do-you-need-to-put-bin-bash-at-the-beginning-of-a-script-file
    - chmod
    - start for windows, open for mac/linux
  - In your solution, provide the content of "post-commit". Finally, take a screencast, or a gif recording of the process. See details below.

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Homework 1.4

- Concepts
  - Reading 01: https://cl.ly/lu7K
  - In your own words, explain the difference between continuous integration, continuous delivery, and continuous deployment.
  - How does DevOps team model (e.g., site reliability engineer) differ than a a NoOps team model (e.g. Netflix team)? What differences in architecture allow for a NoOps model?
  - Explain the principle of Every Feature is an Expertiment
  - Explain the principle of Be Fast to Deploy, Slow(er) to Release.

AssertLab
Advanced Software and Systems Engineering Research Technologies
Centro de Informática
40 años

Submit

- Submit in our Slack team, http://if1004-2017-2.slack.com, a MD file containing the following:
  - Complete slack profile by deadline (20).
  - Screenshot of completed git tutorial (20).
  - Hooks (20)
  - Screencast (20)
  - Concepts (20)
- For your screenshot embed in the markdown file of your LOGIN-HW1.md. Include a link to your screencast video/gif. Include your concept answers in your markdown file.