A picture containing shape

Description automatically generated

**Relatório Sprint 3**

**Turma 3DG-Grupo 37**

1161274- Artur Muiria

1190827- Luís Araújo

1191240- Danilton Lopes

1201183- João Oliveira

**Professor:**

André Moreira, ASC

**Unidade Curricular:**

Administração de Sistemas

**Data: 08/01/2023**

**Índice**

[**User Story 1** 3](#_Toc124114737)

[*Como administrador da organização quero um plano de recuperação de desastre que satisfaça o MBCO definido na US B5.* 3](#_Toc124114738)

[**User Story 2** 5](#_Toc124114739)

[*Como administrador da organização quero que me seja apresentada de forma justificada a ou as alterações a realizar na infraestrutura por forma a assegurar um MTD (Maximum Tolerable Downtime) de 20 minutos.* 5](#_Toc124114740)

[**User Story 3** 7](#_Toc124114741)

[*Como administrador de sistemas quero que seja realizada uma cópia de segurança da(s) DB(s) para um ambiente de Cloud através de um script que a renomeie para o formato <nome\_da\_db>\_yyyymmdd sendo <nome\_da\_db> o nome da base de dados, yyyy o ano de realização da cópia, mm o mês de realização da cópia e dd o dia da realização da cópia.* 7](#_Toc124114742)

[**User Story 4** 9](#_Toc124114743)

[*Como administrador de sistemas quero que utilizando o Backup elaborado na US C3, seja criado um script quer faça a gestão dos ficheiros resultantes desse backup, no seguinte calendário. 1 Backup por mês no último ano, 1 backup por semana no último mês, 1 backup por dia na última semana.* 9](#_Toc124114744)

[**User Story 5** 12](#_Toc124114745)

[*Como administrador de sistemas quero que o processo da US C3 seja mantido no log do Linux, num contexto adequado, e alertado o administrador no acesso à consola se ocorrer uma falha grave neste processo.* 12](#_Toc124114746)

[**User Story 7** 13](#_Toc124114747)

[*Como administrador da organização quero que me seja apresentado um BIA (Business Impact Analysis) da solução final, adaptando se e onde aplicável o(s) risco(s) da US B4.* 13](#_Toc124114748)

[**User Story 8** 15](#_Toc124114749)

[*Como administrador da organização quero que seja implementada uma gestão de acessos que satisfaça os critérios apropriados de segurança.* 15](#_Toc124114750)

[**User Story 10** 17](#_Toc124114751)

[*Como administrador de sistemas quero que o administrador tenha um acesso SSH à maquina virtual, apenas por certificado, sem recurso a password.* 17](#_Toc124114752)

# **User Story 1**

## *Como administrador da organização quero um plano de recuperação de desastre que satisfaça o MBCO definido na US B5.*

O **plano de recuperação de desastres** (**DRP**) é uma abordagem **estruturada** que descreve como é que uma organização pode **retomar** rapidamente o **trabalho** após um **desastre**. Como evidenciado na ***US B5***, a nossa organização apenas considerará **desastres** ao nível da **aplicação**. Este plano está dividido em **várias** fases:

* **Métricas** relativas ao **tempo**: ***Recovery Point Objective* (RPO)** e ***Recovery Time Objective* (RTO)**;
* **Registo** de pessoas **responsáveis** por cada **ação** no plano;
* **Listagem** dos **aplicativos** e/ou **equipamentos** necessários para um **funcionamento** correto da **organização**;
* **Procedimentos** de ***backup***;
* **Procedimentos** de **recuperação** de **serviços**;
* **Locais** para a **recuperação** dos desastres;
* **Procedimentos** de **restauro**.

Relativamente ao primeiro ponto, as duas **métricas** referidas acima são **fundamentais** na **elaboração** de um **plano de recuperação de desastres**. A primeira métrica, ***Recovery Point Objective* (RPO),** define-se como a **quantidade** de **dados** que uma empresa pode **perder** sem ter as suas **operações** vitais **afetadas**. Este limite de **tolerância** face a um desastre está **relacionado** com a rotina de ***backups***. A nossa organização realiza ***backups*** frequentes o que leva a que o **RPO** tenha um limite **menor** para **aumentar** o **volume** de **informação** salva. Este mesmo **RPO** é de **3 horas**. A segunda métrica, ***Recovery Time Objective* (RTO),** define-secomo o **tempo** **máximo** que uma **aplicação** pode ficar **inativa** até que se consiga **restabelecer** o **sistema** ou o **ambiente** afetado pelo **desastre**. A nossa organização definiu o **RTO** de **2 horas**.

Graphical user interface, text, table

Description automatically generated with medium confidenceRelativamente ao segundo ponto, são **necessários** e **fundamentais** os **responsáveis** por cada **ação** no **plano de recuperação de desastre**. Assim, a lista de cada **interveniente** é a seguinte:

A equipa de **testagem** dos **métodos** de **recuperação** testa estes mesmos métodos **duas** vezes por **ano** em conjunto com o **departamento** do ***DRP***. Através desta testagem é possível **criar** um plano bem definido com todos os **procedimentos** e **alternativas** para a **recuperação** dos **serviços** da organização.

Relativamente ao terceiro ponto, a **organização** tem uma **lista** com todos o ***hardware*** e ***software***, bem como de **serviços** de **nuvem** que são necessários para a **operação** diária. Estes equipamentos são:

* **Computadores** e **monitores**;
* ***Routers***;
* **Base de dados principal e secundária**;
* **Serviços** da ***app*** da organização.

Relativamente ao quarto ponto, os **procedimentos** de ***backup*** contam com os **dispositivos**, **pastas** e **arquivos** e a **forma** como a **recuperação** deve ser feita. A organização faz os ***backups*** com todas as **pastas** e **dados** relativos aos **serviços** através da **base** de **dados** de **suporte**. Os trabalhadores com **permissões** de **administração** desta base de dados tratam da **transferência** de todos os **dados** desde o último ***backup***. De acordo com o **MBCO**, é possível utilizar **90%** dos **dados** totais.

Relativamente ao quinto ponto, os **procedimentos** de **recuperação** de serviços tratam da **recuperação** das **componentes** da aplicação. Sendo estes componentes **fundamentais** para o funcionamento da organização, é necessário estarem sempre **disponíveis**. Para isso, a nossa organização em conjunto com a equipa de **testagem** implementou um sistema de ***failover*** em ***cluster***- ***cluster*** **ativo/passivo**. Assim, na **falha** de algum serviço do servidor/aplicação **principal**, o servidor/aplicação de ***failover*** **assume** e **executa** as suas **funcionalidades** estando em constante **monotorização**.

Relativamente ao sexto ponto, o local de **recuperação** dos **desastres** é o **alojador** **alternativo** ***Azure***. Através deste último, é possível anexar **2** **nós**/**máquinas** **virtuais** utilizando o mesmo **disco** compartilhado em ***cluster***. Este disco compartilhado permite ao mesmo tempo **armazenar** **dados** bem como **suporte** de **aplicações** em ***cluster***. Assim, podemos ter a aplicação **principal** e **secundária** a **compartilharem** o mesmo disco em ***cluster* ativo**/**passivo***.*

Por fim, e relativamente ao sétimo ponto, os **procedimentos** de **restauro** podem ser diferentes mediante os **contextos** de desastre. Estes **englobam** e **complementam** os **procedimentos** de ***backup*** e serviços bem como o **local** de **recuperação** falados anteriormente. Para isso devemos considerar **2** **contextos** de acordo com o **MBCO**: **avaria** na **base** de **dados** **principal** e **avaria** num dos **serviços** da aplicação.

Table

Description automatically generated with medium confidencePara o primeiro desastre os procedimentos são:

Para o segundo desastre os procedimentos são:

Table

Description automatically generated

# **User Story 2**

## *Como administrador da organização quero que me seja apresentada de forma justificada a ou as alterações a realizar na infraestrutura por forma a assegurar um MTD (Maximum Tolerable Downtime) de 20 minutos.*

No ***sprint*** anterior fora pedido para que fossa desenvolvido um ***RPO*** e ***WRT***, e uma delas fazem parte do que é pedido agora que é ***MTD****,* então mantendo a coerência do que fora pedido iremos apresentar algumas proposta que já se fora mencionado no ***sprint*** anterior.

Antes uma contextualização dos acrónimos e conceitos que serão utilizados com a apresentação de ***MTD***.

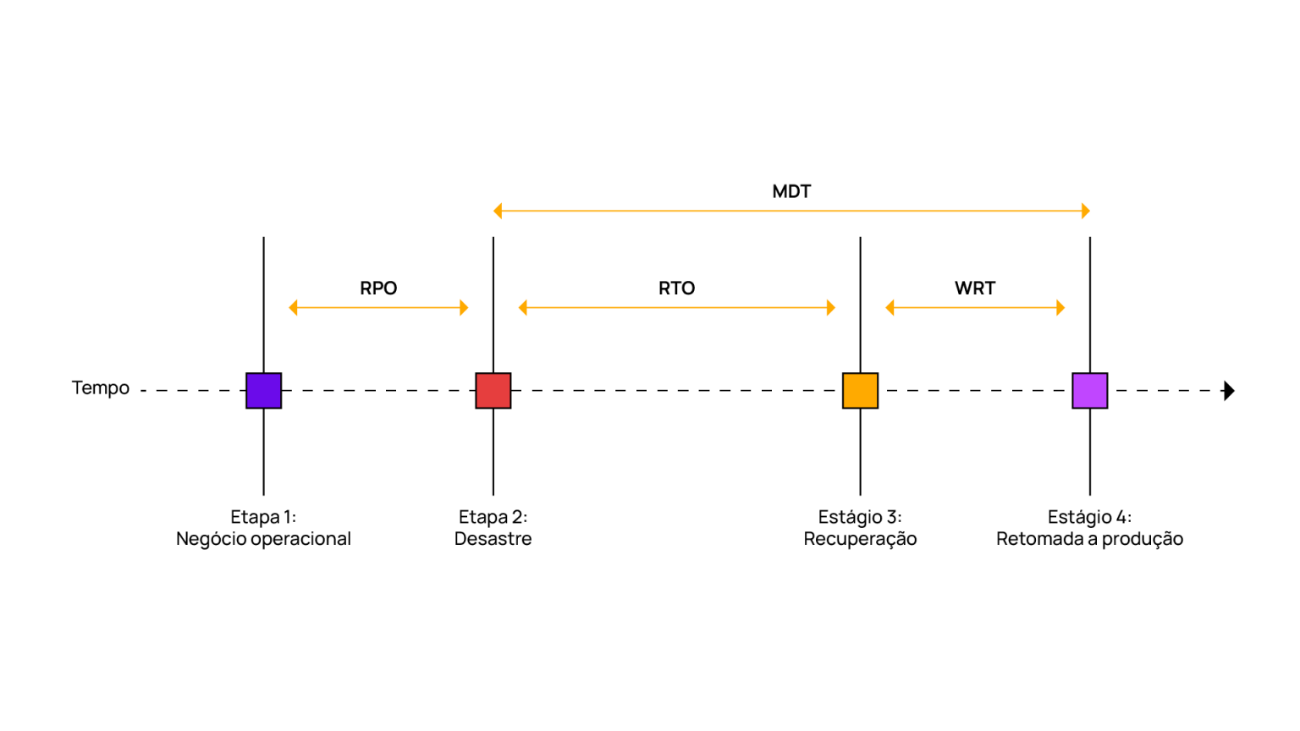


Figura - Linha de tempo de operação impactada por um desastre

* ***RTO***, *Recovery Time Objetive*, é uma métrica para que se calcule o tempo máximo que o que um sistema levaria para que volte a sua operacionalidade normal após um desastre. Para o cálculo de RTO têm-se em conta as questões de prioridade do negócio.
* ***WRT***, *Work Recovery Time*, é usado para determinar o tempo máximo para testar e verificar a operacionalidade de todos os sistemas, sendo estes: **sistemas, aplicações, banco de dados** e outros. É uma etapa crucial na recuperação de desastres. Esta etapa poderá exigir muito esforço ou pouco, mas o pretendido é que não tenha a necessidade de ações humanas, pois com a evolução da tecnologia automatizar processos tornou-se importante, então pretendo com o esforço humano seja nulo ou quase inexistente, tendo um sistema/*software* para faça testes e análise de desempenho dos sistemas, garantindo que esteja tudo a funcionar direito e caso não o estejam, verificar o problema e notificar os responsáveis pela manutenção.
* ***MTD*,** *Maximum Tolerable Downtime*, é o tempo máximo de inoperacionalidade tolerável dos sistemas. Podem existir diferentes ***MTD*** para cada parte diferente da organização e sistemas/softwares utilizados, dependendo também da sua prioridade e o que afeta a inoperacionalidade daquele sistema. O cálculo de ***MTD*** é dado pela soma de ***RTO*** e ***WRT***.

**Algumas alterações ou adaptações à Infraestrutura**

Virtualização de servidores, num único servidor hospedeiro, ou caso a organização tenha possibilidade de adquirir 2 servidores, sendo uma de reserva caso a primeira pare de funcionar e evitando a inconveniência da inoperacionalidade do servidor principal, isto ajudaria na diminuição dos custos quanto à servidores físicos, ajudaria também na potencialização de pedidos que poderiam ser processados de uma vez, uma vez com os servidores virtualizados ter-se-á o desempenho de múltiplos servidores, pois servidores virtuais comportam-se como servidores físicos.

Tendo múltiplos servidores à disposição poder-se-á fazer a divisão das responsabilidades para quais serviços irá receber e tratar, o sistema terá maior dinamismo quando receber um pedido, resolvendo problemas de sobreposição de pedidos e também de rotação de pedidos caso um dos servidores não tiver o comportamento esperado ou estiver indisponível. Desta forma também se evita a não disponibilização de serviços e evita-se a paragem de trabalhos que dependem dos serviços disponibilizados.

Então aconselha-se que para cada conjunto de serviços tenham pelo menos uma dupla de servidores a tratar dos seus pedidos, no qual um servidor anterior poderá ter várias possibilidades para redirecionar o pedido para estes. Fará contactos, tendo de esperar respostas da disponibilidade do servidor e caso não obtenha uma resposta poderá solicitar a outro ou o servidor reserva à sua disponibilidade. De reporta o comportamento emitindo uma mensagem de ***log*** de nível de gravidade **critico** para o servidor de registos de atividades (***facility***podendo ser ***local0*,**que será atribuir ao servidor responsável para redirecionar de pedidos e a ***severity crit*** ou superior dependo de qual seja o servidor), e na mensagem especificando qual servidor apresenta comportamento diferente e qual pedido estava para ser feito.

Desta forma teremos um ***MTD*** nulo ou quase nulo caso um dos servidores virtuais deixe de funcionar corretamente ou pare mesmo de funcionar.

Mas a situação será diferente no caso do servidor hospedeiro principal pare de funcionar. Então será necessário a utilização do servidor reserva. O servidor reserva já comporta como um espelho do servidor principal, ou seja, os dados que estejam no servidor serão “espelhados” para o servidor reserva de forma a evitar que caso haja uma falha de um dos servidores haja uma perda significativa dos dados que ainda não estejam na base de dados, ou no backup. Seguindo a estratégia do sprint anterior o ***mirroring***.

Então com a inoperacionalidade do servidor principal, os pedidos serão redirecionados para o servidor reserva. Como é um servidor é reserva ele não vai ter todos os servidores virtuais a correr evitando o consumo do processamento do hardware e de energia. Tendo a necessidade de iniciar os servidores virtuais que estejam desligados. É então verificado faz-se teste do servidor reserva, no qual se faz testes predefinidos e pedidos dos serviços, analisando a capacidade de processamento e caso haja erro. E depois é colocada à disponibilidade online. Este processo apesar de parecer exaustivo pode ser definido para que seja feito automaticamente pelos sistemas ou pode ser feito pelo administrador de sistemas. O tempo deste processo, no caso ***RTO***, calcula-se que não passe de 15 minutos. E por servidor já ser o destino de ***mirroring*** do servidor principal, acredita-se não haja perda de dados. E no caso de haver seriam quase que mínimos, caso o sistema esteja a funcionar de acordo com o proposto no sprint anterior. E então o tempo para a recuperação de dados, neste caso o ***WRT***, é nulo ou quase isso, não passando de 2 a 3 minutos.

Então com o ***WRT*** e ***RTO*** apresentado teremos que ***MTD (RTO+WRT)*** será de máximo 18 minutos.

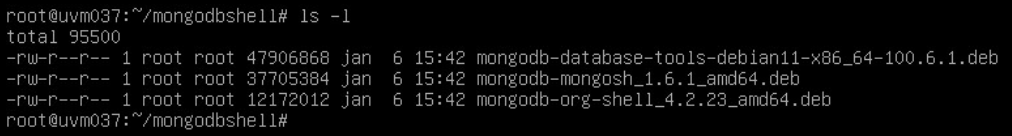
# **User Story 3**

## *Como administrador de sistemas quero que seja realizada uma cópia de segurança da(s) DB(s) para um ambiente de Cloud através de um script que a renomeie para o formato <nome\_da\_db>\_yyyymmdd sendo <nome\_da\_db> o nome da base de dados, yyyy o ano de realização da cópia, mm o mês de realização da cópia e dd o dia da realização da cópia.*

Para desenvolver esta user story foi necessário criar um **script** que faça *backup* das DB’s e no caso do nosso grupo foram 2. Trabalhou-se na **VM** do DEI para esta US e instalou-se 3 ferramentas (consegue-se observar na imagem abaixo) do **MongoDB** em formato **.deb** com a ajuda do comando **apt install** e estas permitem o uso do comando **mongodump.**

Text

Description automatically generatedDe seguida usou-se o comando **dpkg -i** para instalar as ferramentas, optamos por usar o **BitBucket** como nosso ambiente **Cloud.** Portanto foi necessário criar um repositório com o nome: **asistdb** e fez-se um **git clone (sendo este o comando: git clone https://AM57@bitbucket.org/AM89/asistdb.git)** do **https** para dentro da **VM**.



Depois implementou-se o **script** dentro do ficheiro “**script.sh**”, declarou-se as variáveis com os nomes que vamos atribuir às DB’s e que seguem o seguinte formato: **<nome\_da\_db>\_yyyymmdd.tar.** Também foi definido o nome do **dump** de cada DB.  
A seguir fez-se o uso do comando **mongodump** onde se colocou na **–uri** a **connection string** da DB, no **-u** o user e no **-p** a password. Seguidamente fez-se o zip do dump que foi definido na variável (**LOCALIZACAO\_AUTH**/**LOCALIZACAO\_LOG**) com o uso do comando **tar zcvf** (**tar** cria o arquivo e extrai-o no formato **.tar,** as opções **z** (filtra o arquivo através do gzip), **c** (cria um arquivo), **v** (faz a saída detalhada) e o **f** (usa o arquivo compactado) atribuindo um nome ao ficheiro que foi definido na variável (**FICHEIRO\_AUTH**/**FICHEIRO\_LOG**) e por fim eliminou-se o dump, isto tudo foi feito para as duas bases de dados.  
Depois moveu-se os ficheiros zipados para o repositório com o comando **mv**, e um cd para o repositório para possibilitar o **git add** (adiciona uma alteração no diretório de trabalho)**, git commit** (guarda as suas alterações no local do repositório) e por fim o **git push origin** (envia todos os ramos para o ramo principal).   
Isto tudo se mostra exemplificado na imagem abaixo. Text

Description automatically generated

No fim guarda-se o script e executa-se com **./script.sh** e como se pode observar as backups estão criadas.

Graphical user interface, table

Description automatically generated with medium confidence

# **User Story 4**

## *Como administrador de sistemas quero que utilizando o Backup elaborado na US C3, seja criado um script quer faça a gestão dos ficheiros resultantes desse backup, no seguinte calendário. 1 Backup por mês no último ano, 1 backup por semana no último mês, 1 backup por dia na última semana.*

De acordo com o que foi pedido no enunciado foi-se desenvolvido um script que irá utilizar o que fora desenvolvido na US anterior para fazer backups. E então segue o script e bem como a cron

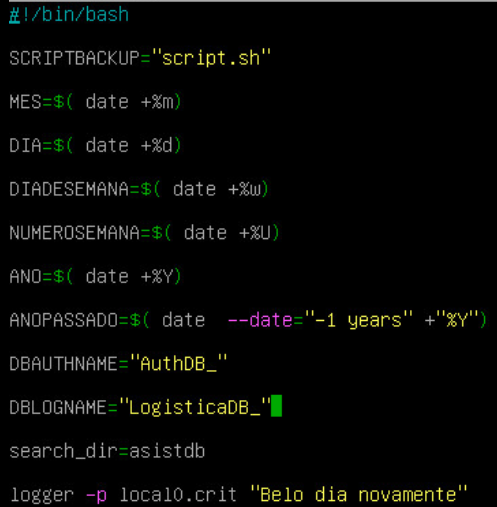


Figura - Declaração de variáveis do script US04

Iniciamos pela declaração das variáveis que posteriormente com o desenvolver do trabalho. Os nomes já são autoexplicativos.

Uma imagem com texto

Descrição gerada automaticamente

Figura - Os backups mensais

As primeiras linhas de instruções só serão executadas quando for um determinado dia do mês. No qual será verificado e depois que confirmado, envia uma mensagem de registo para informar que houve um backup naquele dia. Começa por remover os dados do mesmo mês só que do ano passado. E de seguida executa o ficheiro desenvolvido na US passada.

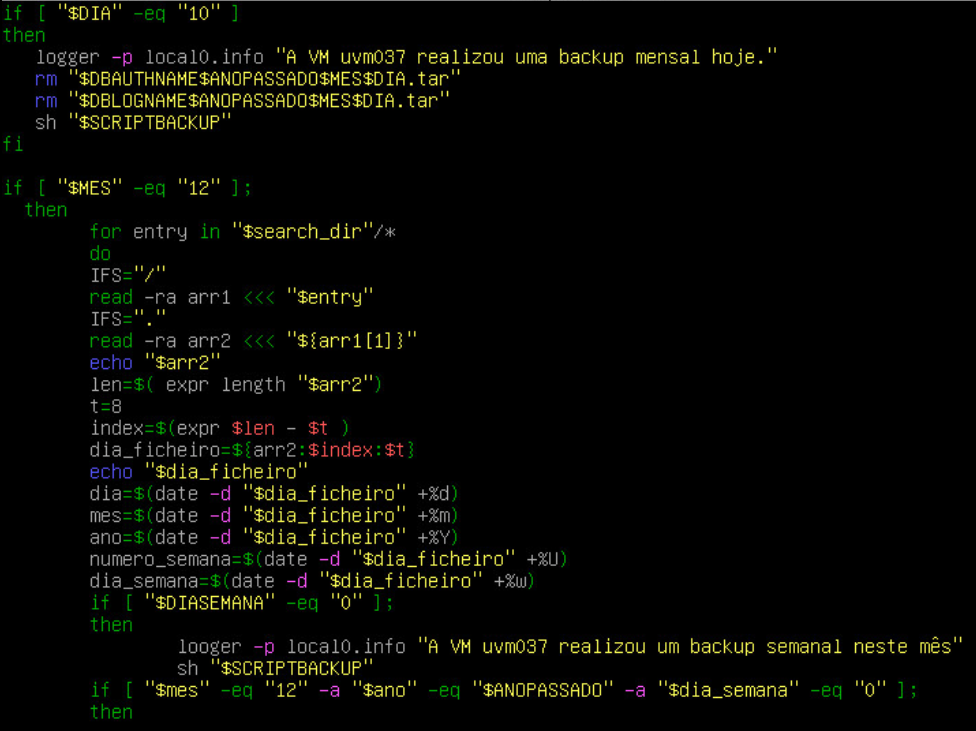
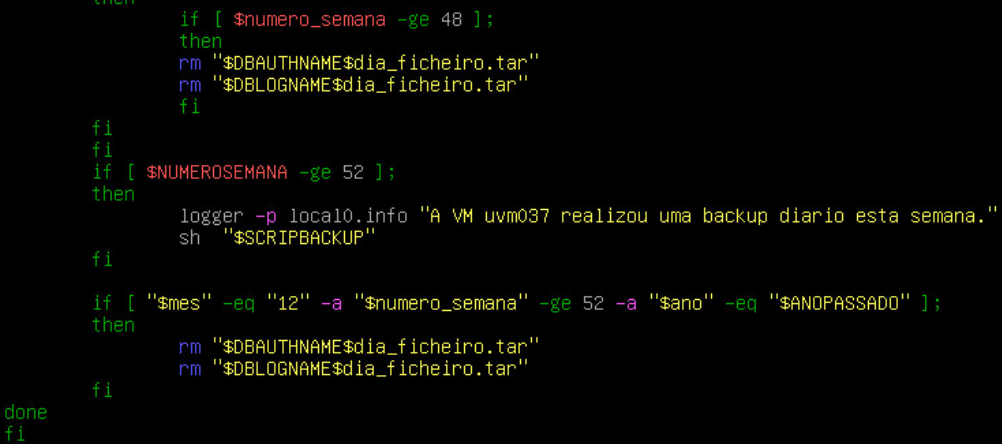
 

Figura - Os backups semanais do último mês e diários da última semana

Para o tratar dos backups no último mês e última semana foi escrito as instruções que estão acima.

Primeiro verifica-se qual o mês atual e cado seja 12, ou seja, o último mês do ano. Caso for ele executa as linhas de instruções dentro da condição. As seguintes linhas de execução criam um ciclo de acordo com o nome dos ficheiros que estão na cloud. Faz os backups dos dias atuais e por fim elimina os do ano passado seguindo as conduções. A mesma coisa acontece quando se verifica que há um ficheiro que no passado foi criado na última semana e copia caso a atual semana seja a última.



Figura - O ficheiro crontab

Para testar adicionado esta linha ao ficheiro de ***crontab*** e ver que o ficheiro era executado ou não.

Como pode-se ver no ficheiro que recebe registo foi executado o script, mas como não era de um dia que respeitava as condições, então deu sucesso.



Figura – Resultado da testagem de crontab

E por fim coloca-se o horário correto para a execução dos backups.



Figura - O crontab final

# **User Story 5**

## *Como administrador de sistemas quero que o processo da US C3 seja mantido no log do Linux, num contexto adequado, e alertado o administrador no acesso à consola se ocorrer uma falha grave neste processo.*

Pretende-se que o administrador após o login seja alertado de qualquer falha em relação a execução do backup. Foi necessário editar o ficheiro **/etc/rsyslog.conf** e descomentar o facility **cron.\*** (é um utilitário de agendamento de trabalho presente em sistemas de unix) e acrescentar o facility **cron.err** e o seu ficheiro de log **-/var/log/cron.err.**

Text

Description automatically generated

De seguida editou-se o ficheiro **/etc/profile** onde criou-se uma variável com o uso de **wc -l** que imprime o número de linhas presentes no ficheiro **/var/log/cron.err.** E por fim fez-se uma condição **if** para verificar se o ficheiro **cron.err** tem conteúdo ou não para depois imprimir uma mensagem consoante o resultado da condição.

Text

Description automatically generated

Text

Description automatically generated

Podemos verificar que não houve erro de backup.

# **User Story 7**

## *Como administrador da organização quero que me seja apresentado um BIA (Business Impact Analysis) da solução final, adaptando se e onde aplicável o(s) risco(s) da US B4.*

Para a realização da *Bussiness Impact Analysis* no nosso projeto, abordamos dois tópicos essenciais no contexto do nosso projeto, sendo os mesmos:

-Valores de risco dos módulos

-Nível de criticidade

**Valores de risco dos módulos:**

Quanto ao valor de risco dos módulos do sistema, conseguimos identificar alguns que são de facto cruciais para o funcionamento do sistema (no valor de risco será usada a escala anteriormente definida na us4 do sprint anterior):

-SPA: Para os utilizadores conseguirem usar qualquer funcionalidade do sistema terá de ter o modulo SPA funcional, logo o valor de risco atribuído é 9.

-Logística: Devido a este modulo conter os camiões e das rotas que serão necessários no planeamento da frota e para listar/criar camiões e rotas, o valor de risco atribuído a este modulo é de 6.

-Gestão de Armazéns: Este modulo é utilizado nas funcionalidades de criar/listar armazém e criar/listar entrega, logo devido a claramente haver um menor risco do que nos outros módulos previamente referidos, o valor de risco atribuído é de 4.

**Nível de Criticidade:**

**-Segundo o RTO, MTPD e POR:**

|  |  |  |  |
| --- | --- | --- | --- |
| RTO | MPTD | RPO | Nível de Criticidade |
| <4h | <4h | >0h e <1h | Muito alto |
| >4h e <1d | >4h e <1,5d | >1h e <1d | Alto |
| >1d e <3d | >1,5d e <4d | >1d e <3d | Moderado |
| >3d e <14d | >4d e <15d | >3d e <7d | Baixo |
| >14d e <30d | >15d e <37d | >7d e <14d | Muito baixo |

**-Segundo o impacto da reputação:**

|  |  |
| --- | --- |
| Impacto no Negócio | Nível de Criticidade |
| Falha ou impedimento no negócio, que podem provocar várias implicações, como por exemplo o cliente ficar insatisfeito, perdas de rendimento, penalidades contratuais e/ou quebra de integridade de dados e que afetam o negócio comprometendo-o de forma irreversível a médio/grande-prazo | Muito alto |
| Falha ou impedimento no negócio envolvente, que podem provocar várias implicações, nomeadamente insatisfação do cliente, perdas de rendimento, penalidades contratuais, quebra de integridade de dados e que afetam o negócio comprometendo-o de forma irreversível a curto-prazo. | Alto |
| Falha ou erros no âmbito de determinadas operações de  negócio, que originam alertas ou reclamações, mas que não  impedem o negócio na sua totalidade | Moderado |
| Falhas com impacto residual no negócio dada  a sua rápida resolução ou pouca relevância parra o negócio | Baixo |
| Falhas internas sem interação com comunicações externas,  que não tem qualquer impacto na continuação do negócio | Muito baixo |

# **User Story 8**

## *Como administrador da organização quero que seja implementada uma gestão de acessos que satisfaça os critérios apropriados de segurança.*

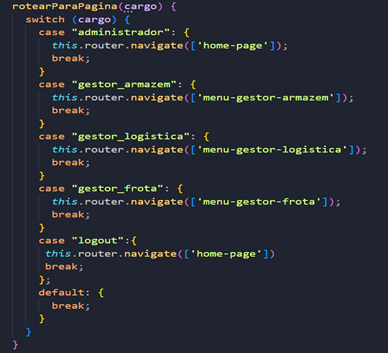
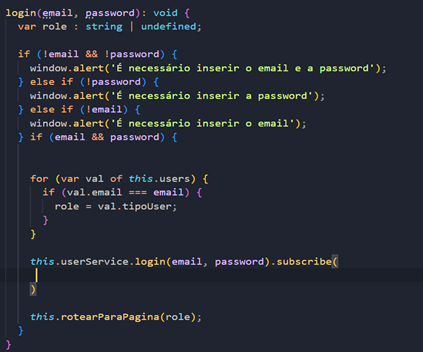
**Gestão de acessos**

Para a realização de uma gestão de acessos que satisfaz os critérios apropriados de segurança é necessário que o plano verifique certos critérios que irei mencionar.

- Autenticação forte: Este parâmetro foca-se em que os utilizadores tenham forte credenciais de login. Para aplicar no nosso projeto utilizamos senhas com caracteres alfanuméricos e codificamos a palavra-passe na base de dados como mostra a seguinte figura:



- Autorização de acesso: Este parâmetro foca-se em limitar o acesso dos utilizadores apenas as informações e funcionalidades que necessitam consoante a sua função. Para aplicar no nosso projeto inserimos um atributo na classe utilizador denominado “tipoUser” que diferencia os utilizadores consoante o seu cargo na empresa, logo condiciona também as funcionalidades que cada utilizador terá acesso.



-Políticas de acesso: Este parâmetro foca-se em estabelecer regras para a utilização dos recursos da organização. Para aplicar no nosso projeto elaboramos as seguintes regras para que os funcionários saibam o que é esperado dos mesmos.

Regras:

-Não utilizar a conta de outro funcionário em qualquer circunstância.

- Na ocorrência de algum problema técnico, informar um utilizador que seja administrador, para o mesmo entrar em contacto com a equipa de desenvolvimento.

- Caso algum dos servidores fique inativo, contactar diretamente a equipa de manutenção.

**-**Controlo de acesso ao hardware: Este parâmetro foca-se na proteção de equipamentos e recursos críticos como por exemplo servidores. Como neste projeto não temos acesso ao hardware dos nossos servidores, não é necessário focar neste tópico.

# **User Story 10**

## *Como administrador de sistemas quero que o administrador tenha um acesso SSH à maquina virtual, apenas por certificado, sem recurso a password.*

De facto, para esta ***User Story*** executaram-se vários comandos tanto em ***Linux*** como em ***Windows***. Por um lado, e relativamente ao ***Linux***, começou-se por aceder à pasta ***/etc/ssh/sshd\_config*** onde se colocou o ***PermitRootLogin*** como ***yes*** de forma a se **permitir** fazer o **login** como ***root*** através de um utilizador **remoto**. Tal se verifica na imagem abaixo.

Text, application

Description automatically generated

Após isso, criou-se um ficheiro ***authorized\_keys*** onde vão estar **especificadas** as chaves ***SSH*** que podem ser usadas para fazer **login** na conta ***root***.

Text

Description automatically generatedPor outro lado, e relativamente ao ***Windows***, o primeiro passo foi executar o comando ***ssh-keygen*** como mostra a imagem abaixo. Resumidamente, este serve para **criar** novos pares de **chaves** de autenticação para ***SSH*** utilizados para automatizar ***logins***.

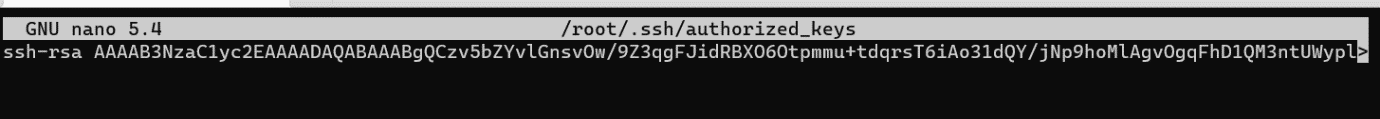
Após isso acedeu-se ao diretório do ***ssh*** onde estavam 2 ficheiros das chaves: um **público** e um **privado**. Abriu-se o ficheiro público ***id\_rsa.pub*** como mostra a imagem abaixo e copiou-se a chave.

Text

Description automatically generated

Para se colocar a chave pública no ficheiro ***authorized\_keys*** em ***Linux*** utilizou-se o comando ***ssh-copy-id*** ao invés de se escrever toda a chave à mão. O resultado está nas duas imagens abaixo. Uma vez que a chave tenha sido **autorizada** para ***SSH***, é **concedido** o **acesso** ao servidor sem uma ***password***.





Por fim, foi dado o ***logout*** da máquina ***Linux*** e tentou-se novamente o ***login***. Como mostra a imagem abaixo, não foi necessária ***password*** para se conectar ao ***root***.

Text

Description automatically generated