

Publicar o Webrun em sistemas Linux Ubuntu.

Pré-requisitos:

- Wine, a partir da versão 1.2 (para utilização do Report Builder);
- JDK ou JRE 1.6, 1.7 ou 1.8.0_121 (Para versão Studio);
- Tomcat 6, Tomcat 7 até a versão 7.0.56 ou Tomcat 8.5.12 (Para versões Studio);
- webrun.war;
- Dependências do webrun (pastas reports, systems e config).

Instalando o Wine1.8

O Wine (Wine is Not a Emulator) é um programa que adapta o funcionamento de sistemas Linux, Mac, Solares e outros, para a utilização de aplicativos e programas nativos do Windows, simulando o sistema de arquivos e instalando inúmeras bibliotecas próprias do Windows.

Primeiramente, remova a versão mais antiga do Wine (caso exista).

- 1- Abra o terminal ;
- 2- Digite `$sudo apt remove wine wine1.8 wine-stable libwine* fonts-wine* && sudo apt autoremove;`

Caso não exista, siga os passos seguintes:

- 3- Adicione o suporte a arquitetura i386 `$sudo dpkg --add-architecture i386;`
- 4- Baixe a chave do repositório `$wget -nc https://dl.winehq.org/wine-builds/Release.key;`
- 5- Adicione o repositório `$sudo apt-add-repository https://dl.winehq.org/wine-builds/ubuntu/ -y;`

Caso utilize o Linux Mint 18.1 adicione o repositório dessa maneira:

- 6- Digite `$sudo apt-add-repository 'deb https://dl.winehq.org/wine-builds/ubuntu/ xenial main';`
- 7- Para versão 17.x adicione digitando `$sudo apt-add-repository 'deb https://dl.winehq.org/wine-builds/ubuntu/ trusty main';`
- 8- Atualize o APT `$sudo apt-get update;`
- 9- Instale o Wine 1.8 `$sudo apt-get install --install-recommends winehq-devel.`

Configurando Wine:

- 1- Abra o terminal;
- 2- Digite `$winecfg` (O comando irá abrir o painel gráfico de configuração do Wine);
 - 3- Ao teclar Enter, antes de abrir o painel de configuração, o Wine será atualizado. Instale qualquer atualização que for solicitada;
 - 4- Após abrir o painel gráfico, na aba “Aplicativos” na opção “Versão do Windows”, selecione o Windows a sua escolha, contanto que seja acima do Windows 7;

5- Na aba “Gráficos”, dentro do setor de “Resolução de Tela”, defina 99 ppp.

Obs.: Não é necessária mais nenhuma configuração adicional para a utilização do Report Builder.

Criando o contexto para as dependências do Webrun:

Certifique-se de copiar as pastas “reports, systems e config” do seu ambiente de desenvolvimento, localizadas na pasta raiz do webrun, e disponibiliza-las no ambiente Linux.

- 1- Abra o terminal;
- 2- Entre na pasta do usuário `$cd /home/<usuário>;`
- 3- Crie a pasta softwell `$mkdir softwell;`
- 4- Entre na pasta softwell `$cd softwell/;`
- 5- Crie a pasta webrun `$mkdir webrun;`
- 6- Entre na pasta webrun `$cd webrun/;`
- 7- Crie a pasta conf `$mkdir conf.`

Agora moveremos as pastas de dependência do webrun para dentro do diretório conf/:

- 1- Abra o terminal;
- 2- Entre no diretório onde encontra-se as pastas de dependência do webrun e mova-as para o diretório, recém criado, conf/;
- 3- Mova a reports `$mv reports/ /home/<usuário>/softwell/webrun/conf/;`
- 4- Mova a systems `$mv systems/ /home/<usuário>/softwell/webrun/conf/;`
- 5- Mova a config `$mv config/ /home/<usuário>/softwell/webrun/conf/;`

Pronto, agora só precisamos instalar o Java (JDK ou JRE) e o Tomcat.

Instalando o Java JDK 7:

- 1- Abra o terminal;
- 2- Entre na pasta do webrun `$cd /home/<usuário>/softwell/webrun/;`
- 3- Baixe a JDK 7 `$wget “http://download.oracle.com/otn-pub/java/jdk/7u79-b15/jdk-7u79-linux-x64.tar.gz”` para ambientes 64 bits ou `$wget “http://download.oracle.com/otn-pub/java/jdk/7u79-b15/jdk-7u79-linux-i586.tar.gz”` para ambientes 32 bits;
- 4- Descompacte o arquivo baixado `$tar -vzf nomedoarquivo.tar.gz;`
- 5- Apague o arquivo .tar.gz `$rm nomedoarquivo.tar.gz;`

6- Renomeie o arquivo descompactado para Java `$mv nomedoarquivo/ java/;`

Agora configuraremos algumas variáveis para que o sistema reconheça esse diretório (java/) como a pasta home da nossa jdk:

7- Abra o profile `$sudo nano /etc/profile;`

8- Antes dos primeiros comandos if escreva:

```
JAVA_HOME=/home/<usuário>/softwell/webrun/java/jdk1.7.0_79 export  
JAVA_HOME
```

```
PATH=$JAVA_HOME/bin:$PATH
```

```
export PATH
```

9- Saia com CTRL+X;

10- Salve;

11- Digite o comando `$java -version;`

12- O retorno deve ser algo parecido com isso:

```
Java version "1.7.0_79"
```

```
Java(TM) SE Runtime Environment (build 1.7.0_79-b15)
```

```
Java HotSpot(TM) 64-Bit Server VM (build 24.79-b02, mixed mode)
```

Pronto, o Java está instalado.

Instalando o Tomcat7:

1- Abra o terminal;

2- Entre na pasta webrun `$cd /home/<usuário>/softwell/webrun/;`

3- Baixe o Tomcat7

```
$wget
```

```
"http://ftp.unicamp.br/pub/apache/tomcat/tomcat-7/v7.0.64/bin/apache-tomcat-7.0.64.  
tar.gz";
```

4- Descompacte o arquivo .tar.gz `$tar -vzf nomedoarquivo.tar.gz;`

5- Apague o arquivo .tar.gz `$rm nomedoarquivo.tar.gz;`

6- Renomeie o arquivo descompactado para tomcat `$mv nomedapasta/ tomcat/;`

Agora precisamos tornar nosso Tomcat acessível como serviço. Para isso, criaremos um pequeno script:

7- Entre no diretório `/etc/init.d` `$cd /etc/init.d;`

8- Crie um arquivo texto com nome tomcat `$sudo gedit tomcat;`

9- Escreva o seguinte script:

```
#!/bin/bash
```

```
JAVA_HOME=/home/<usuário>/softwell/webrun/java/jdk1.7.0_79 export  
JAVA_HOME
```

```
PATH=$JAVA_HOME/bin:$PATH
```

```
export PATH
```

```
CATALINA_HOME=/home/<usuário>/softwell/webrun/tomcat
```

```
case $1 in
```

```
start)
```

```
sh $CATALINA_HOME/bin/startup.sh
```

```
;;
```

```
stop)
```

```
sh $CATALINA_HOME/bin/shutdown.sh
```

```
;;
```

```
restart)
```

```
sh $CATALINA_HOME/bin/shutdown.sh
```

```
sh $CATALINA_HOME/bin/startup.sh
```

```
;;
```

```
esac
```

```
exit 0
```

10- Salve;

11- Torne o script executável `$chmod +x tomcat;`

Agora testaremos, se nosso script está funcionando:

12- Inicie o serviço do Tomcat `$service tomcat start;`

13- Pare o serviço do Tomcat `$service tomcat stop;`

14- Teste o restart do Tomcat `$service tomcat restart;`

Por razões de segurança, nenhum usuário ou senhas são criados para as funções de gerente do Tomcat por padrão. Iremos definir um agora:

15- Abra o arquivo tomcat-users.xml `$CATALINA_HOME/conf/tomcat-users.xml;`

16- Substitua o conteúdo dentro das tags `<tomcat-user></tomcat-user>` por:

```
<tomcat-users>
```

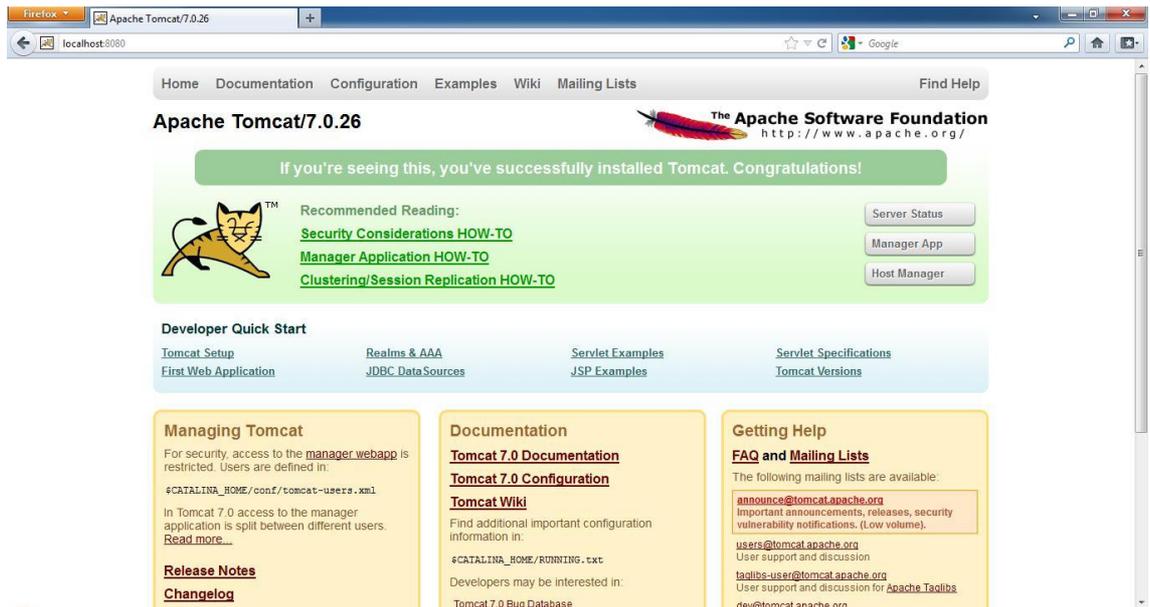
```
<role rolename="manager"/>
```

```
<role rolename="admin"/>
```

```
<user username="admin" password="z102030" roles="admin,manager"/>
</tomcat-users>
```

17- Abra o navegador e digite: <localdoservidor>:8080/;

18- O resultado deve ser esse:



Pronto. O Tomcat está rodando.

Publicando o webrun.war:

Agora iremos copiar o webrun.war do ambiente de produção e disponibiliza-lo no ambiente Linux:

- 1- Copie o webrun.war para o ambiente Linux;
- 2- Entre na pasta onde está o webrun.war;
- 3- Mova para a pasta webapps `$mv webrun.war`

`/home/<usuário>/softwell/webrun/tomcat/webapps/;`

- 4- Reinicie o serviço do tomcat `$service tomcat restart;`

A partir desse momento o nosso webrun já está rodando e funcionando perfeitamente, mas não conseguiremos acessar nossos projetos, mesmo que os nossos.wfres já estejam dentro da pasta systems. Precisamos indicar o caminho deles para o webrun.

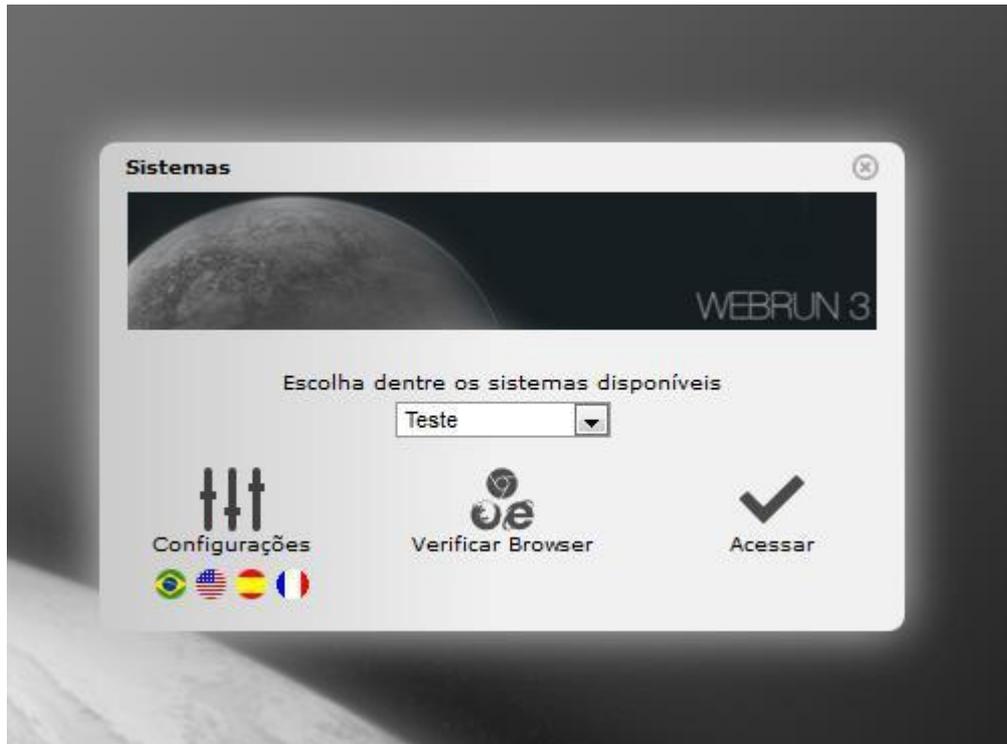
- 5- Entre no diretório `/usr/lib/ $cd /usr/lib/;`
- 6- Crie um novo arquivo nomeando webrun.ini `$sudo gedit webrun.ini;`
- 7- Escreva: `InstallDir=/home/<nome do usuário padrão>/softwell/webrun/;`
- 8- Salve;

9- Reinicie o sistema;

Agora testaremos o Webrun:

1- Abra o navegador e digite: <localdoservidor>:<portapadraowebrun>/webrun/;

2- O resultado deve ser esse:



Obs.: Todos os procedimentos vistos acima são compatíveis com distribuições baseadas no Debian. Em alguns casos será necessário, antes, instalar o repositório PPA.