**O problema:** Quando pensamos em montar um ambiente de desenvolvimento logo pensamos em quais programas vamos precisar instalar em nosso computador, seja um estudante iniciando sua carreira ou um programador em uma grande equipe, logo fica a expectativa de como fazer da maneira mais simples e funcional possível, em nosso caso, precisamos de um ambiente para desenvolvimento web com apache, php, mysql e phpmyadmin, comunicando-se entre si e configurados de forma a aceitar a Reescrita de URL (URL amigável). Basicamente é preciso instalar três programas distintos e configurá-los manualmente para se obter o resultado esperado. Este processo pode muitas vezes ser trabalhoso, não apenas para iniciantes, quem já tem experiência com isso e teve algum problema em seu computador de trabalho, pode ter complicações para retomar as atividades. Muitas variáveis podem existir neste processo, o sistema operacional é uma delas, pois cada sistema tem sua maneira de fazer a instalação e configuração. Hoje a maioria dos cursos on-line é comum o professor disponibilizar um link no github com o ambiente em docker configurado para ser utilizado nas aulas, tornando um ambiente padronizado para todos os alunos evitando perca de tempo desnecessária para montar o ambiente.

**Uma idéia:** Criar um ambiente virtualizado em Docker, utilizando containers distintos representando os programas, mantendo configurações e padrões em qualquer sistema operacional que seja utilizado.

**O que é o Docker?:** Docker é uma engine de administração de containers, é um serviço que gerencia os containers, o container por rua vez é um processo isolado do sistema operacional, onde existe um sistema básico, enxuto, onde é possível rodar as aplicações desejadas sem interferir no ambiente host.

**Virtualização / Containers:** O Docker é um tipo de virtualização mas não uma virtualização convencional como estamos acostumados. Na imagem podemos observar que a virtualização exige mais recursos que os containers para serem executar as aplicações.

Hypervisor é o software de gerenciamento da máquina virtual, (VirtualBox, VirtualMachine, etc). Guest Host é o sistema operacional completo instalado dentro do disco da máquina virtual. Já os containers trabalham de forma mais simples, compartilhando recursos do próprio sistema operacional independente de qual seja. Desta forma o container necessita de menos recursos de hardware para ser executado. Os containers são baseados em Linux, LCX, é possível encontrar containers baseados em várias distribuições, (Debian, Ubuntu, Fedora) porém todos em Linux, não existe containers baseados em Windows ou MacOS.

O container pode conter isolamento de recursos, como limitar a quantidade de memória, processamento e rede que ele pode consumir da máquina host.

**Receita do bolo:** O que vamos precisar para criar nosso ambiente, basicamente o Docker, no sistema operacional Windows e Mac é preciso baixar o instalador, no sistema Linux é instalado via terminal “sudo apt install Docker.io”. Também vamos instalar o Docker-compose que é um orquestrador que auxilia o Docker a gerenciar vários containers de uma vez, sendo mais fácil de utiliza-los, nos sistemas Windows e Mac o compose já vem dentro do instalador, no Linux é preciso instalar separadamente, “sudo apt install docker-compose”. Em um diretório vamos criar uma pasta chama MYAPP e um arquivo docker-compose.yml,

**Docker-compose.yml:** este é o arquivo que vai definir as configurações de cada container e passar para o Docker executar as ações necessárias, criamos uma pasta e dentro o arquivo chamado docker-compose.yml, nele a identação é fundamental, através dela que é definido como cada comando deve ser interpretado, basicamente cada serviço é um container, dentro de cada container a image que vai utilizar como base, environment que são variáveis globais que podem ser definidas, ports onde é possível criar um redirecionamento ou não, volumes onde os arquivos locais são espelhados para um diretório definido dentro do container e links que conecta um container com outro de forma simples.

**Fazendo a mágica:** dentro do diretório com o arquivo docker-compose.yml em um terminal rodar o comando docker-compose up –d, o arquivo vai ser interpretado pelo compose que passa para o Docker os comandos, o sistema vai verificar se necessita baixar alguma imagem do repositório na nuvem e subir os containers como descrito no arquivo.

**Xablau:** seu servidor web com php mysql e phpmyadmin está no ar podendo ser acessado pela url localhot e localhost:8080.