

# Instalação e Configuração do QM 2.40+ em Red Hat (RHEL 7.9)

- [Preparação](#)
- [Licença](#)
- [Instalando o banco de dados repositório \(PostgreSQL 12\)](#)
- [Instalando a camada de aplicação web server \(Apache Tomcat 9.0.43\)](#)

# Preparação

Este tutorial e procedimentos está assumindo que serão 2 máquinas distintas. Uma será a camada de repositório, hospedando o PostgreSQL, e a outra camada, de aplicação, hospedará o web server Apache Tomcat e o MD2 QualityManager.

Para executar o procedimento é necessário:

1. Os 2 servidores terem conexão com os repositórios oficiais na internet:
  1. Camada aplicação:
    1. <https://cdn.redhat.com> -> Repositório oficial da Red Hat. Para download de libs, pacotes de apoio a administração e atualização do S.O..
    2. <https://archive.apache.org> -> Repositório oficial da Apache. Para download do web server Tomcat e suas bibliotecas
  2. Camada repositório:
    1. <https://cdn.redhat.com> -> Repositório oficial da Red Hat. Para download de libs, pacotes de apoio a administração e atualização do S.O.
    2. <https://download.postgresql.org> -> Repositório oficial do PostgreSQL. Para download do RDBMS PostgreSQL e suas bibliotecas.
2. Com os sistemas operacionais licenciados e sincronizados através do "subscription manager".
3. Fazer os downloads da aplicação e arquivos do [repositório oficial MD2](#):
  1. Aplicação: qualityManager-prj.war
  2. Estrutura de banco de dados: DB\_QM.psql.gz
  3. Arquivos da aplicação: qm\_static.tar.gz
  4. Ferramenta de apoio: PortalPass.jar

# Licença

Para verificar se o servidor está com a licença ativa e funcional através do comando.

```
$ sudo subscription-manager list --consumed
```

Caso contrário, será necessário fazer o registro com o comando

```
$ sudo subscription-manager register
```

# Instalando o banco de dados repositório (PostgreSQL 12)

Instalação do repositório RPM:

```
$ sudo yum install -y https://download.postgresql.org/pub/repos/yum/repos/pms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
```

Instalação do servidor RDBMS PostgreSQL e o pacote "contrib":

```
$ sudo yum install -y postgresql12-server
$ sudo yum install -y postgresql12-contrib
```

Iniciar o banco de dados:

```
$ sudo /usr/pgsql-12/bin/postgresql-12-setup initdb
```

Configurar o serviço (daemon) para subir automaticamente com o sistema operacional:

```
$ sudo systemctl enable postgresql-12
$ sudo systemctl start postgresql-12
```

Configurar o firewall para aceitar a porta do PostgreSQL. A porta 5432 é a padrão do RDBMS:

```
$ sudo firewall-cmd --permanent --add-port=5432/tcp
$ sudo firewall-cmd --reload
```

Conectar no banco de dados e fazer as criações dos usuários:

```
$ sudo -u postgres psql

create user md2net with encrypted password 'md2net2018';
alter user md2net with superuser;
```

```
create user mdm with encrypted password 'trocarsenha';
```

Permitir conexões remotas e o método de autenticação no PostgreSQL.

```
$ sudo vim /var/lib/pgsql/12/data/pg_hba.conf

# IPv4 local connections:
host    all             md2net          0.0.0.0/0              md5
# IPv6 local connections:
host    all             md2net          ::1/128                md5
```

Permitir conexão remota:

```
$ sudo vim /var/lib/pgsql/12/data/postgresql.conf

listen_address='*'
```

Reiniciar o serviço:

```
$ sudo systemctl restart postgresql-12.service
```

Configurar o perfil do usuário no SO para conectar no banco:

```
$ cd ~
$ vim .pgpass

127.0.0.1:5432:qualitymanager:md2net:md2net2018

$ chmod 600 .pgpass
```

Criar o banco de dados, repositório da ferramenta MD2 QualityManager, importando o arquivo de estrutura do banco:

```
$ createdb --lc-collate pt_BR.UTF-8 --lc-ctype pt_BR.UTF-8 -E UTF-8 -O postgres -T template0 -
e qualitymanager
$ gunzip -c DB_QM.psql.gz | psql -h 127.0.0.1 -U md2net qualitymanager
```

# Instalando a camada de aplicação web server (Apache Tomcat 9.0.43)

Instalação do Java JDK através do repositório oficial:

```
$ sudo yum install java-1.8.0-openjdk-1:1.8.0.312.b07-1.el7_9.x86_64
```

Criação do grupo e usuário "tomcat" no sistema operacional:

```
$ sudo groupadd --system tomcat  
$ sudo useradd -d /opt/tomcat -r -s /bin/false -g tomcat tomcat
```

Download do pacote 'wget' para utilizar no download, descompactação, movimentação e definição de usuário de S.O.:

```
$ sudo yum install wget  
$ sudo yum install vim  
$ cd /tmp  
$ sudo wget https://archive.apache.org/dist/tomcat/tomcat-9/v9.0.43/bin/apache-tomcat-9.0.43.tar.gz  
$ sudo tar xvf apache-tomcat-9.0.43.tar.gz -C /opt  
$ sudo ln -s /opt/apache-tomcat-9.0.43/ /opt/tomcat  
$ sudo chown -R tomcat:tomcat /opt/tomcat  
$ sudo chown -R tomcat:tomcat /opt/apache-tomcat-9.0.43
```

Configuração para que seja um serviço de sistema (daemon):

```
$ sudo vim /etc/systemd/system/tomcat.service  
  
[Unit]  
Description=Tomcat Server  
After=syslog.target network.target
```

```
[Service]
Type=forking
User=tomcat
Group=tomcat

Environment=JAVA_HOME=/usr/lib/jvm/jre
Environment=' JAVA_OPTS=-Djava.awt.headless=true'
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=' CATALINA_OPTS=-Xms512M -Xmx1024M'
ExecStart=/opt/tomcat/bin/catalina.sh start
ExecStop=/opt/tomcat/bin/catalina.sh stop

ReadWritePaths=/opt/qm_static/

[Install]
WantedBy=multi-user.target

$ sudo systemctl daemon-reload
$ sudo systemctl start tomcat
$ sudo systemctl enable tomcat
$ systemctl status tomcat
```

Liberação de porta no firewall:

```
$ sudo firewall-cmd --permanent --add-port=8080/tcp
$ sudo firewall-cmd --reload
```

Teste para conferir se aplicação Tomcat está funcional:

```
$ curl -v http://127.0.0.1:8080
```

Descompactação dos arquivos:

```
$ tar -xvzf qm_static.tar.gz
$ sudo mv /tmp/qm_static /opt
$ sudo chown -R tomcat:tomcat /opt/qm_static/
```

Configuração para que a aplicação reconheça a máquina remota que hospeda o repositório (utilizar a ferramenta de apoio "PortalPass.jar" para gerar a senha criptografada):

```
$ sudo vim /opt/tomcat/lib/qm.app.properties

qm.app.db.hibernate.connection.url=jdbc:postgresql://ipDoPostgreSQL:5432/qualitymanager
qm.app.db.hibernate.connection.username=md2net
qm.app.db.hibernate.connection.password=aaAAaadasdweasdeA==
qm.app.db.hibernate.default_schema=public
```

Publicação do WAR da aplicação no Tomcat:

```
$ sudo mv /tmp/qualityManager-prj.war /opt/tomcat/webapps/
$ sudo systemctl restart tomcat.service
```

Pronto! Teste via browser.

```
http://ipdoqm:8080/qualityManager-prj/login.xhtml
```