

Lectures Notes on Cloud Computing

Camilo de Lellis

October 22, 2025

Contents

1 Aula - 09/09/2025	2
1.1 Ementa da Disciplina	2
1.2 Histórico e Contextualização dos sistemas de computação em Nuvem.	3
2 Aula - 10/09/2025	4
2.1 Aula 02: - Surgimento dos Containers -Orquestração de Containers -Kubernetes	4
3 Aula - 16/09/2025	4
3.1 Introdução aos tipos de nuvens e Benefícios, desafios e riscos das plataformas de serviços.	4
4 Aula - 17/09/2025	4
4.1 SAAS, IAAS e PAAS - Introdução a Containers	4
5 Aula - 23/09/2025	4
5.1 Operações com container;	4
6 Aula - 24/09/2025	4
6.1 Operações com container.	4
7 Aula - 30/09/2025	4
7.1 Configurações de limites no container docker.	4
8 Aula - 07/10/2025	4
8.1 Configuração de recursos de hardware para conteineres.	5
9 Aula - 08/10/2025	5
9.1 Volumes em docker	5
10 Aula - 14/10/2025	5
10.1 Configuração de volume reutilizável com read-only em docker.	5
11 Exam - 21/10/2025	5
11.1 Question 6	5
12 Lecture - 22/10/2025	5
12.1 Question 6	6
12.2 How to create a new Docker environment	6
12.3 FROM command	6
12.4 RUN command	6
12.5 ENTRYPOINT command	6
12.6 CMD command	6
12.7 Multi-stage builds	7
12.8 Docker container commit	7
12.9 Best practices	7
12.10 Images management commands	7

1 Aula - 09/09/2025

Conteúdos ministrados: Aula 00 - Apresentação da disciplina; - Aula 01 - Histórico e Contextualização dos sistemas de computação em Nuvem.

1.1 Ementa da Disciplina

Curso: Curso Superior de Tecnologia em Sistemas para Internet **Disciplina:** Desenvolvimento Web para a Nuvem **Carga-Horária:** 60h (80h/a) **Pré-Requisito(s):** Desenvolvimento Web Back-end **Número de créditos:** 4

EMENTA

Conhecer o estado da arte sobre desenvolvimento web para a nuvem.

PROGRAMA

Objetivos

- Aprender sobre os modelos de computação em nuvem;
 - Conhecer os desafios do ambiente em nuvem;
 - Conhecer cenários de utilização em ambientes em nuvem;
 - Conhecer e desenvolver softwares como serviço.

Bases Científico-Tecnológicas (Conteúdos)

- 1. Princípios
 - 1.1. Histórico e Contextualização dos sistemas de computação em Nuvem;
 - 1.2. Introdução aos tipos de nuvens;
 - 1.3. Benefícios, desafios e riscos das plataformas de serviços;
 - 1.4. Cenários de Utilização;
 - 1.5. Modelos de negócios aplicáveis às nuvens.
 - 2. Modelos de Computação em Nuvem
 - 2.1. Software como serviço (SaaS);
 - 2.2. Infraestrutura como serviço (IaaS);
 - 2.3. Plataforma como serviço (PaaS);
 - 2.4. Middlewares para computação em nuvem.
 - 3. Configuração
 - 3.1. Administração e Regras;
 - 3.2. Escalonamento;
 - 3.3. Balanceamento de recursos em computação nas nuvens.
 - 4. Desafios de Programação para Computação em Nuvem
 - 4.1. Segurança;
 - 4.2. Privacidade;
 - 4.3. Legado

- 5. Gerenciamento de Dados e Otimização
 - 5.1. Gerenciamento de dados e desafios de manutenção nos sistemas de computação nas nuvens;
 - 5.2. Visão geral de técnicas de otimização incluindo o gerenciamento de consumo de energia elétrica.
- 6. Migração e Transformação de Servidores
 - 6.1. Migração e transformação de servidores para provedores de nuvem;
 - 6.2. Desafios na área de descoberta do ambiente fonte;
 - 6.3. Definição de ambiente destino;
 - 6.4. Decisões de estratégias de migração e transformação.

Procedimentos Metodológicos

Aulas expositivas; estudos dirigidos; seminários; vídeos; dinâmicas de grupo; visitas técnicas; palestras.

Recursos Didáticos

Quadro branco e pincel; computador; internet; projetor de multimídia.

Avaliação

Trabalho escrito; apresentação de seminários; relatórios; avaliação escrita.

Bibliografia Básica

- 1. ERL, Thomas. Cloud Computing: Concepts, Technology & Architecture. Editora Prentice Hall. 2013.
- 2. VELTE, Anthony T. Cloud Computing. Computação Em Nuvem: Uma Abordagem Prática. Alta Books. 2012.
- 3. FOX, Armando; PATTERSON, David. Construindo Software como Serviço (SaaS): Uma Abordagem Ágil Usando Computação em Nuvem (Portuguese Edition). Editora Strawberry Canyon LLC. 2015.

Bibliografia Complementar

- 1. BRIAN, J.S. Chee; FRANKLIN, Jr., Curtis. Computação em Nuvem: Cloud Computing - Tecnologias e Estratégias. 1. ed. M.Books. 2013.
- 2. Above the Clouds: A Berkeley View of Cloud Computing. Relatório Técnico. 2009.
- 3. BIRMAN, Kenneth. Guide to Reliable Distributed Systems: Building High-Assurance Applications and Cloud-Hosted Services. Springer. 2012.
- 4. VERAS, Manoel. Computação em Nuvem: Nova Arquitetura de TI. 1. ed. 2015.
- 5. KAVIS, Michael J. Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS). Editora Wiley. 2014.

Software(s) de Apoio:

- IDEs.

1.2 Histórico e Contextualização dos sistemas de computação em Nuvem.

...

2 Aula - 10/09/2025

Conteúdos ministrados: Aula 02: - Surgimento dos Containers -Orquestração de Containers -Kubernetes

2.1 Aula 02: - Surgimento dos Containers -Orquestração de Containers -Kubernetes

...

3 Aula - 16/09/2025

Conteúdos ministrados: Introdução aos tipos de nuvens e Benefícios, desafios e riscos das plataformas de serviços.

3.1 Introdução aos tipos de nuvens e Benefícios, desafios e riscos das plataformas de serviços.

...

4 Aula - 17/09/2025

Conteúdos ministrados: SAAS, IAAS e PAAS - Introdução a Containers

4.1 SAAS, IAAS e PAAS - Introdução a Containers

...

5 Aula - 23/09/2025

Conteúdos ministrados: Operações com container;

5.1 Operações com container;

...

6 Aula - 24/09/2025

Conteúdos ministrados: Operações com container.

6.1 Operações com container.

...

7 Aula - 30/09/2025

Conteúdos ministrados: Configurações de limites no container docker.

7.1 Configurações de limites no container docker.

...

8 Aula - 07/10/2025

Conteúdos ministrados: Configuração de recursos de hardware para conteineres.

8.1 Configuração de recursos de hardware para conteineres.

...

9 Aula - 08/10/2025

Conteúdos ministrados: Volumes em docker

9.1 Volumes em docker

...

10 Aula - 14/10/2025

Conteúdos ministrados: Configuração de volume reutilizável com read-only em docker.

10.1 Configuração de volume reutilizável com read-only em docker.

...

11 Exam - 21/10/2025

Content of the exam: docker concepts and fundamentals.

11.1 Question 6

We were presented a broken dockerfile. To create the fixed one, the following command was used:

```
touch Dockerfile && echo "FROM nginx:latest
COPY ./sites /usr/share/nginx/html
EXPOSE 80" >> Dockerfile
```

Before building the image and running the container, we need to create the folder "sites" and a file to change the **nginx** default **index.html**. This was done:

```
mkdir sites
touch sites/index.html
echo "testando" >> sites/index.html
```

To build the image:

```
docker build -t exam:latest .
```

To run a container:

```
# You can also assign a name to it using --name
docker run -ti -d -p 8080:80 exam:latest
```

To get the default page and check it's contents:

```
wget localhost:8080
cat index.html
```

12 Lecture - 22/10/2025

Content taught: correction of the exam, creating and managing images.

12.1 Question 6

The wrong Dockerfile was:

```
# Broken Dockerfile
FROM nginx:latest

# Copying HTML files to a folder
COPY ./site /usr/share/nginx/html_site

# Default port to Nginx
EXPOSE 80
```

To run the code wehad to change some things. The first thing to change about the dockerfile was the source directory `site` that did not exist from the first place. Then we had to change the target folder. The default nginx folder was `/usr/share/nginx/html`. So the correct dockerfile would be something like this:

```
# Fixed Dockerfile
FROM nginx:latest

# Copying HTML files to the correct folder
COPY ./site /usr/share/nginx/html

# Default port to Nginx
EXPOSE 80
```

12.2 How to create a new Docker environment

The first step to create a environment inside a docker container, is to create a Dockerfile. We can then build the Dockerfile to create an image.

After the image is built, we can run a new container using our newly made image.

Example dockerfile:

```
FROM debian

RUN echo "hello, world"
```

To build it, just run:

```
# Inside the folder where the Dockerfile is
docker build -t <image-name>:<version-name> .
```

12.3 FROM command

Using the `FROM command` you can select which image you can base your Dockerfile upon.

12.4 RUN command

The `RUN command` lets you execute bash commands when creating the image while reading the Dockerfile.

12.5 ENTRYPOINT command

The `ENTRYPOINT command` let we use an executable when initializing the image.

12.6 CMD command

We can use `CMD command` to when passing parameters to the ENTRYPOINT command.

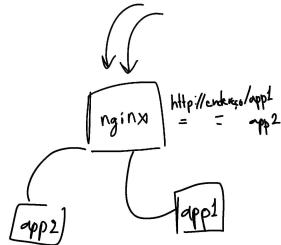


Figure 1: Draft of the use of nginx as a reverse proxy.

12.7 Multi-stage builds

[Multi-stage builds...](#)

12.8 Docker container commit

[Docker container commits](#) seem to me like version control for docker containers, meaning that you can change the image and have a version controller that remembers the last version.

12.9 Best practices

1. Minimal base images
2. Minimal layers
3. Cleaning after the packages installations
4. Non-root app running

12.10 Images management commands

```
docker image ls
docker image rm
docker prune
docker tag
```

12.11 Network management

Container's network management.

References