

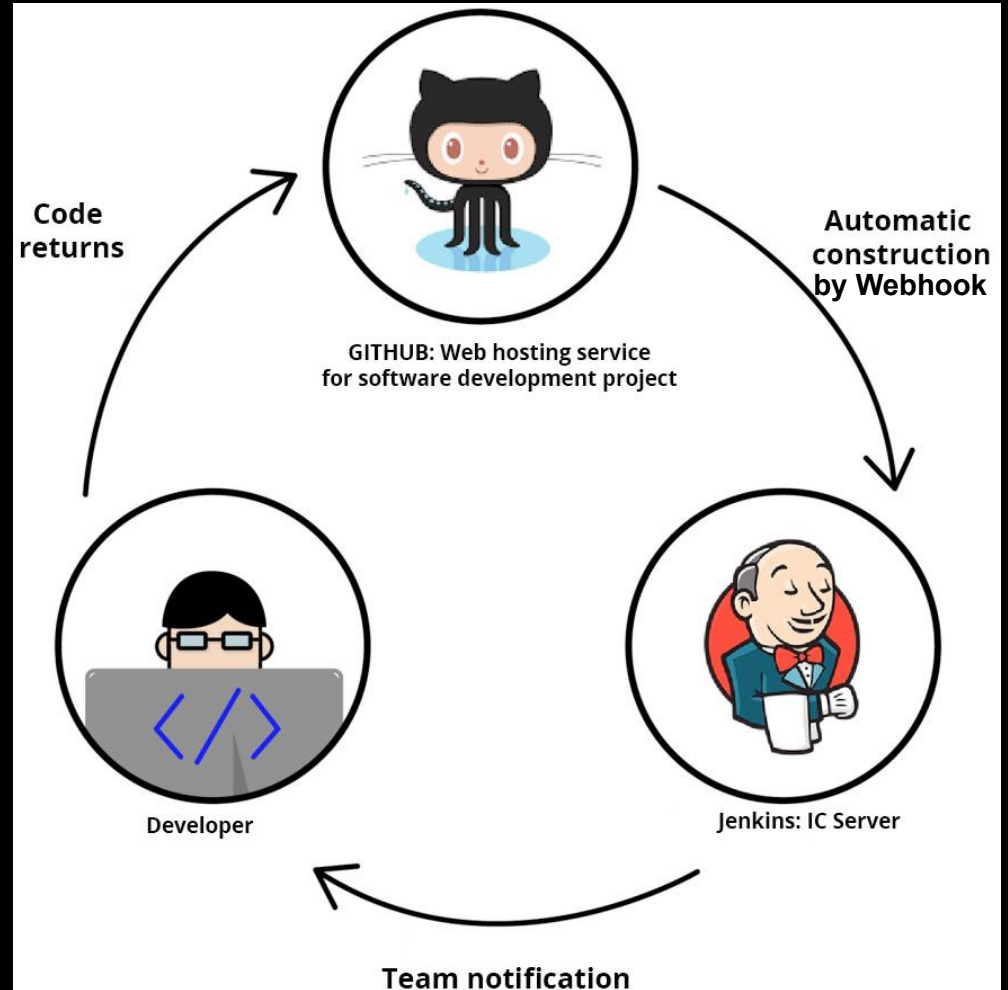
Software Project Management - Laboratory

Lecture n° 19
A.Y. 2020-2021

Prof. Fabrizio Fornari

Continuous Integration with Jenkins

Jenkins triggers a build upon every commit to the source code repository, typically to a development branch.



Build Status

README.md



SPM2020Template

This is a repository for the SPM2020 laboratory course held at the University of Camerino, Computer Science Department.

Especially it provides examples of JUNIT tests, Selenium Tests, and a sort of guide for setting up a github repository.

You can git clone it and import it as a Maven project.

license **GPL (>= 2)**

<https://shields.io>

[Jenkins] **build** **passing**

[Travis CI] **build** **canceled**

Build Status



Jenkins

search



Fabrizio Fornari

log out

Jenkins > Plugin Manager

Back to Dashboard

Manage Jenkins

Embeddable Build Status

Updates

Available

Installed

Advanced

Install ↑

Name

Version

Released

Embeddable Build Status



User Interface

2.0.3

1 yr 1 mo ago

This plugin adds the embeddable build status badge to Jenkins so that you can easily hyperlink/show your build status from elsewhere.

Install without restart

Download now and install after restart

Update information obtained: 1 day 11 hr ago

Check now

Embeddable Build Status

Jenkins > spmProject2020 >

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Configure](#)

[Delete Project](#)

[GitHub Hook Log](#)

[GitHub](#)

[Rename](#)

Embeddable Build Status

[Build History](#) [trend](#) ^

Project spmProject2020

This job is related to the spm2020 project

[Workspace](#)

[Recent Changes](#)

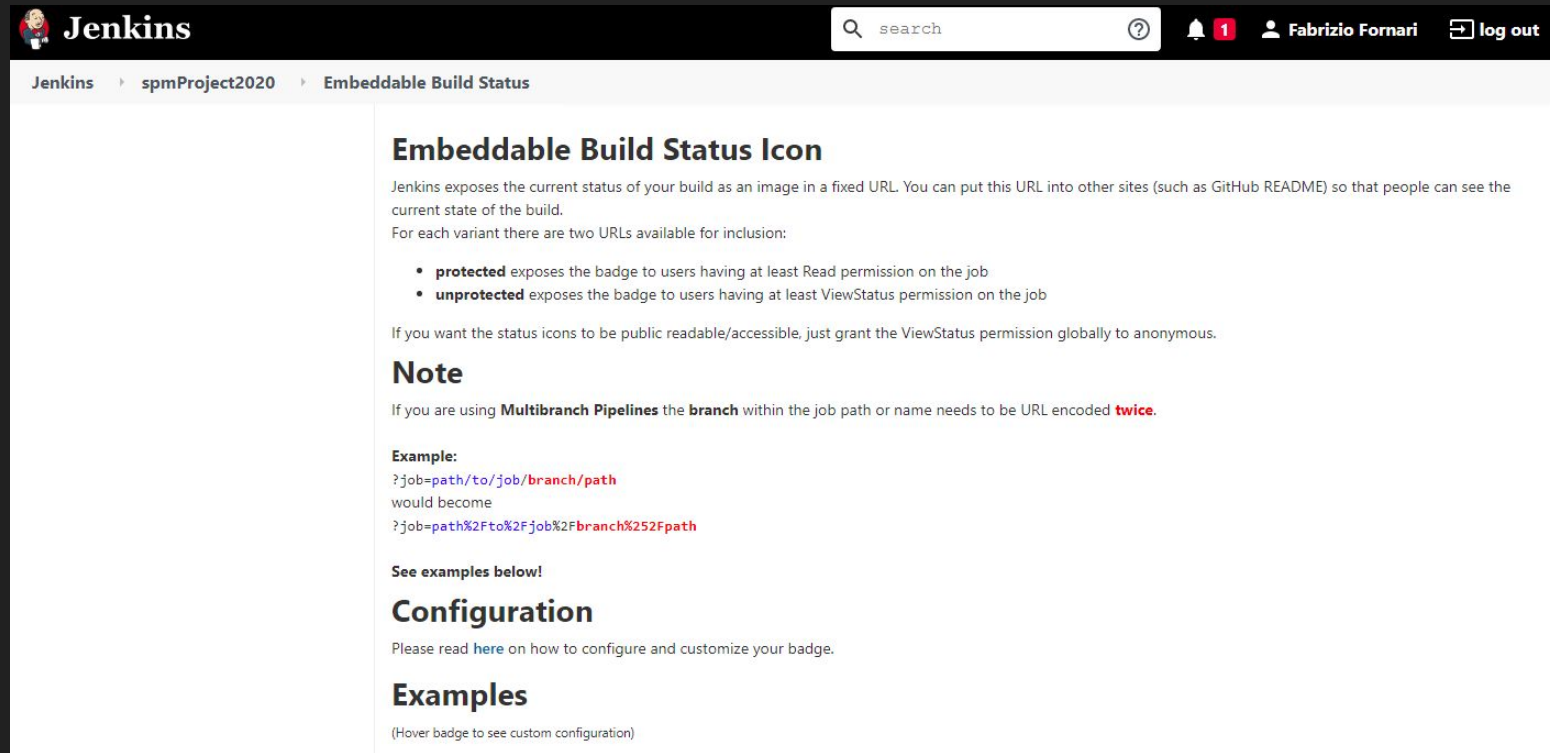
Downstream Projects

[spmProject2020AcceptanceTest](#)

Permalinks

- Last build (#87), 11 hr ago
- Last stable build (#85), 1 day 2 hr ago
- Last successful build (#85), 1 day 2 hr ago
- Last failed build (#87), 11 hr ago
- Last unsuccessful build (#87), 11 hr ago
- Last completed build (#87), 11 hr ago

Embeddable Build Status



The screenshot shows the Jenkins web interface. At the top, there is a navigation bar with the Jenkins logo, a search bar, a notification bell with a '1', the user name 'Fabrizio Fornari', and a 'log out' button. Below the navigation bar, the breadcrumb trail reads 'Jenkins > spmProject2020 > Embeddable Build Status'. The main content area has a heading 'Embeddable Build Status Icon'. The text explains that Jenkins exposes the current status of a build as an image in a fixed URL, which can be used in other sites like GitHub README. It lists two URL variants: 'protected' (for users with Read permission) and 'unprotected' (for users with ViewStatus permission). A note mentions that for public readability, the ViewStatus permission should be granted globally to anonymous users. An example shows how a path with slashes is URL-encoded. A 'Configuration' section points to a link for more details. Finally, an 'Examples' section is partially visible with a note to hover over a badge for configuration.

Jenkins

search

1 Fabrizio Fornari log out

Jenkins > spmProject2020 > Embeddable Build Status

Embeddable Build Status Icon

Jenkins exposes the current status of your build as an image in a fixed URL. You can put this URL into other sites (such as GitHub README) so that people can see the current state of the build.

For each variant there are two URLs available for inclusion:

- **protected** exposes the badge to users having at least Read permission on the job
- **unprotected** exposes the badge to users having at least ViewStatus permission on the job

If you want the status icons to be public readable/accessible, just grant the ViewStatus permission globally to anonymous.

Note

If you are using **Multibranch Pipelines** the **branch** within the job path or name needs to be URL encoded **twice**.

Example:

?job=path/to/job/branch/path
would become
?job=path%2Fto%2Fjob%2Fbranch%252Fpath

See examples below!

Configuration

Please read [here](#) on how to configure and customize your badge.

Examples

(Hover badge to see custom configuration)

Markdown:

```
([Build Status]http://address/jenkins/job/spmProject2020/badge/icon)(http://address/jenkins/job/spmProject2020/)
```

Build Status Markdown on GitHub

FabrizioFornari / SPM2020Template

Unwatch 1

Star 0

Fork 0

<> Code Issues 4 Pull requests Actions Projects 3 Wiki Security Insights Settings

SPM2020Template / README.md

Cancel

<> Edit file

Preview changes

Spaces 2

Soft wrap

```
1 # SPM2020Template
2
3 This is a repository for the SPM2020 laboratory course held at the University of Camerino, Computer Science Department.
4
5 Especially it provides examples of JUNIT tests, Selenium Tests, and a sort of guide for setting up a github repository.
6
7 You can git clone it and import it as a Maven project.
8
9 ![CRAN/METACRAN](https://img.shields.io/cran/l/devtools.svg)
10
11 https://shields.io
12
13 [Jenkins]
14 [!][Build Status](http://apomore.unicam.it:80/jenkins/buildStatus/icon?job=spmProject2020)](http://apomore.unicam.it:80/jenkins/job/spmProject2020/)
15
16 [Travis CI]
17 [!][Build Status](https://travis-ci.org/FabrizioFornari/SPM2020Template.svg?branch=main)](https://travis-ci.org/FabrizioFornari/SPM2020Template)
18
19
```

Build Status Configure Job

Set GitHub commit status (universal)

Where:

Commit SHA: Latest build revision

Repositories: Any defined in job repository

What:

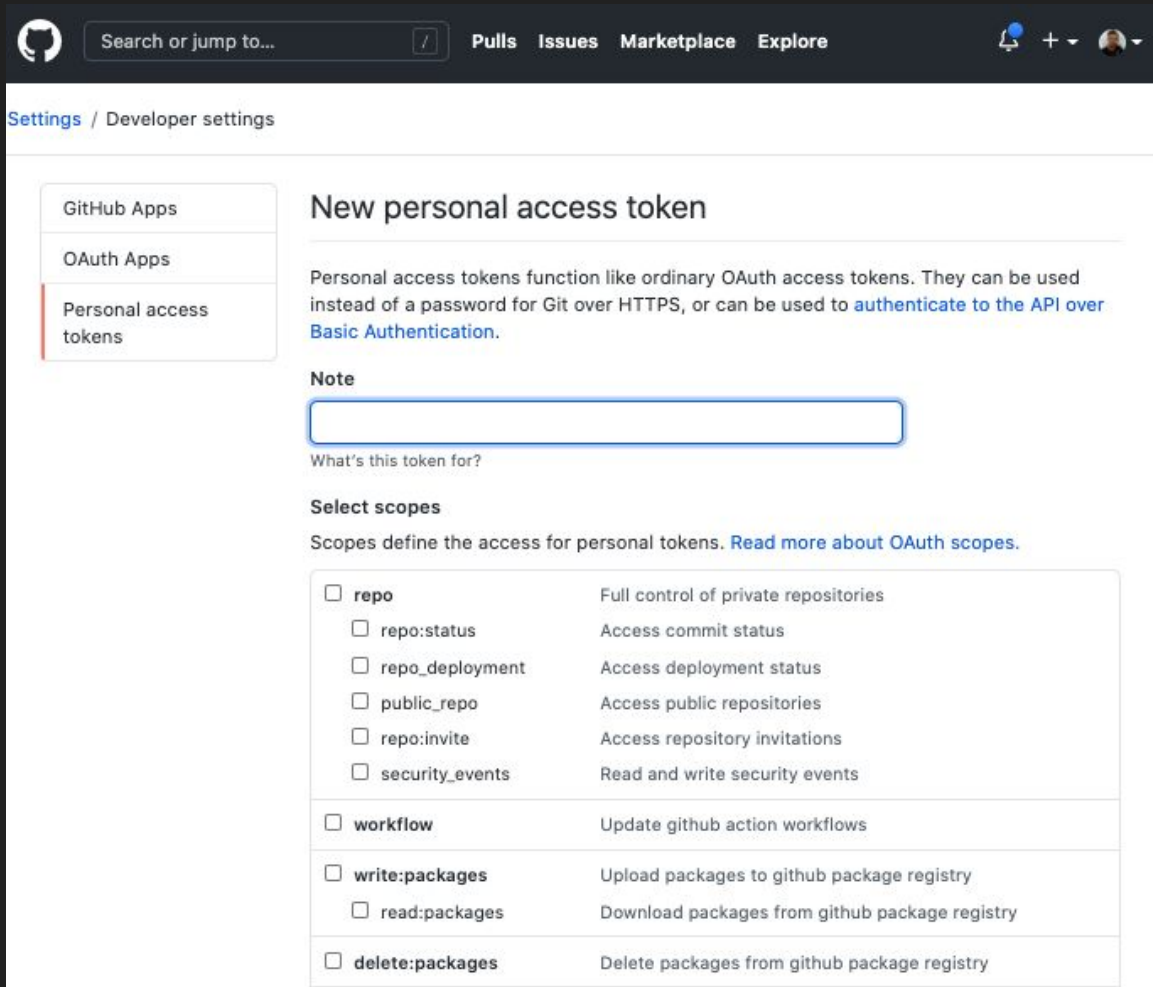
Commit context: From GitHub property with fallback to job name

Status result: One of default messages and statuses

Status backref: Backref to the build

Advanced...

GitHub Secret Text



The screenshot shows the GitHub interface for creating a new personal access token. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Pulls, Issues, Marketplace, and Explore. Below this, the breadcrumb 'Settings / Developer settings' is visible. On the left, a sidebar menu contains 'GitHub Apps', 'OAuth Apps', and 'Personal access tokens', with the latter being the active selection. The main content area is titled 'New personal access token' and includes an introductory paragraph, a 'Note' section with an empty text input field, a 'Select scopes' section with a list of checkboxes and their descriptions, and a 'What's this token for?' label.

Search or jump to... / Pulls Issues Marketplace Explore

Settings / Developer settings

- GitHub Apps
- OAuth Apps
- Personal access tokens

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

What's this token for?

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

<input type="checkbox"/> repo	Full control of private repositories
<input type="checkbox"/> repo:status	Access commit status
<input type="checkbox"/> repo_deployment	Access deployment status
<input type="checkbox"/> public_repo	Access public repositories
<input type="checkbox"/> repo:invite	Access repository invitations
<input type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update github action workflows
<input type="checkbox"/> write:packages	Upload packages to github package registry
<input type="checkbox"/> read:packages	Download packages from github package registry
<input type="checkbox"/> delete:packages	Delete packages from github package registry

GitHub Secret On Jenkins Credentials

The image shows the Jenkins configuration page for a GitHub server. The breadcrumb navigation at the top left is "Jenkins > configuration", which is circled in red. Below this, the "Lockable Resources Manager" section is visible. The main section is titled "GitHub" and contains "GitHub Servers". A single server is listed with the following fields: "Name" (empty), "API URL" (https://api.github.com), and "Credentials" (GitHub secret text). The "Credentials" field and its "Add" button are circled in red. Below the "Credentials" field is a checked checkbox for "Manage hooks". At the bottom of the server configuration is an "Add GitHub Server" button. On the right side of the page, there are several buttons: "Test connection", "Advanced...", "Delete", and another "Advanced..." button at the bottom right.

Jenkins > configuration

Lockable Resources Manager

Lockable Resources [Add Lockable Resource](#)

GitHub

GitHub Servers

GitHub Server

Name

API URL

Credentials [Add](#)

Manage hooks

[Test connection](#)

[Advanced...](#)

[Delete](#)

[Add GitHub Server](#)

[Advanced...](#)

Build Status

README.md



SPM2020Template

This is a repository for the SPM2020 laboratory course held at the University of Camerino, Computer Science Department.

Especially it provides examples of JUNIT tests, Selenium Tests, and a sort of guide for setting up a github repository.

You can git clone it and import it as a Maven project.

license [GPL \(>= 2\)](#)

<https://shields.io>

[Jenkins] [build passing](#)

[Travis CI] [build canceled](#)

Commit Status

FabrizioFornari / SPM2020Template

<> Code ! Issues **4** 🔗 Pull requests ▶ Actions 📁 Projects **3** 📖 Wiki 🛡️ Security 📈 Insights ⚙️ Settings

🔗 main ▾ 🔗 1 branch 🏷️ 0 tags

Go to file

Add file ▾

📄 Code ▾



FabrizioFornari Update README.md

● 1a88ba3 11 minutes ago 🕒 124 commits

📁 .settings

📁 WebContent

📁 drivers

📁 resources

Some checks haven't completed yet

1 pending and 1 successful checks



continuous-integrati...

[Details](#)



spmProject2020 — ...

[Details](#)

webapp

18 days ago

Parenthesis Closed

Software Development Process

Software Development Process is the process of dividing software development work into distinct phases to improve design, product management, and project management. It is also known as a software development life cycle (SDLC)

Waterfall Development Process

The Waterfall Model is document driven.

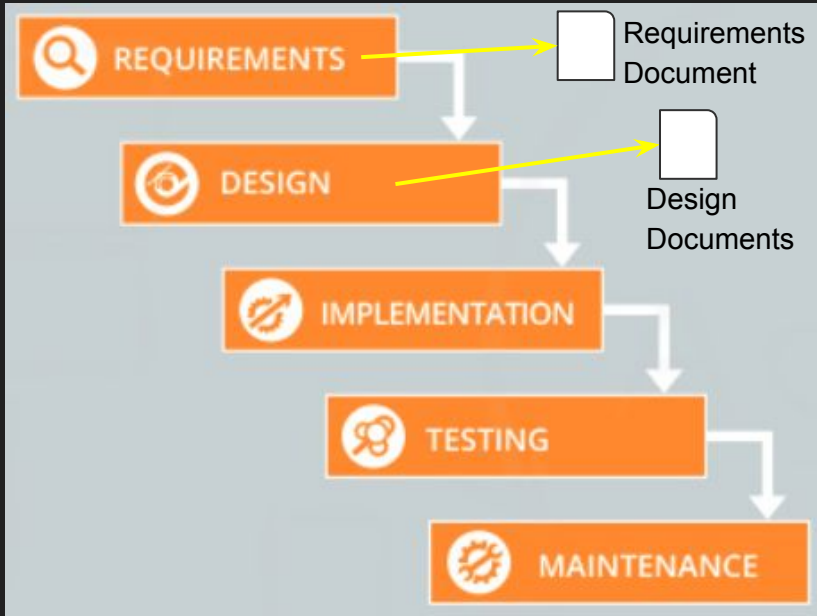
Each step of the process yields artifacts that are documents.

Documents produced during one step are needed for the next step and possibly for later steps.

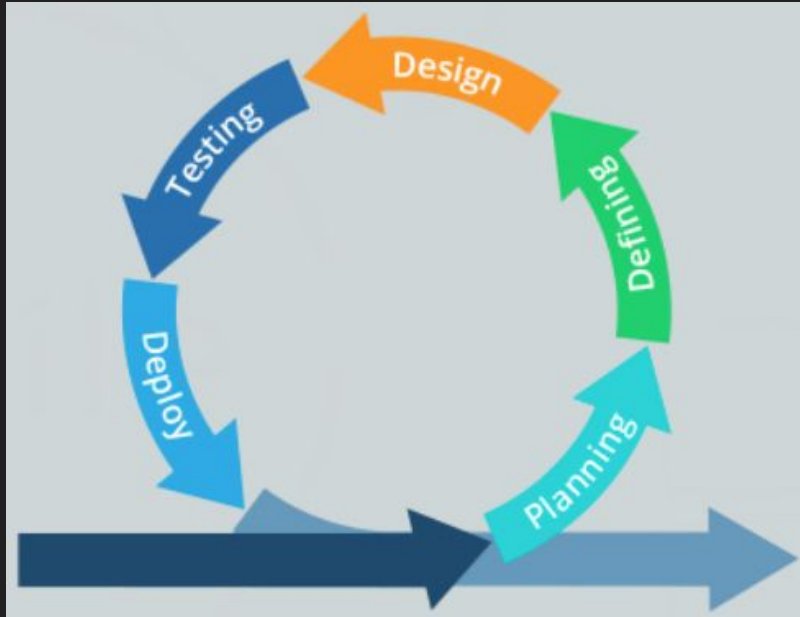
Managers like the waterfall model because progress is observable and measurable.

Knowledge is lost if team members leave before the project is completed, and it may be difficult for a project to recover from the loss. That is the reason why documents are important.

However, it has repeatedly proven to be very easy to produce impressive documents that eventually prove to be incomplete, inconsistent, hard to consult, or otherwise worthless.



Agile Development Process



Manifesto for Agile Software Development

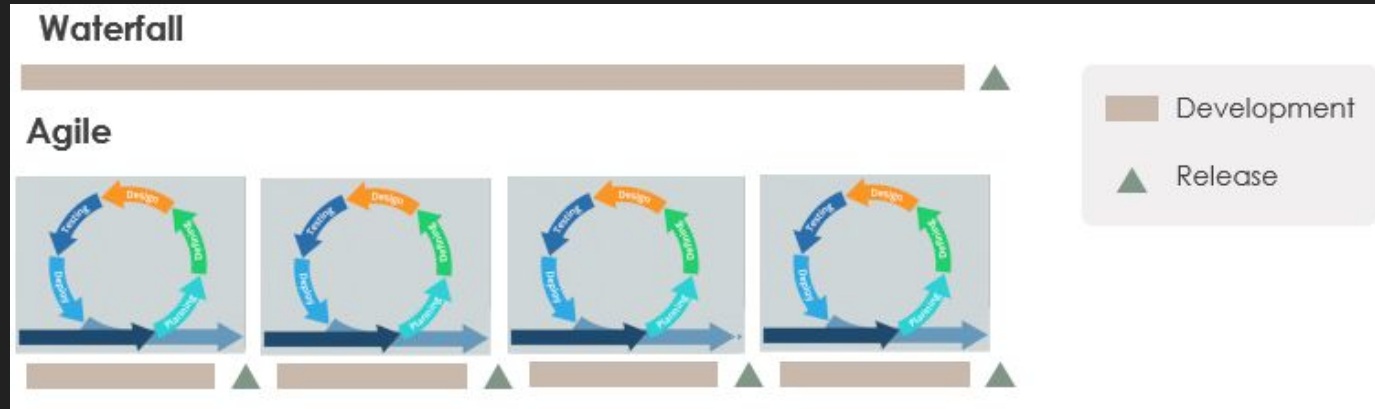
Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

Waterfall vs Agile



Agile characteristics:

- Complex work is divided in simple pieces
- Large organizations are divided into small teams
- Far-reaching project are divided into short time lists of task called sprints

What about your project?

Does it have any kind of documentation?

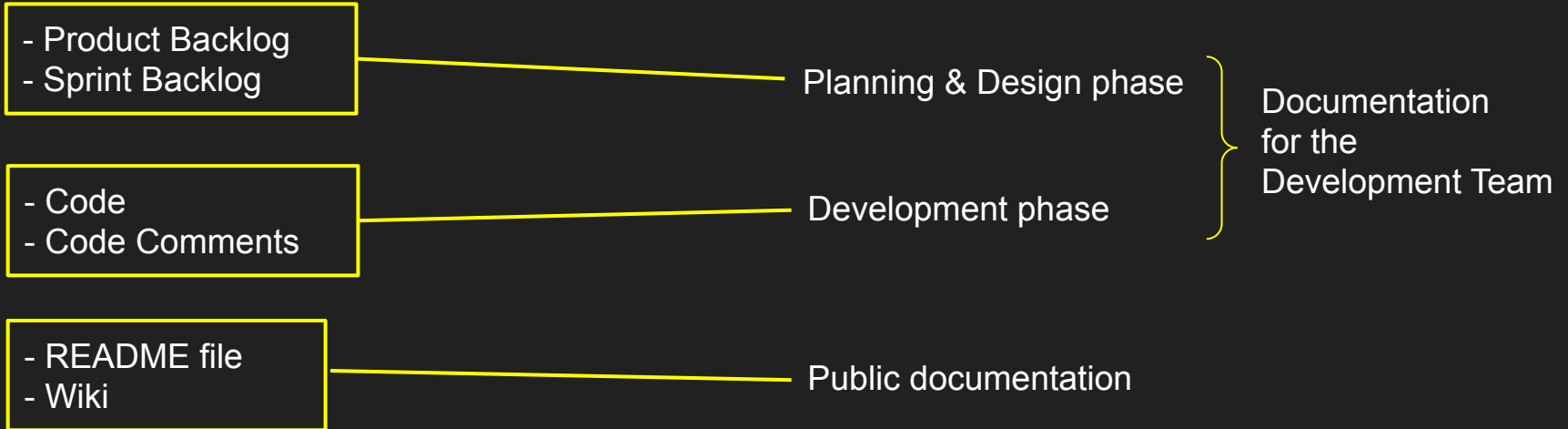
SCRUM Artifacts

The SCRUM artifacts are used to help define the workload coming into the team and currently being worked upon the team.

The main artifacts:

- Product backlog - a collection of user stories which present functionalities required/wanted by the product team. Usually the product owner takes responsible for this list.
- Sprint backlog - a collection of stories which could be included in the current sprint.

Which kind of documentation in a Agile/Scrum project?



README

You can add a README file to a repository to communicate important information about your project. A README, along with a repository license, contribution guidelines, and a code of conduct, communicates expectations for your project and helps you manage contributions

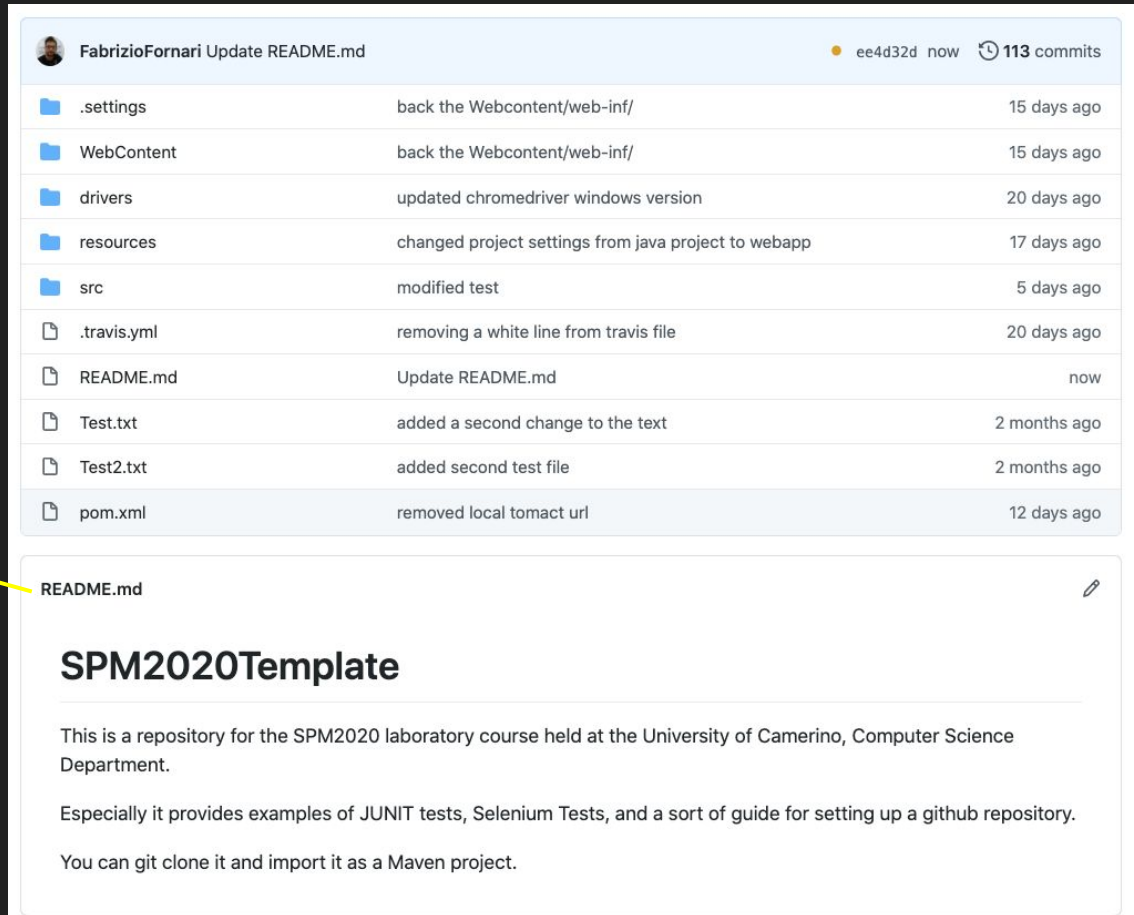
A README is often the first item a visitor will see when visiting your repository. README files typically include information on:

- What the project does
- Why the project is useful
- How users can get started with the project
- Where users can get help with your project
- Who maintains and contributes to the project

If you put your README file in your repository's root, `docs`, or hidden `.github` directory, GitHub will recognize and automatically surface your README to repository visitors.

README

README file



The screenshot shows a GitHub repository page for 'FabrizioFornari Update README.md'. It features a commit history table with columns for file/folder names, commit messages, and commit dates. The 'README.md' file is highlighted in the commit history. Below the table, the content of the README.md file is displayed, starting with the title 'SPM2020Template' and a description of the repository's purpose for a laboratory course at the University of Camerino.

File/Folder	Commit Message	Commit Date
.settings	back the Webcontent/web-inf/	15 days ago
WebContent	back the Webcontent/web-inf/	15 days ago
drivers	updated chromedriver windows version	20 days ago
resources	changed project settings from java project to webapp	17 days ago
src	modified test	5 days ago
.travis.yml	removing a white line from travis file	20 days ago
README.md	Update README.md	now
Test.txt	added a second change to the text	2 months ago
Test2.txt	added second test file	2 months ago
pom.xml	removed local tomact url	12 days ago

README.md

SPM2020Template

This is a repository for the SPM2020 laboratory course held at the University of Camerino, Computer Science Department.

Especially it provides examples of JUNIT tests, Selenium Tests, and a sort of guide for setting up a github repository.

You can git clone it and import it as a Maven project.

Github - Wiki

Every GitHub repository comes equipped with a section for hosting documentation, called a **wiki**. We can use our repository's wiki to share long-form content about our project, such as how to use it, how we designed it, or its core principles. We can use a wiki to provide additional documentation.

If you create a wiki in a public repository, the wiki is available to the public. If you create a wiki in an internal or private repository, people with access to the repository can also access the wiki.

You can edit wikis directly on GitHub, or you can edit wiki files locally. By default, only people with write access to your repository can make changes to wikis, although you can allow everyone on GitHub to contribute to a wiki in a public repository.

Cloning wikis to your computer

```
$ git clone https://github.com/YOUR_USERNAME/YOUR_REPOSITORY.wiki.git  
# Clones the wiki locally
```

Github - Wiki








FabrizioFornari / SPM2020Template Unwatch 1

<> Code Issues 4 Pull requests Actions Projects 3 **Wiki** Security Insights Settings

Create new page

Home

Write Preview

h1 h2 h3   **B** *i* <>      Edit mode: Markdown

Welcome to the SPM2020Template wiki!

Edit message

Write a small message here explaining this change. (Optional)

Save Page

Github - Wiki Permissions

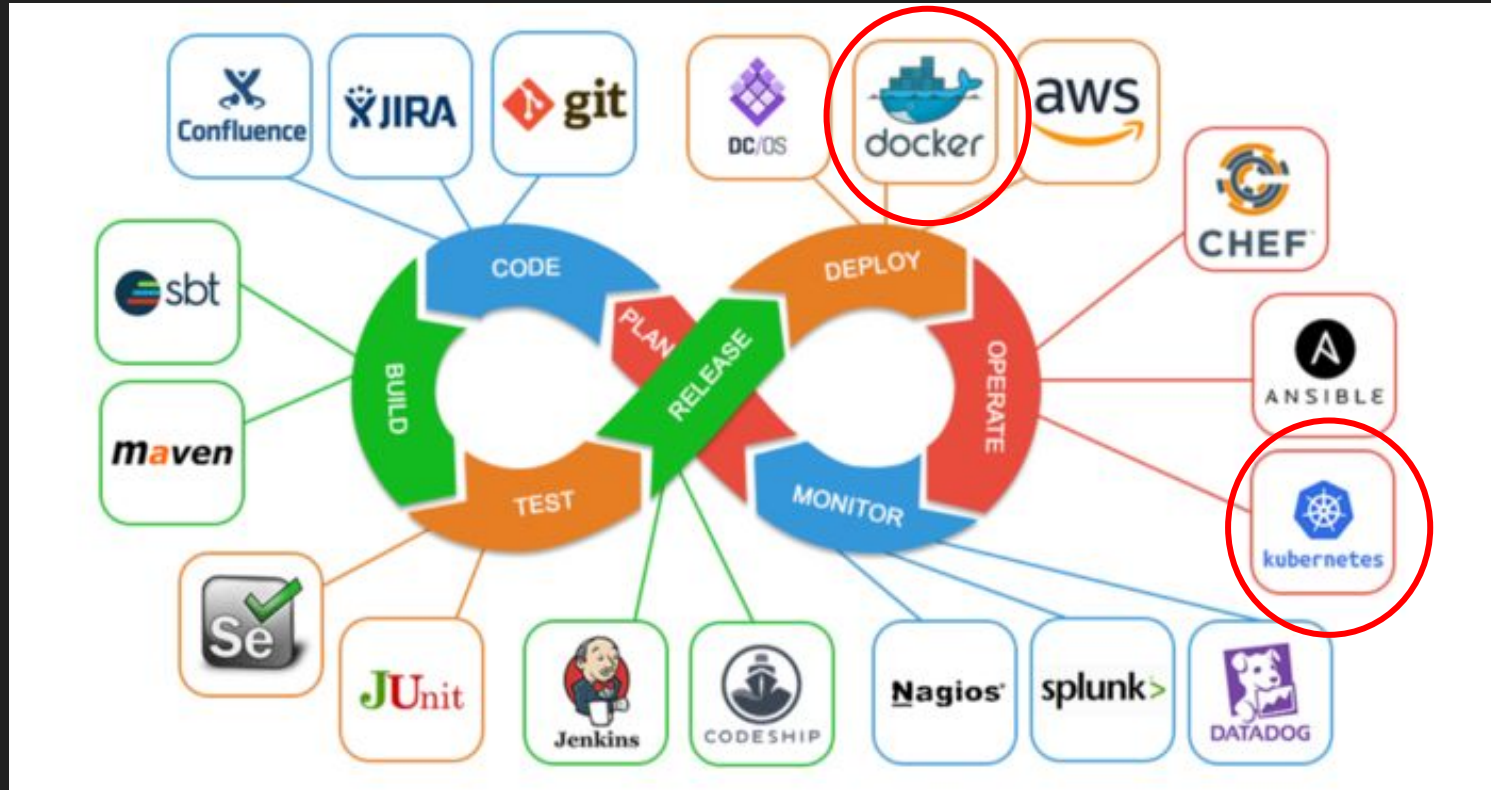
Features

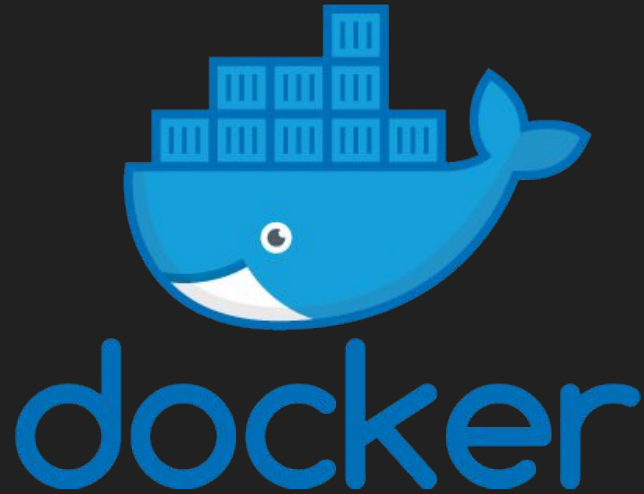
- Wikis
- Restrict editing to collaborators only**
Public wikis will still be readable by everyone.

Looking ahead...



DevOps Technologies





<https://www.docker.com/>

Docker is an open platform for developing, shipping, and running applications.

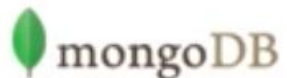
Docker enables you to separate your applications from your infrastructure so you can deliver software quickly.

With Docker, you can manage your infrastructure in the same ways you manage your applications.

Web Server



Database



Messaging



Orchestration



Libraries

Dependencies

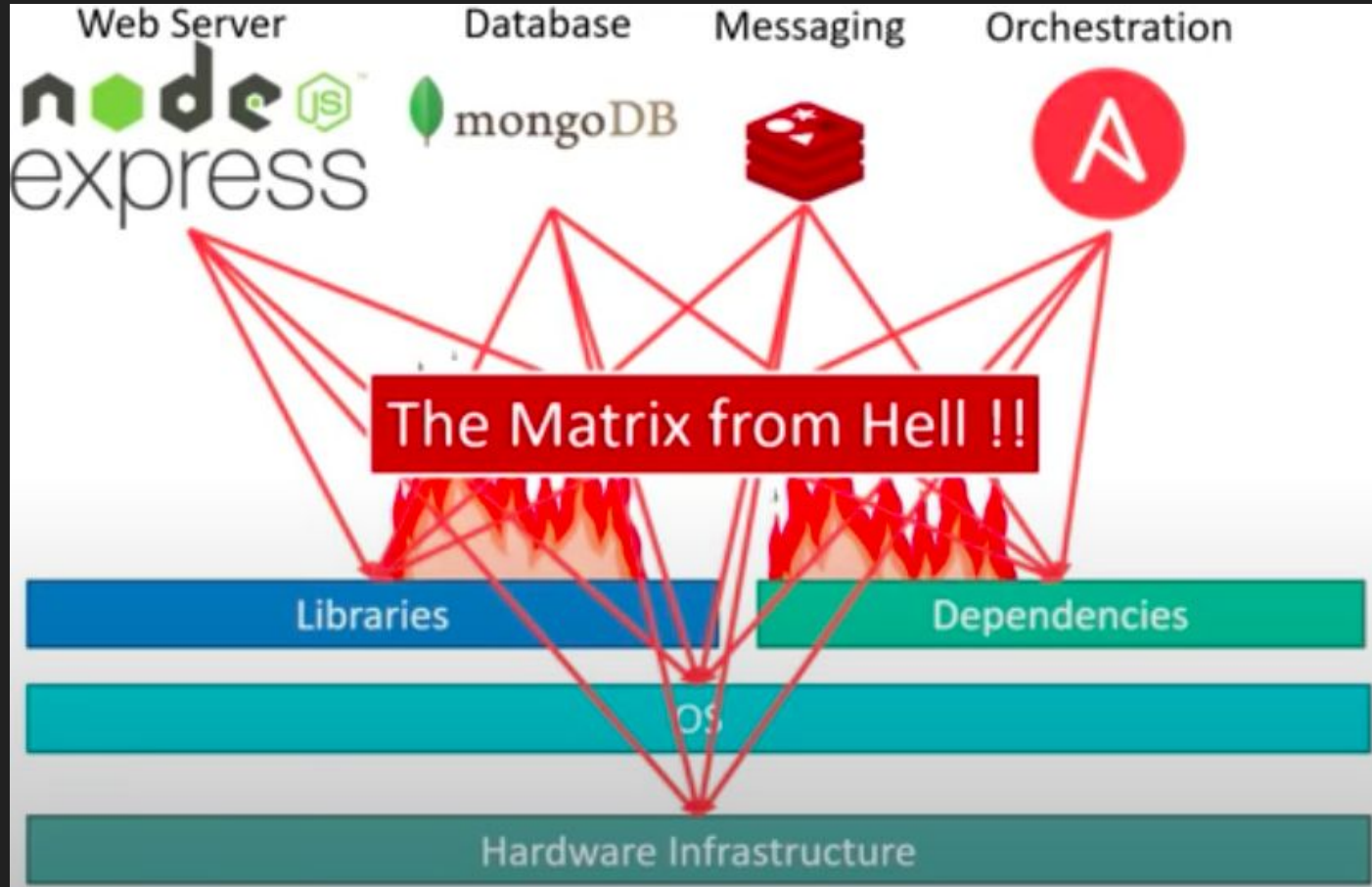
OS

Hardware Infrastructure

Compatibility/
Dependency

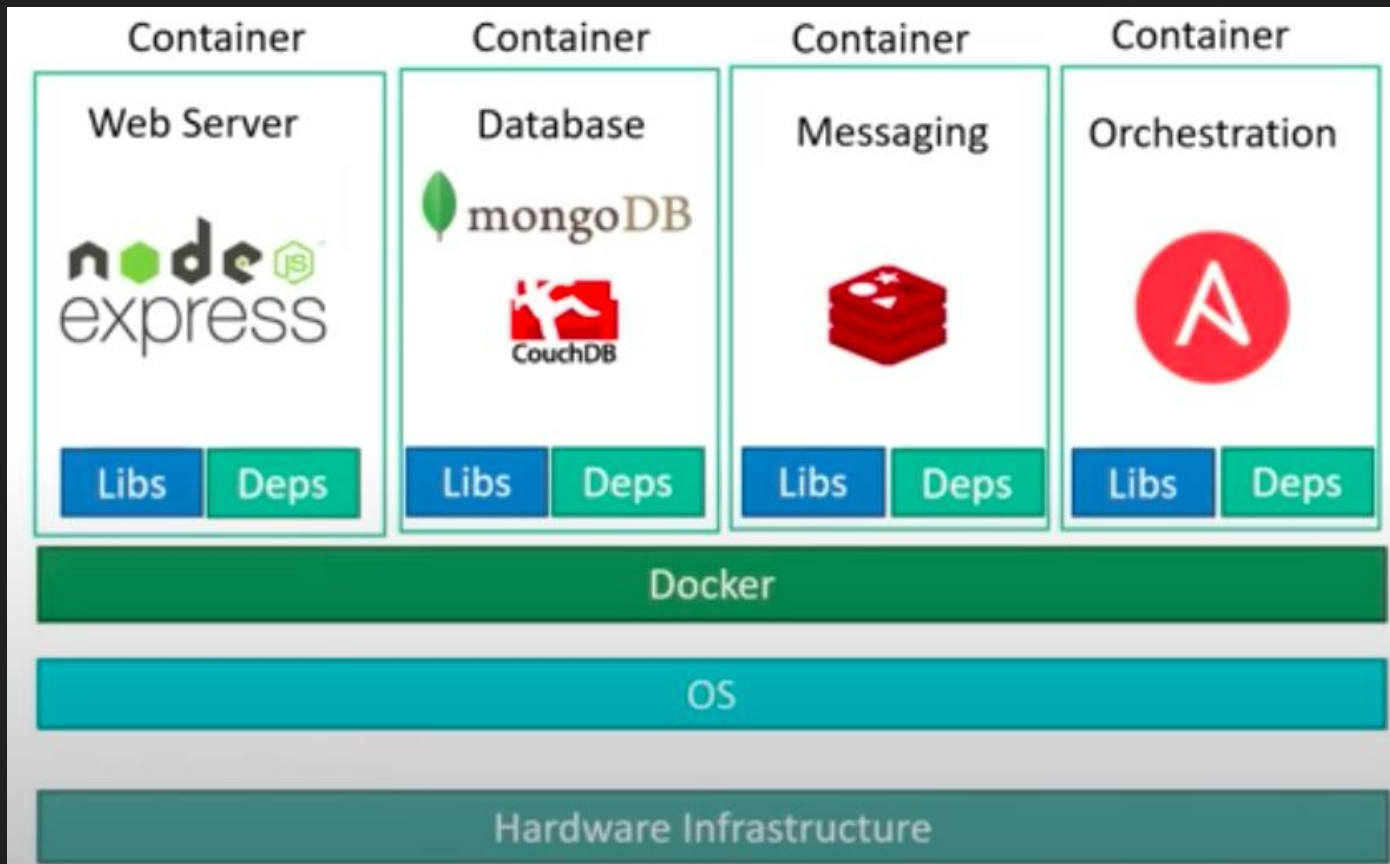
Long setup
time

Different
Dev/Test/Prod
environments



Containerized Application

Run each service with its own dependencies in separate containers



From Application to Container



Developer



App.war



Guide



Operations



From Application to Container



Developer



App.war



DockerFile



Docker Image



Operations

It Fixes the traditional “but it works on my machine”

From Application to Container



Developer



Docker Image



Operations

It Fixes the traditional “but it works on my machine”

Public Docker Images Repository



<https://hub.docker.com/>

Jenkins + Docker



docker

jenkins

...so a Docker Host



What if a Docker Host fails?



Orchestrating Hosts



Orchestration technology focuses on clustering and managing containers and hosts.

Docker Swarm: Easy to setup but lacks autoscaling

Kubernetes: from Google, difficult to setup but supports many advanced features, all public cloud supports it

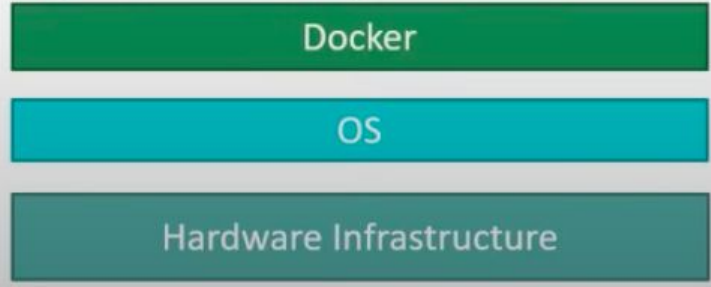
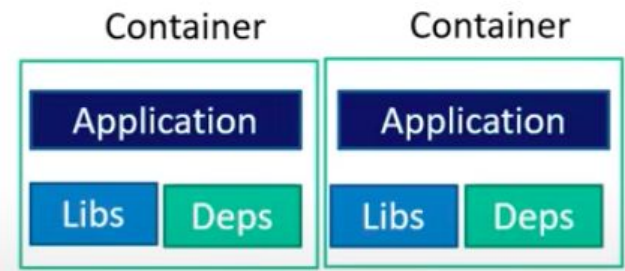
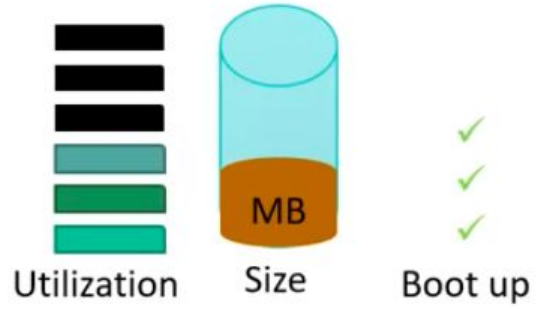
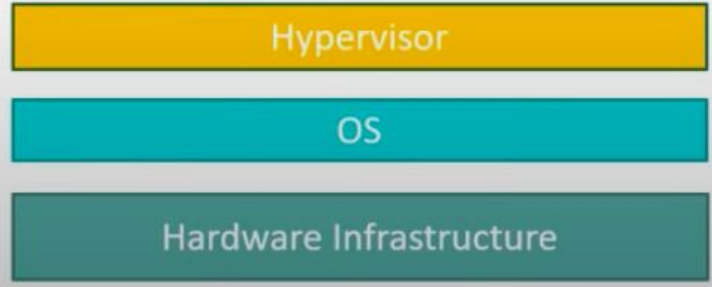
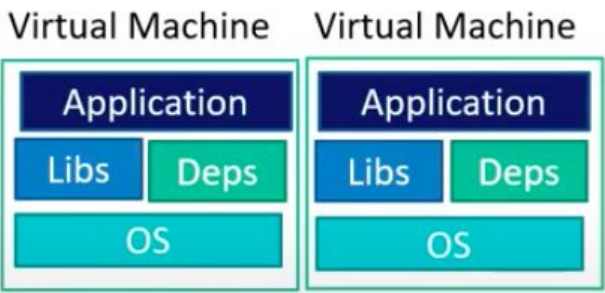
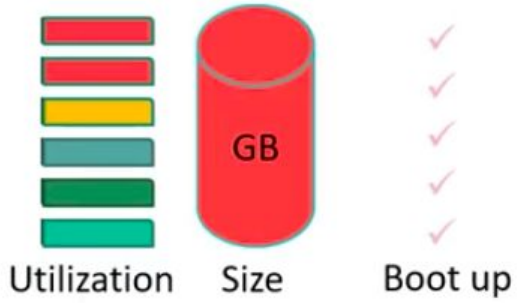
MESOS: from Apache, difficult to setup but supports many advanced features,

Kubernetes



A fundamental difference between Kubernetes and Docker is that Kubernetes is meant to run across a cluster while Docker runs on a single node. Kubernetes is more extensive than Docker Swarm and is meant to coordinate clusters of nodes at scale in production in an efficient manner.

VMs vs Containers



What's next?

Date	Topic
18/12/2020	Project Status Check with Groups
07/01/2020	Review of the Entire Course
08/01/2020	Sprint Review / Final Project Discussion

Course Overview

Course Objective

The course introduced the students to the basic knowledge of complex software system production following the **DevOps methodology**.

Learning Outcome

The student will be able to manage the organization and the development of a software applying DevOps methodology.

Overview

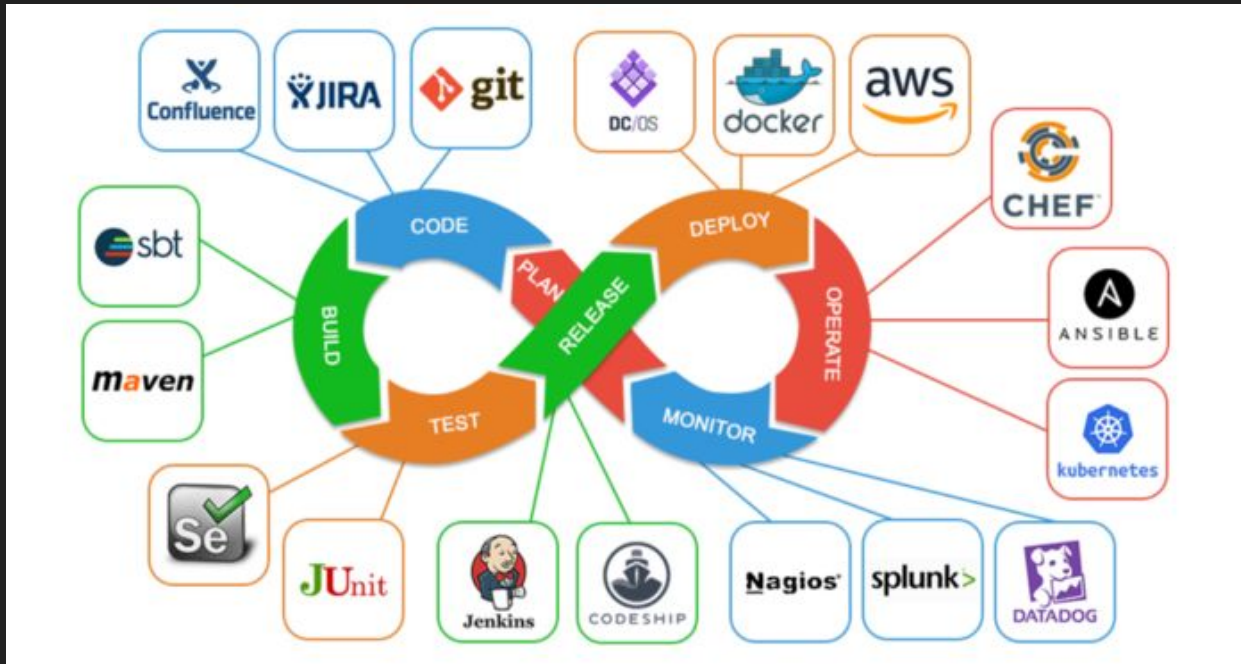
Website:

- [http://didattica.cs.unicam.it/doku.php?id=didattica:magistrale:spm:ay_2021:main&s\[\]=spm](http://didattica.cs.unicam.it/doku.php?id=didattica:magistrale:spm:ay_2021:main&s[]=spm)

You can contact me for any question related to the course: fabrizio.fornari@unicam.it

Note: only email coming from the @studenti.unicam.it domain will be processed.

What's next?



My Topics

Business Process Management
Business Process Modeling and Verification
Business Process and IoT
Process Mining
Software Project Management

Tools:

- BProVe, Business Process Verifier
- BEBoP, understandaBility vErifier for Business Process models
- RePROSitory, Repository of open PROcess models



Group Projects or Thesis

I supervise group projects and experimental thesis.

I try to apply together with the students the methodology and tools that we have seen during the course.

You can contact me for any question related to the course and for additional information about projects and thesis:

fabrizio.fornari@unicam.it

Note: only email coming from the @studenti.unicam.it domain will be processed.

