ADministração e Gestão de Redes e Sistemas

Instalação de Serviços

[Escreva a síntese do documento aqui. Normalmente, a síntese é um breve resumo do conteúdo do documento. Escreva a síntese do documento aqui. Normalmente, a síntese é um breve resumo do conteúdo do documento.]



Índice

[Índice de Figuras 2](#_Toc439991146)

[Lista de Acrónimos 3](#_Toc439991147)

[Capítulo 1: Introdução 4](#_Toc439991148)

[Capítulo 2: Cenário e configuração da Rede 5](#_Toc439991149)

[Capítulo 3: Configuração de Serviços 7](#_Toc439991150)

[3.1. Serviços Pressupostos 8](#_Toc439991151)

[3.1.1 – Serviço DNS 8](#_Toc439991152)

[3.1.1.1 – Sobre o serviço 8](#_Toc439991153)

[3.1.1.2 – Funcionamento prático do serviço 8](#_Toc439991154)

[3.1.1.3 – Configuração do serviço 9](#_Toc439991155)

[3.1.2 – Serviço NFS 14](#_Toc439991156)

[3.1.3 – Serviço NTP 18](#_Toc439991157)

[3.1.4 – Serviço SSH 21](#_Toc439991158)

[3.2. Serviços Estudados 23](#_Toc439991159)

[3.2.1 – Configuração do Apache 24](#_Toc439991160)

[3.2.2 – Configuração do HAProxy 26](#_Toc439991161)

[3.2.3 – Configuração do PostgreSQL e PHP5 29](#_Toc439991162)

[3.2.4 – Configuração do Moodle e GIT 32](#_Toc439991163)

[3.2.5 – Configuração do Webmin 36](#_Toc439991164)

[Capítulo 4: Scripts 38](#_Toc439991165)

[Capítulo 5: Conclusão 39](#_Toc439991166)

[Capítulo 6: Anexos 40](#_Toc439991167)

[6.1 – Serviços Pressupostos 40](#_Toc439991168)

[6.1.1 – DNS 40](#_Toc439991169)

[6.2 – Serviços Estudados 42](#_Toc439991170)

[6.2.3 Instalação do Moodle 42](#_Toc439991171)

[Bibliografia 47](#_Toc439991172)

Índice de Figuras

**No table of figures entries found.**

# Lista de Acrónimos

DHCP - Dynamic Host Configuration Protocol

DNS - Domain Name Service

GPG - GNU Privacy Guard

IP - Internet Protocol

LTS - Long Time Support

NFS - Network File System

NTP - Network Time Protocol

PHP - Hypertext Preprocessor

RAID - Redundant Array of Independent Drives

RAM - Random Access Memory

SSH - Secure Shell

WWW - World Wide Web

Capítulo 1: Introdução

Este trabalho consiste na implementação de um um conjunto de serviços pressupostos, lecionados na mesma Unidade Curricular, em conjun de serviços estudados

A utilização do serviço *Bind* foi fortemente influenciada pela documentação experiência anterior

A adoção do NFS nas máquinas foi levada a cabo pela simplicidade de configuração e

Motivos levados à implementação do *HAProxy*

A escolha do Moodle 3.0 recaiu no facto de ser um dos mais populares serviços de ensino na atualidade. A instalação do mesmo, via git, permite que obtenhamos o mais recente repositório de ficheiros criados pela Moodle.org, sendo facilitada a sua instalação e configuração em sistemas Linux/Unix.

Para além de estar referido na escolha do tema para este trabalho, a escolha assente na base de dados PostgreSQL permitiu alargar os nossos horizontes a nível do conhecimento de bases de dados. Anteriormente já tínhamos trabalhado com MySQL e apesar de ser uma das bases de dados recomendadas pelo Moodle, optámos pelo PostgreSQL para desenvolver a nossa intelectualidade.

A nível de interfaces gráficas para facilitar a manutenção do sistema, o Webmin foi a nossa escolha. A sua instalação é muito simples e ao acrescentarmos o repositório do Webmin ao utilitário APT, possibilitou, um acesso ainda mais rápido e fluído ao pacote de instalação.

Capítulo 2: Cenário e configuração da Rede

(tenho de editar isto bem, colei só como já tinha)

Este trabalho de investigação visa implementar um cenário aplicável a um estabelecimento de ensino, no qual se verifica uma típica configuração de serviços na infraestrutura de rede que garanta o alojamento de páginas Web pertencentes à instituição, bem como a integração de uma plataforma de ensino para facilitar o acesso a documentos, informações sobre unidades curriculares e intercâmbio de contacto entre membros registados, com registo de informações como dados de acesso de utilizadores através de software de gestão de bases de dados.

Recorrendo ao conceito de virtualização por software (execução de máquinas virtuais a partir de uma física, através da partilha de recursos de hardware), foi configurada uma máquina virtual para cada máquina física (no total três máquinas físicas e três virtuais), simulando o funcionamento de instâncias completamente independente, seja ou não com sistemas operativos idênticos.

Foi estabelecida a utilização de uma máquina virtual por cada máquina física,

(falta o diagrama com exemplificação gráfica da situação)

Garantir um fornecimento contínuo de serviços através de uma máquina central que redireciona pedidos (?de forma alternada entre dois servidores com os mesmos serviços configurados, garantindo balanceamento de carga bem como ?) uma continuidade de serviço em caso , assegurando alta disponibilidade através de um segundo meio de acesso em caso de falhas ou negação de serviço.

Tenciona-se projetar um acesso transparente por parte dos acessos de clientes, isto é, serão omitidas possíveis alternações entre os servidores que atendem os pedidos Web (na prática, a página devolvida será sempre semelhante), transmitindo a perspetiva de um ambiente direto e sempre disponível

cluster: nó central, interligado às restantes máquinas implementadas.

servidor principal que suporta os serviços essenciais que asseguram o

moodle servers - suportam serviços moodle (Apache, Moodle, PostgreSQL)

periódica sincronização via rSync

Todas máquinas utilizam Ubuntu Server 14.04.3 LTS e possuem serviços em comum, um cliente NTP e acesso NFS, e a virtualização aloca um total de 500 MB de memória RAM em cada máquina.

(Situação não é realista; num cenário prático duma empresa, há sempre distribuição dos conteúdos disponibilizados em conjuntos de discos (RAID - Redundant Array of Independent Drives)

Capítulo 3: Configuração de Serviços

## 3.1. Serviços Pressupostos

### 3.1.1 – Serviço DNS

### 3.1.1.1 – Sobre o serviço

Ao efetuar ligação a qualquer página da Internet, ou a configurar dispositivos numa rede de média ou larga escala, é impensável a necessidade de memorizar todos os endereços de recursos com que se quer comunicar. Esta dificuldade gerou necessidade em desenvolver o DNS (*Domain Name Service* - Serviço de Nomes de Domínio), que consiste num serviço de resolução de nomes em endereços IP implementado em servidor. Este pode também funcionar de forma inversa, em que a partir de um endereço IP se faz uma pesquisa por nomes associados (chamada de pesquisa inversa - *reverse* *lookup*).

Neste trabalho de investigação, foi implementado um domínio local que permite um acesso facilitado à plataforma *Moodle* do sistema, denominado “**agrsmoodle.pt**”, com respetivas associações de nomes de máquinas aos respetivos endereços IP.

### 3.1.1.2 – Funcionamento prático do serviço

O DNS consiste na atribuição de um nome ao endereço IP de uma máquina (por exemplo, é possível aceder alternativamente à página da *Google* Portugal “www.google.pt”, através da ligação direta ao endereço de um dos servidores da *Google* que responde a este pedido de acesso, como o 216.58.210.99).

Um servidor de DNS pode também em qualquer momento passar simultaneamente por cliente, visto que conforme disponibiliza resoluções de nomes, pode também fazer consultas de nomes ou endereços, questionando outros servidores externos. Exemplos de aplicações de cliente DNS são o ***Dig*** (em *Linux*), ***nslookup*** (*Windows* e *Linux*), ou **qualquer** **navegador da Internet**.

É possível verificar no *Ubuntu* a que servidor DNS uma máquina está a recorrer, através da visualização do ficheiro ***“/etc/resolv.conf***”, no campo "*nameserver*" (servidor de nomes).

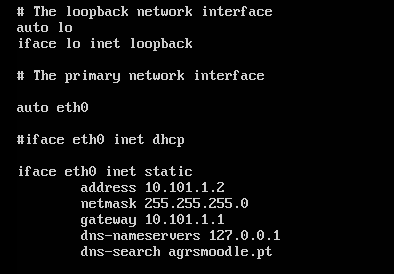
### 3.1.1.3 – Configuração do serviço

De forma a configurar um servidor DNS estático, será necessário instalar *software* apropriado. Em sistemas operativos baseados em *Unix*, por norma recorre-se ao *Bind*, sendo gratuito e a solução mais popular para este fim. A sua versão atual é a 9.10.3, lançada a 16 de setembro de 2015.

Para instalar o *Bind*, recorre-se ao comando:

sudo apt-get install bind9

Antes de iniciar a sua configuração, é recomendável a definição de endereços estáticos nas máquinas da rede interna, editando o ficheiro das *interfaces* de rede, localizado em **“/etc/network/interfaces”:**



* O campo “***address*”** deverá ser alterado conforme o endereço IP pretendido;
* No campo “***netmask***” deverá ser inserida a máscara de rede correspondente à rede;
* No campo “***gateway***” deverá ser inserido o endereço IP do *gateway* da rede;
* No campo “***dns-nameservers***” deve ser colocado 127.0.0.1 no caso do servidor DNS (este endereço que indica a própria máquina, o *localhost*), e o endereço IP do servidor DNS nas restantes máquinas;
* No campo “***dns-search***” insere-se o domínio privado implementado com o *Bind*, neste caso “agrsmoodle.pt”.

Para finalizar este passo, deve-se reiniciar o serviço de rede e forçar uma nova atribuição de endereço IP, recorrendo aos comandos:

* ***sudo /etc/init.d/networking restart***
* ***sudo ifdown eth0***
* ***sudo ifup eth0***

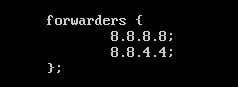
Inicia-se então à configuração do *Bind*. Os seus ficheiros de configuração estão localizados na diretoria “**/etc/bind**”, que dispõe dos seguintes ficheiros:

* **db.** - Ficheiros predefinidos nomeadamente destinados à configuração de zonas, como o “db.0” ou “db.255” que parametrizam uma zona de *broadcast* privada;
* **named.** – Definem os principais parâmetros de funcionamento do *bind*. Distribui-se por:
  + **named.conf** - ficheiro que indica que ficheiros de configuração o *bind* deverá utilizar;
  + **named.conf.default-zones** - contém as zonas predefinidas do *bind*;
  + **named.conf.options** - destinado principalmente à definição de *forwarders*, isto é, encaminhadores alternativos que o servidor deverá utilizar para respostas não-autoritárias (pedidos de resolução externos a outros servidores na *Internet*, caso o servidor não consiga fazer a resolução de um nome)
  + **named.conf.local** - utilizado para parametrizar zonas locais.

**Configuração de named.conf.options**

Neste ficheiro deve ser retirado o comentário à estrutura da diretiva "*forwarders*", e adicionar servidores DNS alternativos, um por linha, que serão utilizados para respostas não-autoritárias.

Neste cenário, optou-se por definir os servidores DNS da *Google*, resultando no seguinte formato:



**Criação do domínio local agrsmoodle.pt**

De seguida, será necessário criar o domínio local. Este procedimento foi aprofundado para implementar o domínio **agrsmoodle.pt**, que permite um acesso facilitado por parte de um cliente na rede à plataforma *Moodle* sem necessidade de memorizar o endereço IP do servidor *Web*.

Para tal, é recomendável **criar primeiro uma diretoria adicional** dentro da pasta do *bind*, denominada por exemplo "*zones*", onde serão alojados os ficheiros de configuração do domínio local da zona, utilizando o comando:

sudo mkdir /etc/bind/zones

Para criar a zona **agrsmoodle.pt**, utiliza-se como modelo de estrutura o ficheiro predefinido "*db.local*", copiando-o para a pasta "*zones*" com o novo nome:

sudo cp /etc/bind/db.local /etc/bind/zones/db.agrsmoodle.pt

**Configuração da zona de DNS (db.agrsmoodle.pt)**

É no ficheiro acabado de criar que será de seguida realizada a configuração de nomes das máquinas da zona.

Na **segunda** linha será necessário substituir os campos "*localhost*." pelo nome da máquina do domínio que configura o DNS, e o campo seguinte por um possível endereço de contacto de *e-mail* do administrador do domínio, resultando numa linha como:

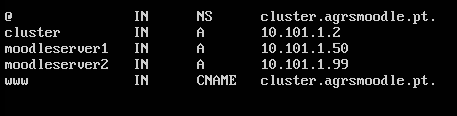
https://i.gyazo.com/54e500d006c483922525d0333d69b461.png

Alguns parâmetros gerais da zona que podem ser configurados são:

* **Serial**- versão do ficheiro, geralmente alterado conforme mudanças nos valores do ficheiro;
* **Refresh** - tempo entre sincronizações da zona;
* **Retry** - em caso de falha, tempo de espera até tentar novas sincronizações;
* **Expire** - tempo total permitido para pedidos;
* **Negative cache TTL** - TTL (*Time to Live* – número máximo de saltos entre nós até descartar um pacote), para configuração de servidores DNS de *cache*.

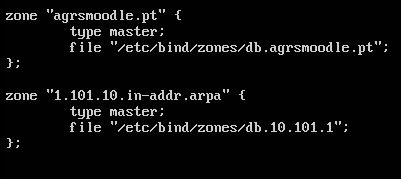
No final do documento, colocam-se todas as máquinas e respetivos parâmetros, atribuindo as máquinas aos encargos e nomes correspondentes. Pode ser utilizada a seguinte nomenclatura:

* **NS** - Servidor que responde pela zona do *domínio;*
* **A** - IPv4 *Address* (usado para destinar nomes a endereços IPv4);
* **AAAA** - IPv6 *Address* (usado para destinar nomes a endereços IPv6);
* **CNAME** - Usado na atribuição de nomes alternativos a máquinas do domínio;
* **MX** - Servidores responsáveis pelo intercâmbio de *mail*;
* **PTR** - *Pointer* (apontador utilizado nas resoluções inversas de DNS);



(Nos anexos será possível verificar a versão final implementada no cenário.)

**Configuração de named.conf.local**

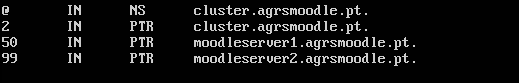
 Neste ficheiro de configuração será incluída a zona acabada de criar bem como a zona de resolução inversa que será criada a seguir(na zona inversa os octetos do endereço IP são escritos do fim para o início):

**Configuração da zona inversa (db.10.101.1)**

A zona inversa é definida da mesma forma que a anterior, exceto que estará localizada num ficheiro à parte, utilizando como modelo o ficheiro “*db.127*”. Copia-se, alterando o nome para os octetos do endereço IP da rede:

sudo cp /etc/bind/db.127 /etc/bind/zones/db.10.101.1

Segue-se o mesmo procedimento que no passo “Configuração da zona de DNS“, exceto que na primeira coluna se coloca apenas o octeto do *host* das máquinas, e se usa a nomenclatura “PTR”.



Para finalizar, reinicia-se o serviço *bind* (“***sudo service bind9 restart***”) nas máquinas afetadas. Pode ser testado o seu devido funcionamento, utilizando os seguintes comandos:

* **“Dig** (endereço IP do servidor)”
* **“Dig** **-x** (nome da máquina do servidor)”

Uma correta configuração devolverá o nome da máquina no primeiro caso e o endereço IP respetivo ao nome inserido no segundo caso.

### 3.1.2 – Serviço NFS

3.1.2.1 - Sobre o serviço

O NFS (*Network File System* – Sistema de Ficheiros Remoto) é um serviço implementado para sistemas baseados em *Unix* e destinado à partilha de partições através da rede. É destacado pela sua simplicidade na configuração, fiabilidade, e pelo facto de ser totalmente gratuito. A sua versão mais recente é o NFSv4.1, sendo estável e mais robusta.

Existem outros tipos semelhantes de *Software*, adotados consoante diferentes tipos de necessidades, como por exemplo o Samba, que já suporta caraterísticas de partilha entre diferentes tipos de sistemas operativos, notavelmente de *Linux* para *Windows*.

Este serviço foi implementado em todas as máquinas da rede, em que há transferência de *backups* de ficheiros de configuração de serviços dos servidores *Moodle* para a máquina de alojamento em rede (NAS) e também da máquina central do *cluster* para a mesma.

3.1.2.2 - Funcionamento prático do serviço

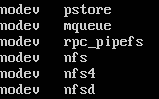
O NFS consiste em realizar a partilha de uma partição ou diretoria do disco rígido de uma máquina-servidor com outras máquinas, sendo destinado ao acesso via rede, em que o espaço é apenas ocupado no servidor. Visualmente, e na prática, o cliente acede a esta diretoria e interage com a mesma (conforme as permissões atribuídas) como se o recurso pertencesse ao seu próprio sistema operativo.

3.1.2.3 - Configuração do serviço

**Configuração do ambiente em servidor**

De forma a configurar um servidor NFS, será necessário em primeiro lugar instalar o pacote completo através do comando:

sudo apt-get install nfs-kernel-server

 Para confirmar a correta instalação, recorre-se ao comando ***“cat /proc/filesystems”***, verificando na lista resultante se os *filesystems* “**nfs**”, “**nfs4**“ e “**nfsd**” estão presentes.

Antes de iniciar as configurações de partilhas, caso ainda não exista a(s) diretoria(s) a partilhar, será necessário criá-la(s). Neste cenário, como se utiliza a NAS para alojar os *backups* de ficheiros de configuração das restantes três máquinas, foi planeada a criação de **uma diretoria principal** na raiz, contendo **três subdiretorias**, uma para cada das máquinas:

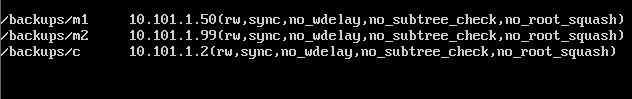
sudo mkdir -p /backups/m1

sudo mkdir /backups/m2

sudo mkdir /backups/c

Uma vez criadas, estas diretorias podem então ser definidas no ficheiro de diretorias exportadas pelo NFS, localizado em ***“/etc/exports***”, que serão posteriormente montadas a partir das máquinas-cliente. Deverá ser indicada uma diretoria por linha, assim como o endereço IP com quem a partilhar, junto com alguns parâmetros, tais como:

* **Ro -** Restringe as permissões do conteúdo da diretoria para apenas leitura;
* **Rw -** Restringe as permissões do conteúdo da diretoria para leitura e escrita;
* **Sync –** Otimiza as sincronizações com as máquinas a fazer partilha;
* **No\_wdelay –** Evita atrasos na escrita do disco. Requer a opção “*sync*” ativada;
* **No\_root\_squash –** Mantém permissões *root* ao exportar o conteúdo da diretoria para outra máquina.
* **No\_subtree\_check –** Desativa a validação de pesquisas em subdiretorias exportadas dentro de diretorias não exportadas pelo servidor NFS.

As seguintes configurações foram aplicadas no trabalho de investigação:

Para finalizar, atualizam-se as configurações utilizando o comando “*exportfs”* e reinicia-se o serviço NFS:

sudo exportfs -ra

sudo service nfs-kernel-server restart

É possível verificar as diretorias exportadas, pelo comando:

sudo exportfs -v

**Configuração do cliente NFS**

Para configurar um cliente NFS, será necessário instalar o pacote básico para garantir o acesso a um servidor:

sudo apt-get install nfs-common

(Ou, em vez do anterior, pode ser instalado de novo o pacote “*nfs-kernel-server*”, para também preparar a máquina para uma possível futura implementação como servidor)

Uma vez mais, pode ser efetuada a confirmação do suporte aos *filesystems* do NFS usando o comando "***cat /proc/filesystems***", verificando se estes estão presentes.

Daqui, resta criar uma diretoria-destino para receber a partilha:

sudo mkdir /nfs

https://i.gyazo.com/63c8f728f7362d3cb53f8cbedfe36e28.pngE, por fim, adicionar uma linha referente ao endereço IP e a partição exportada no ficheiro de montagem automática de *filesystems* aquando do arranque da máquina, localizado em ***“/etc/fstab”***, junto com as configurações-padrão da partilha:

Para garantir o acesso à diretoria, é necessário reiniciar a máquina, verificando posteriormente se esta se encontra presente no caminho definido.

Daqui, testa-se este serviço criando ficheiros de teste na diretoria, confirmando se estes se localizam em ambas as máquinas.

### 3.1.3 – Serviço NTP

3.1.3.1 – Sobre o serviço

O NTP (*Network Time Protocol*) consiste num protocolo desenvolvido para realizar sincronizações com alto nível de precisão da data e hora de um sistema a partir de um servidor. Este é realizado através da rede, sendo possível realizar pedidos a servidores configurados internamente (servidores privados), ou através da *Internet* (questionando servidores públicos).

O cenário implementado consiste na configuração de um servidor NTP privado na máquina central do *cluster*, que possui também funcionalidades de cliente, respondendo a pedidos deste tipo às outras máquinas existentes na rede e questionando esporadicamente dois servidores públicos, de forma a manter uma constante sincronização.

3.1.3.2 – Funcionamento prático do serviço

Na prática, existem **clientes** e **servidores NTP**: um cliente **apenas questiona** um servidor de forma a receber estas informações da forma mais precisa possível, realizando assim a sua própria sincronização, enquanto que um servidor as **disponibiliza** a outras máquinas, podendo **também opcionalmente recorrer** a funcionalidades de cliente, onde este acede a uma *pool* (conjuntos de servidores) pública.

3.1.3.3 – Configuração do serviço

**Configuração do servidor NTP**

Em primeiro lugar, será necessário adquirir o pacote e instalar o NTP, de forma a configurar o ficheiro do serviço:

sudo apt-get install ntp

Procede-se então à configuração do NTP, cujo ficheiro está localizado em ***“/etc/ntp.conf”***.

Aqui, no campo dos servidores NTP a utilizar serão acrescentadas duas linhas correspondentes aos servidores da *Internet* que a máquina utilizará para se sincronizar (é recomendável utilizar no mínimo dois servidores, de forma a obter um valor redundante e mais correto), que serão:

* **server 0.pt.ntp.pool.org**
* **server 2.pt.ntp.pool.org**

Outras duas linhas a acrescentar correspondem à configuração da máquina como servidor NTP, sendo estas:

* **server 127.127.1.0** - endereço interno que aponta para a própria máquina para se disponibilizar como servidor privado para outras máquinas na rede;
* **fudge 127.127.1.0 stratum 10** - garante que o servidor da máquina local se situa num nível de servidores NTP (*stratum*) menos prioritário que os públicos, para evitar conflitos.

Para terminar, comenta-se a linha **“*server ntp.ubuntu.com”*** (acrescentando um símbolo “***#***” atrás do seu começo) de forma a forçar o funcionamento do NTP com apenas os três servidores anteriormente especificados (nos anexos será possível observar a configuração implementada no trabalho de investigação).

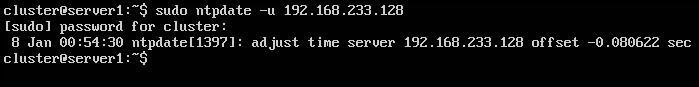
As alterações só serão aplicadas reiniciando o *daemon* (processo executado com o arranque da máquina que garante o funcionamento do serviço) do NTP, utilizando o comando:

sudo /etc/init.d/ntp restart

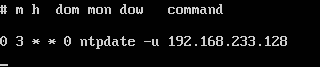
Para testar o funcionamento do serviço, pode-se optar por forçar uma sincronização a um dos servidores escolhidos, recorrendo ao comando “***ntpdate***” (com opção “**-u**” para atualizar durante a execução do processo NTP):

Esta mensagem indica que houve atualização com sucesso.

**Configuração do cliente NTP**

 Não existe qualquer configuração específica para efetuar pedidos como cliente - apenas será necessário realizar testes de ligação ao servidor NTP, tal como no passo anterior, recorrendo ao comando “***ntpdate****”*:

Tendo realizado uma correta sincronização, pode-se **definir um agendamento automático** da execução deste comando nas máquinas de forma personalizada, através da edição do ficheiro para tal destinado, o ***Cron***, com o comando ***“sudo crontab -e”***



Adiciona-se a última linha, que neste caso permitirá **realizar o comando de sincronização NTP todos os domingos, às 3 horas da manhã**.

(**Nota**: a configuração de um servidor NTP privado gerou problemas e falhas de sincronização, cuja investigação e *troubleshooting* levou à conclusão que a infraestrutura da rede onde se realizaram testes deste serviço impunha limitações ao seu funcionamento, tendo sido necessário recorrer a um acesso externo configurado com NAT, o que originou a utilização de uma diferente gama de endereços IP para verificar o funcionamento do serviço. Teoricamente, o NTP foi corretamente projetado e aplicado, porém a nível prático, poderão existir resultados inconclusivos consoante a rede onde se testa.)

### 3.1.4 – Serviço SSH

3.1.3.1 – Sobre o serviço

O SSH (*Secure Shell*) consiste num serviço que permite efetuar acessos a máquinas remotas por meio de um túnel cifrado, garantindo uma segurança reforçada contra possíveis ataques de furto ou manipulação de informações trocadas. Este serviço trabalha na porta 22 TCP.

3.1.3.2 – Funcionamento prático do serviço

Este serviço destina-se a efetuar ligações remotas, possibilitando trocas de dados de uma forma cifrada, protegendo-os. É uma potente ferramenta que facilita a administração de máquinas à distância, simultaneamente garantindo segurança nesta tarefa.

3.1.3.3 – Configuração do serviço

Para realizar a sua instalação recorre-se ao comando:

sudo apt-get install ssh

A partir daqui o serviço encontrar-se-á a funcionar, mas é recomendado que se proceda à sua configuração. Para isso teremos de aceder ao ficheiro de configuração ***“/etc/ssh/sshd\_config”***, onde será possível definir alguns parâmetros personalizados como:

* **AllowUsers \*@xxx.xxx.xxx.xxx** – para restringir o acesso a apenas algumas máquinas (substituir o último campo pelo endereço IP da máquina a permitir acesso; repetir o comando quantas vezes necessário, um por linha);
* **LoginGraceTime** **30** – para definir o tempo máximo permitido para fazer *login* (neste caso 30 segundos);
* **PermitRootLogin no** – para negar o acesso ao utilizador *root*.

Estas configurações serão aplicadas assim que o serviço for reiniciado com o comando:

sudo service ssh restart

**Acesso de clientes SSH**

No lado do cliente, existem duas principais formas de aceder a uma outra máquina por SSH: por linha de comandos, ou por método gráfico (exemplo: *software* putty).

Para aceder à *shell* por comando, utilizamos como exemplo:

* **ssh 'cluster@10.101.1.2'**

Aparecerá um aviso a indicar que a autenticidade da máquina-destino não pôde ser verificada, perguntando se pretende continuar mesmo assim; seleciona-se "*yes*". Com esta confirmação, será transmitida uma mensagem de aviso confirmando que a máquina indicada foi adicionada à lista de *hosts* conhecidos. Será pedida então a *password* do utilizador que se pretende conectar, que, após inserida corretamente, terá acesso à máquina.

Para aceder por GUI, por exemplo em *windows*, podemos utilizar o ***putty***. Apenas temos de inserir o endereço IP destino no campo ***"Host Name(or IP address)"*** e confirmar que se está a utilizar a porta 22.

Após clicar em “*Open”* será pedido o nome de utilizador e a respetiva palavra-passe, estabelecendo assim ligação.

(***Nota***: Se existir uma configuração por acesso restrito à máquina, é necessário certificar que o endereço IP foi previamente adicionado à lista de utilizadores permitidos.)

## 3.2. Serviços Estudados

### 3.2.1 – Configuração do Apache

Apache

O Apache é um *Software* destinado à implementação de um servidor onde seja possível criar, alojar e disponibilizar páginas *Web*. A sua estabilidade, fiabilidade e simplicidade, bem como a sua distribuição totalmente gratuita e de livre edição para qualquer programador (*Open-source Software*)tornou este a mais popular escolha a quem pretenda configurar um servidor *Web*.



Para proceder à sua instalação, será necessário recorrer ao comando:

sudo apt-get install apache2

Este permitirá adquirir os pacotes do *software* bem como todos os seus requisitos automaticamente, garantindo imediatamente o funcionamento do serviço. A instalação ocupará cerca de 5 MB de espaço no disco rígido.

A única configuração restante a realizar para este serviço será a alteração da diretoria raiz de acesso aos ficheiros da página principal do servidor, acedendo ao seu ficheiro de configuração (“/***etc/apache2/sites-available/000-default.conf***”), alterando a linha:

* **DocumentRoot /var/www/html**

Para:

* **DocumentRoot /var/www/html/moodle**

Termina-se, reiniciando o serviço *Apache*:

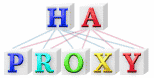
sudo service apache2 restart

### 3.2.2 – Configuração do HAProxy

Haproxy

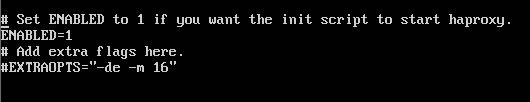
O *HAProxy* (*High* *Availability* Proxy – *Proxy* de alta disponibilidade) consiste num serviço gratuito implementado a partir de uma máquina central, ocupando o encargo de encaminhar o tráfego recebido de pedidos de máquinas entre um dado número de servidores.

Na prática, conforme as configurações estipuladas, este serviço carateriza-se por assegurar balanceamento de carga, sendo capaz de alternar acessos e comunicação entre diferentes servidores interligados, ou garantir alta disponibilidade, transferindo o funcionamento de um serviço para outra máquina em caso de falha com a principal, melhorando o *uptime* dum sistema.



Em primeiro lugar, será necessário adquirir o pacote e instalar o *HAProxy*, utilizando o comando:

sudo apt-get install haproxy

Antes de realizar posteriores configurações, é recomendável assegurar de imediato que este será iniciado automaticamente com o arranque da máquina, acedendo ao ficheiro *“****/etc/default/haproxy****”*, onde se altera o valor de “***ENABLED***” para ‘1’:

De seguida, procede-se à configuração do serviço, no ficheiro *“****/etc/haproxy/haproxy.cfg***”

O cenário implementado no trabalho de investigação visa adotar e assegurar um funcionamento contínuo do serviço *Web* que fornece a plataforma *Moodle* (método *failover)*, garantindo o *uptime* de um sistema em caso de falha ou negação de serviço no servidor principal.

Assim sendo, de uma forma geral foram mantidas as configurações-padrão do ficheiro, destacando-se a adição das seguintes linhas:

**Temporização de ligações:**

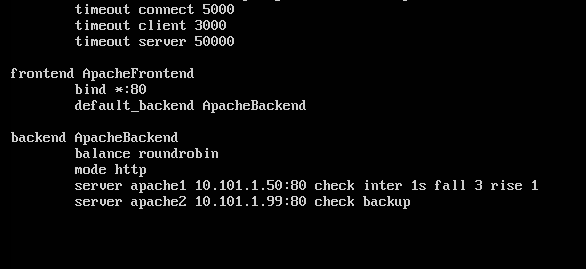
Os seguintes valores podem ser alterados consoante necessidade do administrador. Estes apenas são aplicáveis quando se configura o *HAProxy* em modo de balanceamento de carga.

* **Timeout connect** – tempo máximo para um cliente se tentar ligar ao servidor, até alternar de servidor que atende os pedidos;
* **Timeout client** – tempo máximo de inatividade por parte do cliente (troca de dados) até alternar o servidor atendedor dos pedidos;
* **Timeout server** – tempo máximo de inatividade por parte do servidor (troca de dados) até alternar o servidor atendedor dos pedidos;

**Configurações do funcionamento do serviço**

* **Frontend –** esta configuração define como os pedidos são encaminhados para os servidores definidos. Aqui deve-se integrar o porto do serviço pelo qual os servidores configurados estarão à escuta (no caso de servidores *Web*, o porto 80);

* **Backend –** esta diretiva permite estabelecer as configurações dos pedidos enviados aos servidores. Nesta secção, define-se principalmente o método de balanceamento de tráfego estabelecido, e a indicação das máquinas com que o *HAProxy* vai funcionar.

Efetuou-se a seguinte configuração neste ficheiro, de forma a permitir o funcionamento *Failover*:

* A opção “***balance roundrobin***” permite a alternação de pedidos entre as diferentes máquinas;
* A opção “***check***” confirma a comunicação (ou não) com o servidor listado;
* A opção “***inter***” define o intervalo entre cada tentativa de comunicação com o servidor (caso esta falhe);
* A opção “***fall***” consiste no número de tentativas de comunicação com o servidor (caso esta falhe uma vez), até o classificar como desligado e transferir o serviço para o servidor seguinte;
* A opção “***rise***” consiste no número de tentativas de comunicação com o primeiro servidor, caso haja recuperação de serviços, até a reclassificar como operacional;
* A opção “***backup***” é adicionada ao servidor destinado à recuperação de serviços, indicando que esta só será utilizada caso a primeira falhe.

O serviço estará pronto a utilizar, sendo apenas necessário reiniciar o serviço para aplicar as alterações:

sudo service haproxy restart

### 3.2.3 – Configuração do PostgreSQL e PHP5

PostgresqL E PHP5

PostgreSQL é uma poderosa ferramenta open source de gestão de base de dados. É uma das muitas alternativas ao MySQL.

Já o PHP, na sua versão 5 é uma linguagem de scripting utilizada em aplicações que utilizam bases de dados como MediaWiki, Facebook, Wordpress, Joomla, entre outras.



De modo a possibilitar uma correta configuração do Moodle, o servidor necessitará de ter instalado a linguagem de scripting PHP versão 5 e a base de dados PostgreSQL.

O sistema operativo necessitará das seguintes dependências de PHP5:

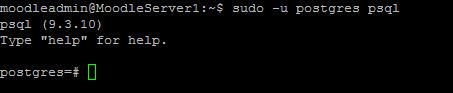
sudo apt-get install php5-fpm php-apc php5-curl php5-gd php5-xmlrpc php5-intl

Já a instalação do PostgreSQL necessitará destas:

sudo apt-get install postgresql postgresql-contrib php5-pgsql

Após a instalação, iniciamos a base de dados com o comando:

sudo -u postgres psql



O Administrador deve agora criar uma conta de utilizador dentro da prompt do postgres:

CREATE USER moodleadmin WITH PASSWORD ‘1’;

https://scontent-lhr3-1.xx.fbcdn.net/hphotos-xpf1/v/t34.0-12/12388256_1136322493059615_431579939_n.jpg?oh=0a2524e5ec0f8b2057b364f8fb4d6b3f&oe=568DCCA2

É criada agora a base de dados para o utilizador ‘moodleadmin’

É protegido o acesso à base de dados com uma password de admin, neste caso ‘1’.

sudo -u postgres psql template1

# ALTER USER postgres WITH PASSWORD ‘1’;

# \q

sudo -u postgres created -E utf8 -O moodleadmin moodle

Após a criação bem-sucedida do utilizador e da base de dados, o administrador pode agora sair da prompt do postgres com o comando \q + tecla ENTER.

Instalamos ainda algumas dependências restantes para o serviço:

sudo apt-get install graphviz aspell php5-pspell php5-curl php5-gd php5-intl php5-mysql php5-xmlrpc php5-ldap clamav

(A instalação do módulo php5-mysql é opcional).

Reinicia-se a base de dados:

sudo /etc/init.d/postgresql-x.x restart

(é inserido o valor da versão atual do postgresql que esteja a ser utilizada no lugar dos ‘x’ aqui representados).

Com a configuração inicial do PostgreSQL, do PHP5 e com a criação de um utilizador dá-se por concluída a instalação destes serviços.

Prossegue-se então à instalação do serviço de controlo Git para obter a versão mais recente do Moodle a partir dos repositórios oficiais no subcapítulo seguinte.

### 3.2.4 – Configuração do Moodle e GIT

Git e Moodle

Git é um sistema de controlo de versões focado na rapidez e na integração de dados. É um software Open Source.

O Moodle é um serviço Web criado a pensar nas necessidades dos estudantes e das entidades de Ensino. Pretende disponibilizar uma plataforma onde o conteúdo pode ser partilhado entre professores e alunos. O Moodle possibilita ainda a capacidade de ensino à distância, sendo possível a disponibilização de aulas e de testes na plataforma.



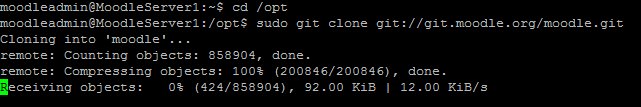
Para iniciar a instalação do software Git, procedemos à seguinte inserção na linha de comandos:

sudo apt-get install git-core

Após a instalação, prossegue-se à configuração do repositório local do administrador. Ao utilizar o Git está a facilitar o seu trabalho futuramente no que diz respeito à forma como atualiza a aplicação do Moodle.

Da localização onde está na bash, dirija-se à diretoria /opt através do comando:

cd /opt

Faça download do código fonte do Moodle e do seu índex através de uma clonagem aos repositórios oficiais do Moodle:

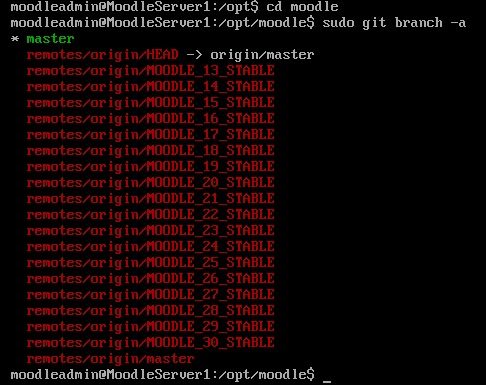
sudo git clone git://git.moodle.org/moodle.git

Mude de diretoria para a pasta onde transferiu o Moodle:

cd moodle

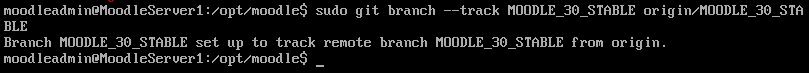
Faça uma recolha de todos os “branches” disponíveis no repositório:

sudo git branch -a



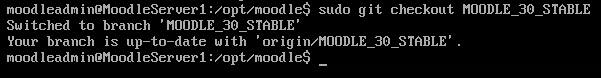
Diga ao Git que “branch” pretende utilizar (Neste caso foi utilizada a versão 3.0 do Moodle):

sudo git branch --track MOODLE\_30\_STABLE origin/MOODLE\_30\_STABLE

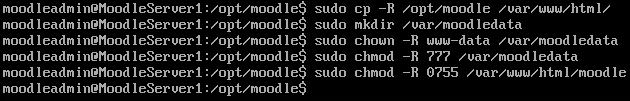


Verifique qual é a mais recente versão do Moodle:

sudo git checkout MOODLE\_30\_STABLE



Copie o repositório local para a diretoria /var/www/html/



Explicação sucinta: Como se configurou um repositório local no passo anterior, este irá ser copiado para o webroot após quaisquer updates ou alterações que tenham sido feitos. Ter o repositório local fora do webroot, tal como foi feito na diretoria /opt, permite que os upgrades ao Moodle sejam realizados de uma forma mais eficiente. Por exemplo, quando se instalam plugins, normalmente, é transferido o plugin e este é copiado para o repositório local do Moodle. Possivelmente, terá depois de editar o ficheiro localizado na diretoria /opt/moodle/.git/info/exclude. Dentro desse ficheiro, serão configuradas as pastas que devem ser excluídas pelo git quando este faz um “pull” aos updates do repositório assim que é efetuado o comando “sudo git pull”.

sudo chmod -R 0755 /var/www/html/moodle

7

sudo chmod -R 777 /var/moodledata

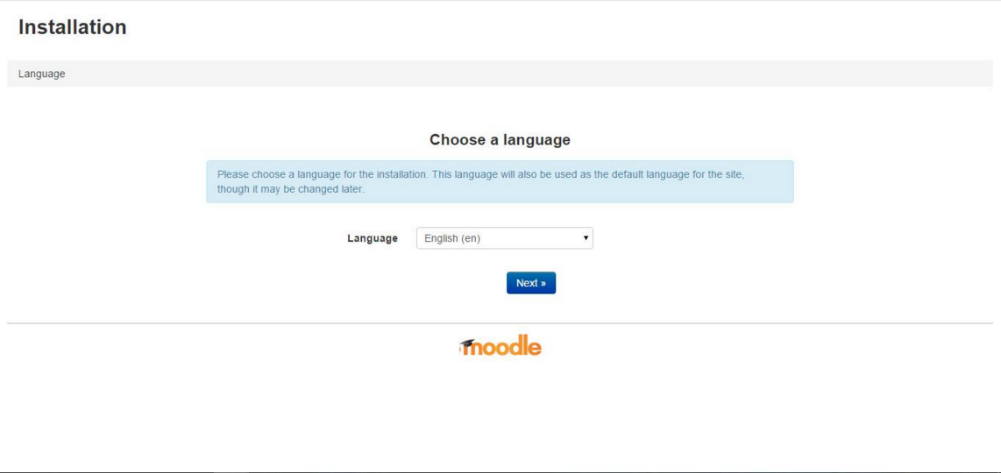
7

sudo chown -R www-data /var/moodledata

sudo mkdir /var/moodledata

sudo cp -R /opt/moodle /var/www/html/

Por fim, através de um browser é possível finalizar a instalação do Moodle 3.0 através do endereço http://localhost/moodle (em que o localhost é o ip definido na máquina do servidor). Após toda a instalação, **é necessário apontar os servidores *Moodle* para o domínio local criado**, alterando a linha *“****$CFG->wwwroot***” do ficheiro *“****/etc/www/html/moodle/config.php***” para o valor “http://www.agrsmoodle.pt”.



Para uma instalação detalhada, verifique o subcapítulo 6.1. (Anexos).

### 3.2.5 – Configuração do Webmin

Webmin

Webmin é uma ferramenta de administração gráfica web-based que facilita a administração de um sistema operativo Linux/Unix através de um simples browser.



A instalação do Webmin neste projeto foi realizada através de repositórios oficiais utilizando o comando apt. Para tal, foi necessário adicionar os repositórios à lista de repositórios do sistema operativo.

Adicionaram-se as seguintes linhas no fim do ficheiro:

deb http://download.webmin.com/download/repository sarge contrib

deb http://webmin.mirror.somersettechsolutions.co.uk/repository sarge contrib

Ainda dentro do editor nano, pressiona-se CTRL + X, Y e ENTER para salvar o ficheiro com as modificações.

sudo nano /etc/apt/sources.list

É necessário importar uma chave GPG pela qual este repositório se cinge. Para tal são utilizados os seguintes comandos:

wget http://www.webmin.com/jcameron-key.asc

sudo apt-key add jcameron-key.asc

Após a configuração dos repositórios, a instalação do webmin torna-se mais simples. A sua localização predefinida é /usr/share/webmin.

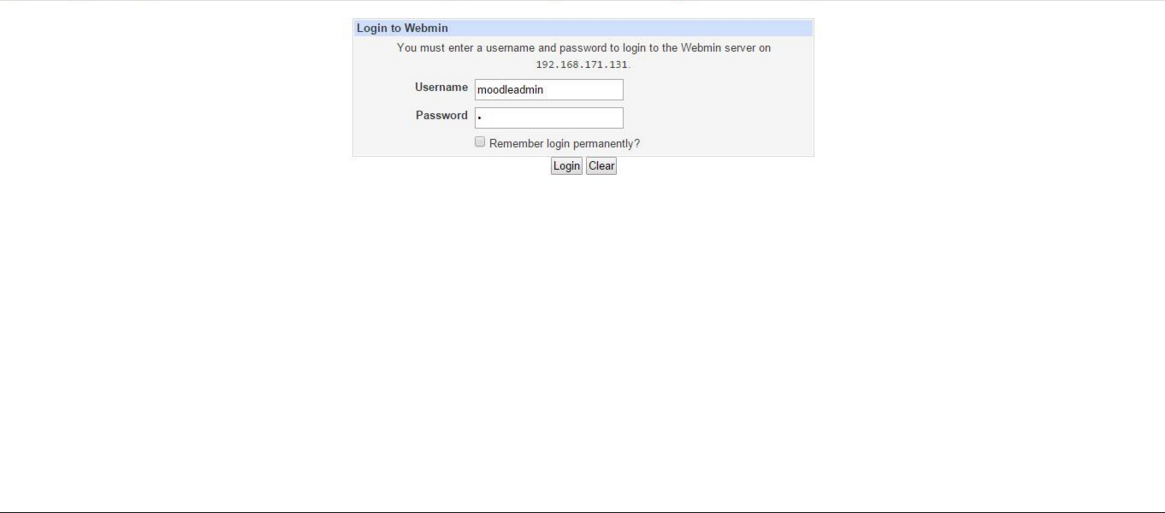
Com a colocação destes simples comandos, a instalação do webmin está concluída.

sudo apt-get install webmin

sudo apt-get update

O acesso à sua interface é feito por browser, acedendo ao site <https://localhost:10000> sendo que localhost é o IP da máquina e 10000 é o porto em que o serviço do webmin está à escuta.

As credenciais de acesso são as mesmas que estão configuradas para o sistema operativo.



# Capítulo 4: Scripts

# Capítulo 5: Conclusão

Dos objetivos que traçámos ao longo do período de concretização deste projeto, debruçámo-nos sobre diversos problemas, nomeadamente a nível da integridade das bases de dados PostgreSQL.

A instalação dos serviços essenciais foi realizada de forma bem-sucedida, sem grandes percalços, com uma especial exceção para o NTP, que apesar de estar bem configurado, não nos foi possível verificar a sua resolução nos computadores Cliente devido aos bloqueios impostos pelo IPLeiria.

No que toca aos restantes serviços, tentámos implementar sincronização nos dois servidores Moodle (ver diagrama) através do utilitário rSync. No entanto, após algum esforço, verificámos que essa não seria a melhor prática e com os consequentes problemas que tivemos a níveis de permissões, decidimos abandonar essa ideia e investigar a alternativa de criar um sistema de NFS na tentativa de recriar a prática que é realizada nos dias de hoje, de possibilitar um array de discos à parte de todo o restante equipamento informático.

Após sucessivas intervenções no nosso Moodle, verificámos que ao modificar o conteúdo das páginas numa máquina (Exemplo: MoodleServer1), a máquina MoodleServer2, ou vice-versa, não recebiam corretamente a sincronização das bases de dados PostgreSQL e chegava até a corromper os dados da base de dados do Moodle, sendo que o acesso aos dados era feito diretamente nos discos da máquina virtual que criámos para fazer de NAS.

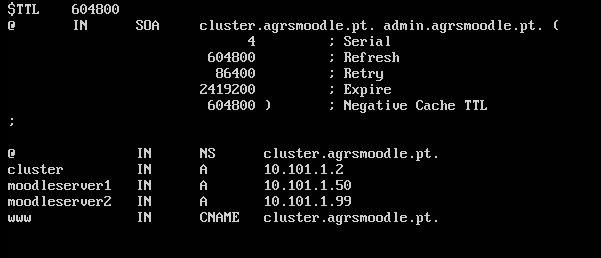
Preferimos, portanto, adotar ainda outra medida de acesso às bases do PostgreSQL, fazendo uma replicação das bases de dados da máquina MoodleServer1 e MoodleServer2 através do HAProxy, mas o que podia funcionar na teoria não se conseguiu adaptar na prática.

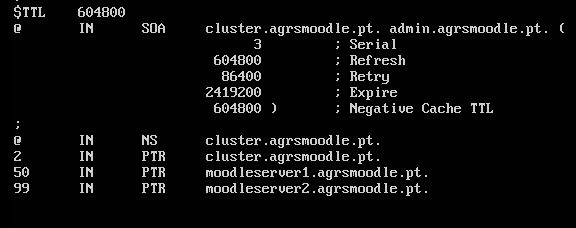
Até à data de escrita deste relatório, estamos de momento a identificar outras soluções para implementar a sincronização das bases de dados.

# Capítulo 6: Anexos

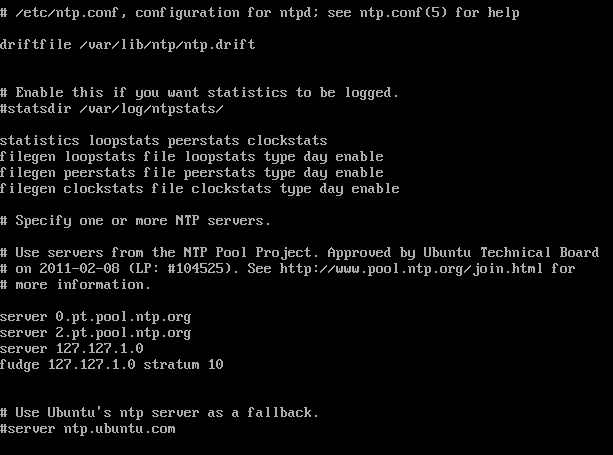
## 6.1 – Serviços Pressupostos

### 6.1.1 – DNS

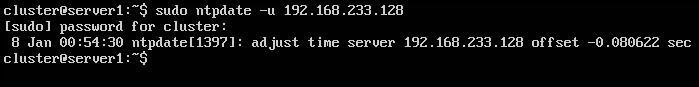


****

**6.1.2 – NTP**

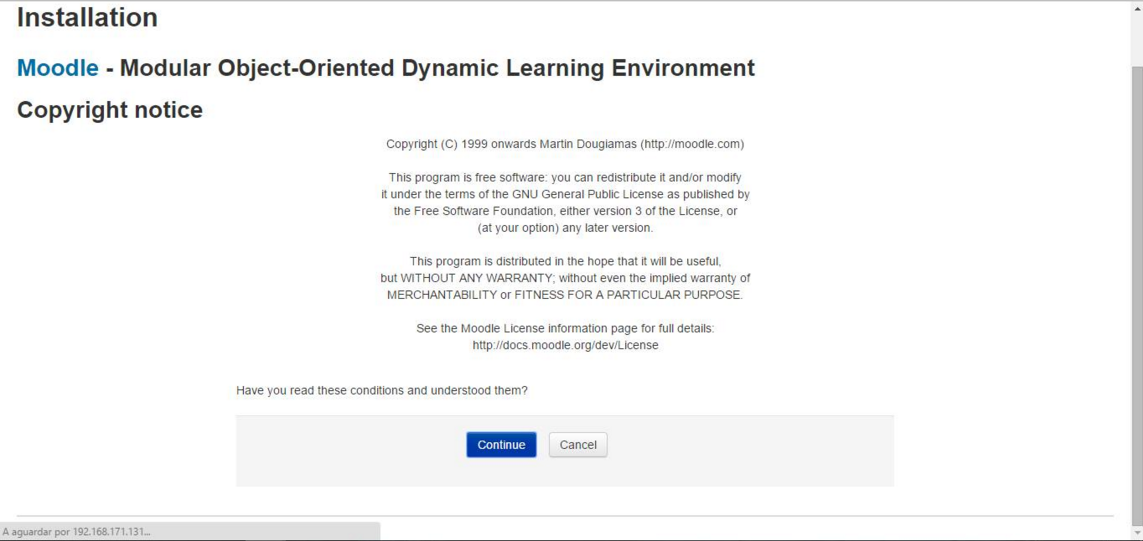




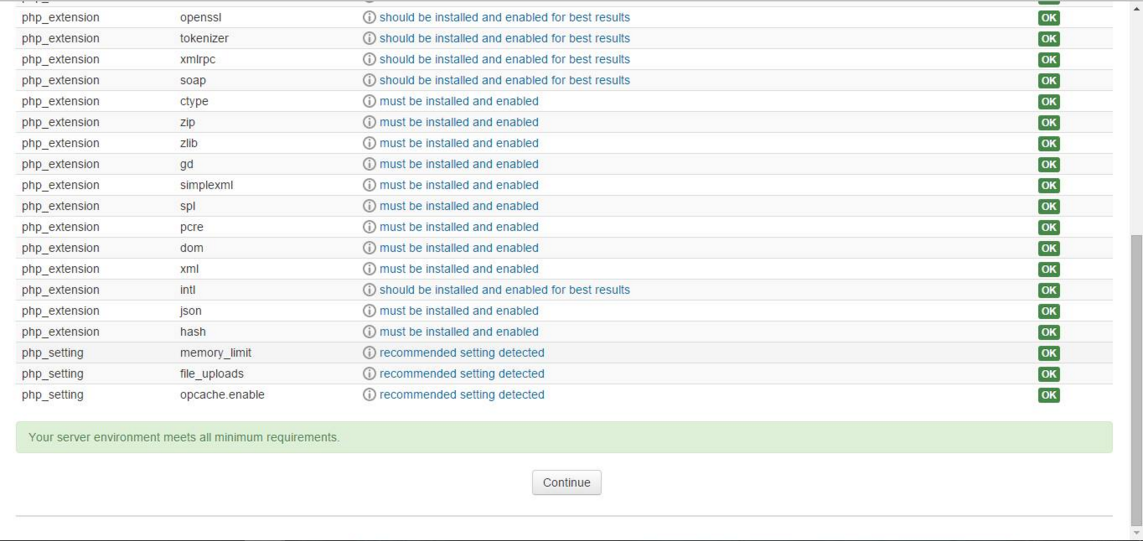


## 6.2 – Serviços Estudados

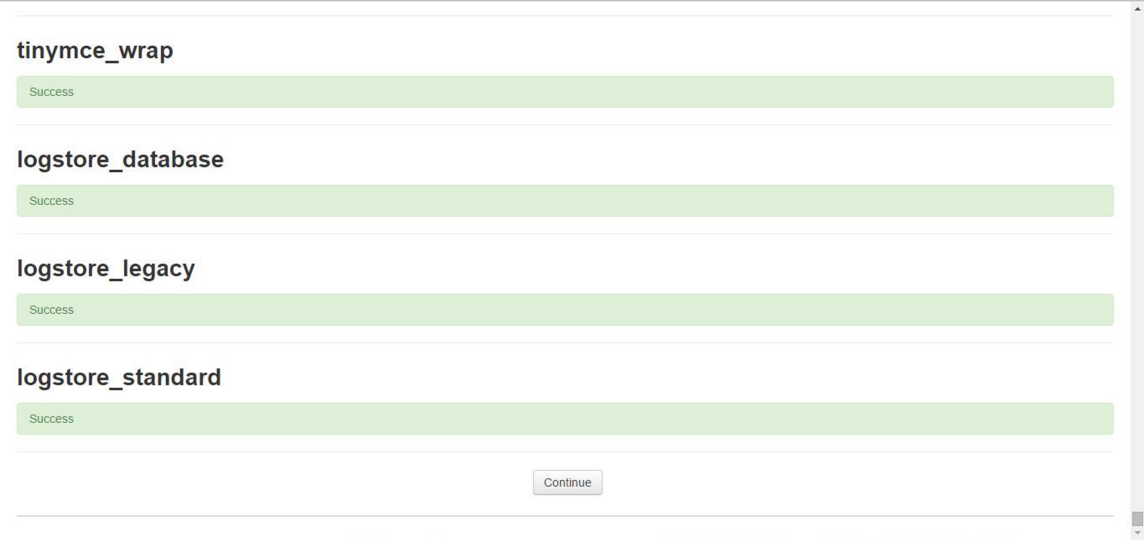
### 6.2.3 Instalação do Moodle

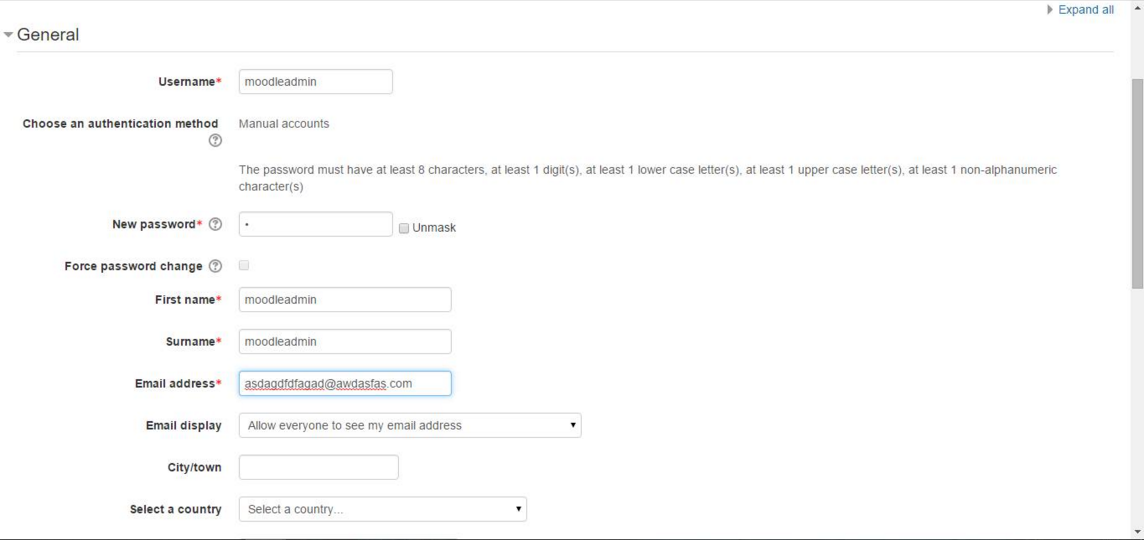


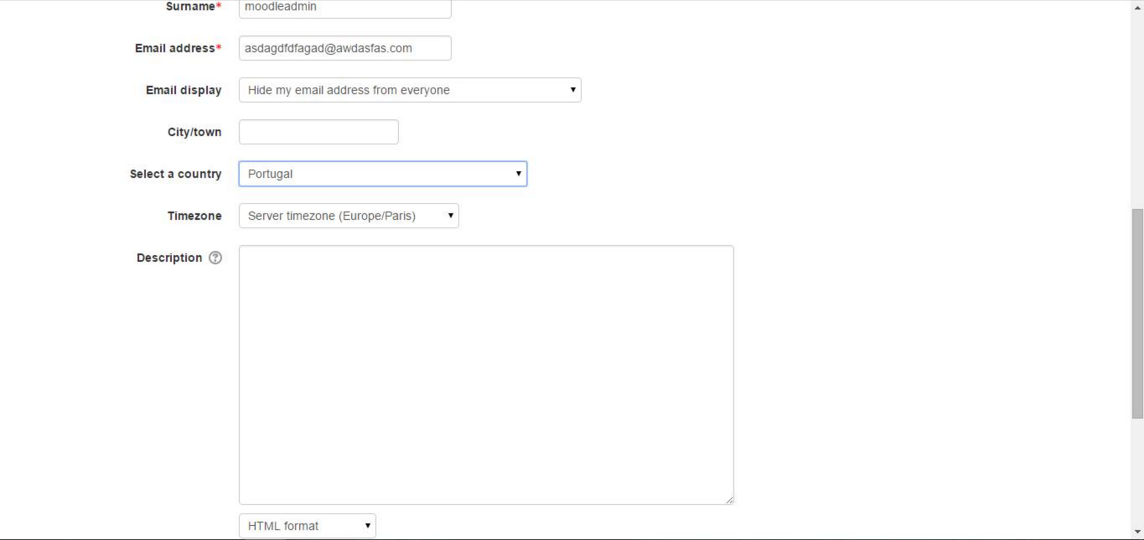


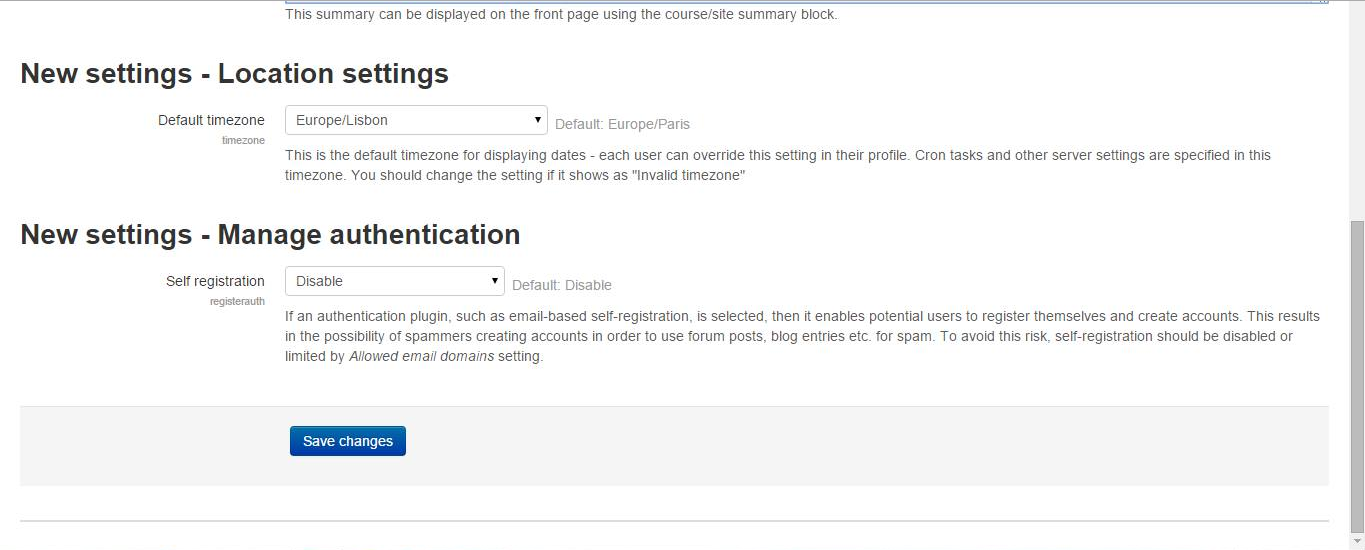


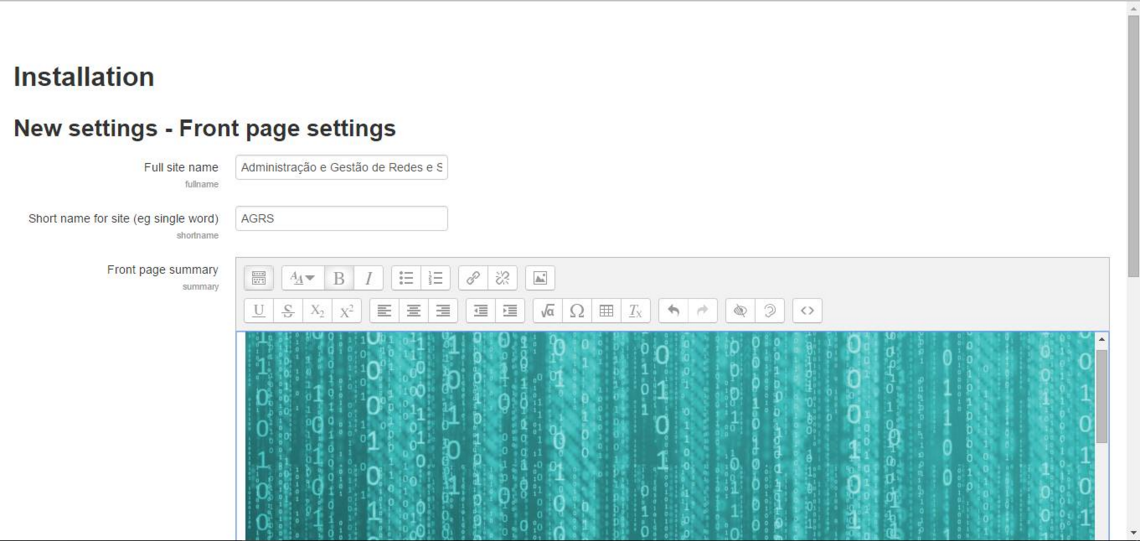


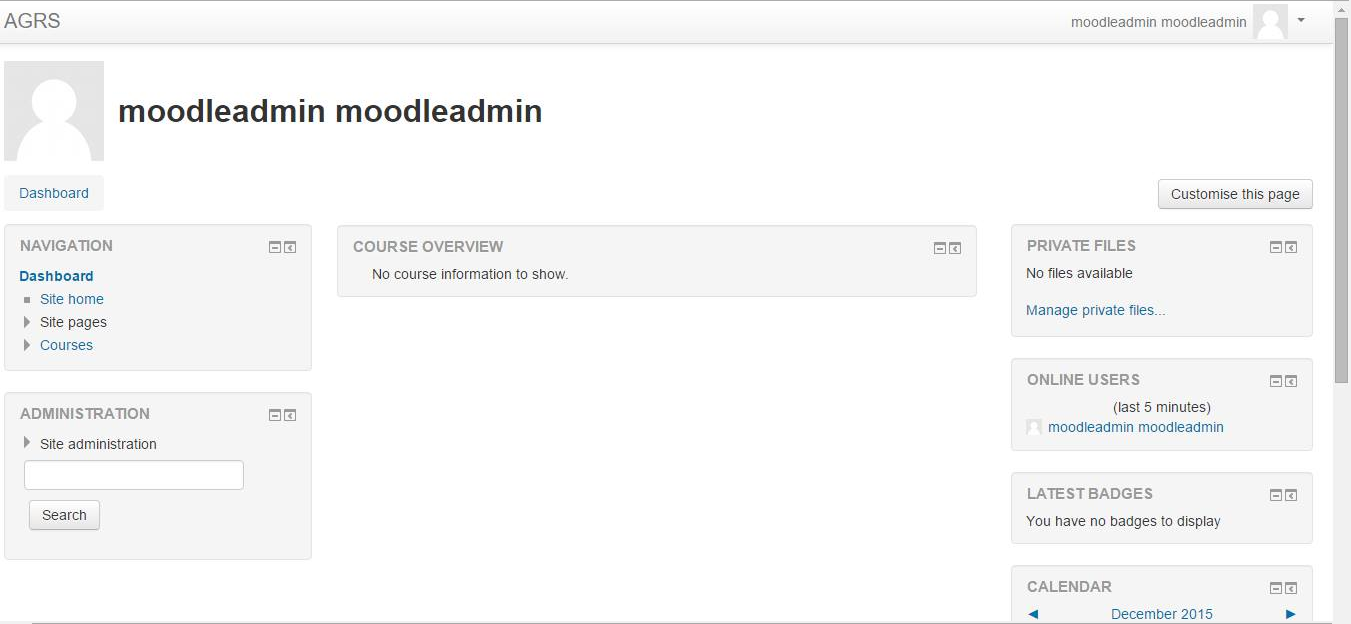


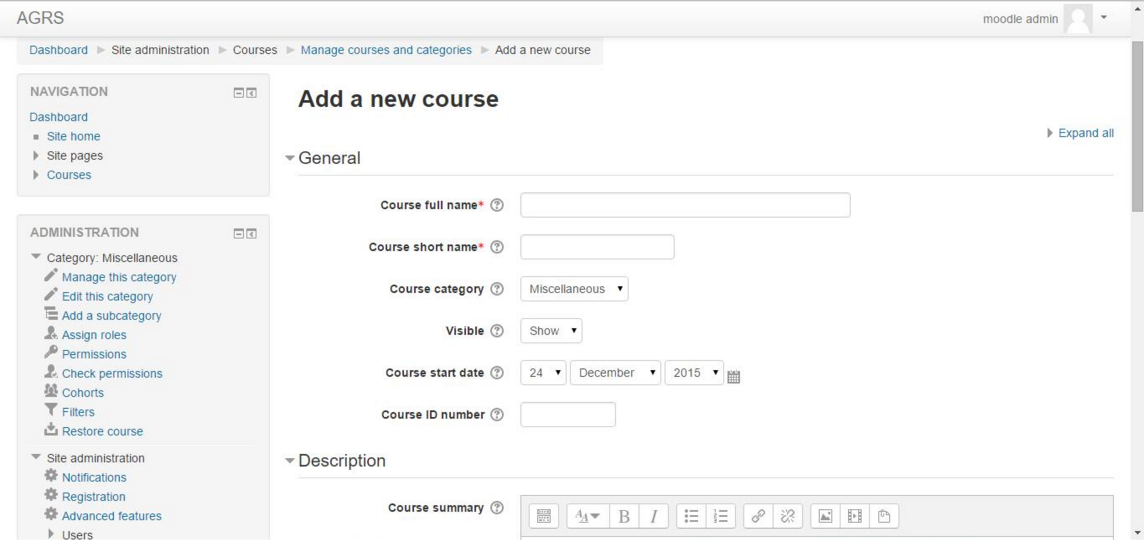


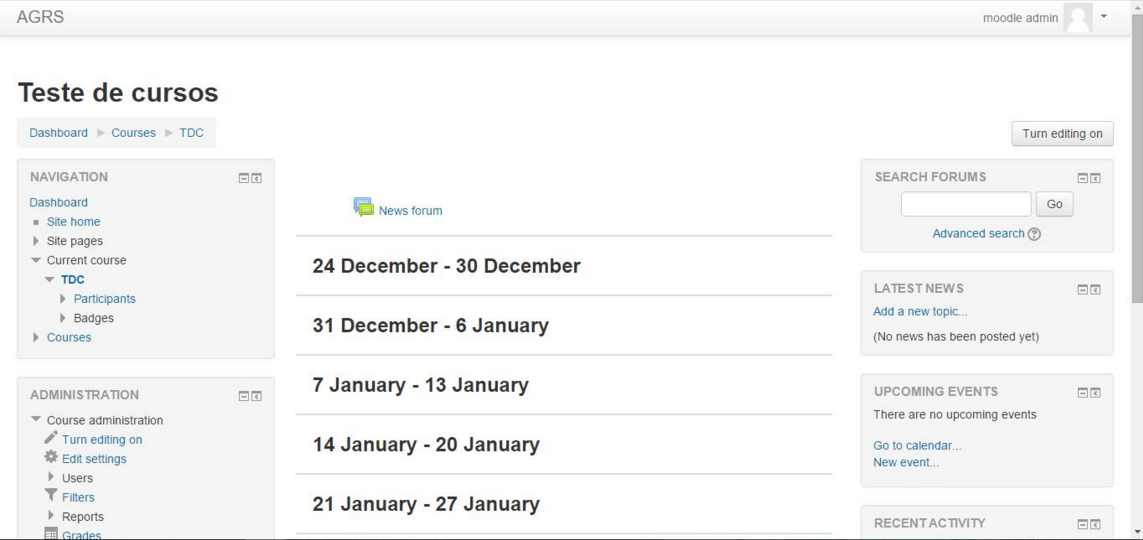














# Bibliografia