

Instalação do aplicativo de atendimento (Linux)

Este manual demonstra a instalação do aplicativo de atendimento em servidores Linux. Para este tutorial será utilizado o Ubuntu 16.04.

Os pré-requisitos para instalação do aplicativo são:

- SGBD MySQL
- Python3 instalado
- Bibliotecas Python
 - pymysql
 - pillow
 - bottle
 - openpyxl
 - paste
 - requests
- Porta 8080 liberada

Atendimento aos pré-requisitos

1 – Instalando o MySQL e criando um banco de dados

Vamos iniciar a instalação do MySQL para armazenar-mos o nosso banco de dados do atendimento, para isto execute o seguinte comando no terminal:

```
sudo apt-get update && sudo apt-get install mysql-server-5.7 -y
```

Será solicitado uma senha digite-a duas vezes e confirme, agora depois de instalado o MySQL, crie o banco de dados para armazenar as tabelas do atendimento. Para o nosso exemplo o nome do banco de dados será **atendimento**. Com o seguinte comando acessamos o mysql.

```
mysql -u root -p
```

Será solicitada a senha que você configurou, digite-a e então o acesso ao mysql é liberado, um prompt como o que segue abaixo será exibido.

```
“  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.
```

```
Type 'help;' or 'h' for help. Type 'c' to clear the current input statement.
```

```
mysql> “
```

Um cursor fica piscante informando que é aguardado comando ser digitado. Digite o seguinte comando para criar o banco de dados **atendimento**.

```
CREATE DATABASE atendimento;
```

A base de dados será criada (se ela não existir) e então digite o comando abaixo para fechar o prompt do MySQL, terminamos a configuração do banco agora.

exit;

Isto encerra a etapa de criação do banco de dados, agora vamos realizar a instalação do **Python3** e as dependências do aplicativo de **atendimento**.

2 – Instalando o Python3 e as dependencias do projeto

Geralmente o python já se encontra instalado no Linux, mas para não ficarmos na dúvida instale-o e o pip (assistente de instalação de bibliotecas de terceiros) com o comando abaixo

sudo apt-get update && sudo apt-get install python3 python3-pip -y

Sua senha será solicitada, digite-a e então dê enter, em alguns minutos o python3 e o pip estarão instalados.

Depois de instalado o Python3 e o pip vamos instalar as bibliotecas de terceiros necessárias para o projeto funcionar corretamente. O comando abaixo instala todas as dependências do aplicativo de atendimento.

sudo pip3 install bottle pillow openpyxl paste requests pymysql

Depois de instaladas estas dependências todos os pré-requisitos terão sido atendidos e então poderemos passar para a configuração do aplicativo.

Baixando os fontes do projeto atendimento

O aplicativo está hospedado no github e o mesmo pode ser baixado facilmente usando o comando git clone. Primeiro verifique se tem o git instalado para poder fazer o download do fonte direto do github. Execute o comando abaixo.

sudo apt-get update && sudo apt-get install git -y

Depois de instalado o git baixe o fonte com o comando abaixo

git clone <https://github.com/plimo263/atendimento.git>

Uma pasta chamada atendimento é criada no diretório corrente, acesse ela, dentro dela vamos executar todos os passos abaixo.

Nota: Caso não queira utilizar o git me envie um e-mail que te envio os arquivos pelo e-mail.

Configuração do aplicativo de atendimento

A configuração é centralizada no arquivo **config.py** dentro da pasta do projeto que armazena as variáveis usadas para acessar o banco de dados MySQL.

Arquivo config.py

Ele contém somente algumas variáveis que são usadas para orientar o aplicativo onde ele pode encontrar o banco de dados usando o usuário e senha informado. Abaixo segue uma simples explicação de como configurar cada variável corretamente.

- **MY_USUARIO** → Local o usuário de conexão ao banco de dados deve ser informado ex: 'root'
- **MY_SENHA** → Local para informar a senha de acesso ao MySQL ex: 'marcos'
- **MY_BANCO** → Local para informar o banco de dados onde vamos salvar os dados ex: 'atendimento'

- MY_SERVIDOR → Local para armazenar o endereço IP do servidor MySQL ex: 'localhost'

Para nosso exemplo vamos configurar as variáveis dentro do config.py desta forma

```
MY_USUARIO = 'root'
```

```
MY_SENHA = 'mfelipe'
```

```
MY_BANCO = 'atendimento'
```

```
MY_SERVIDOR = 'localhost'
```

Depois de configurado este arquivo, vamos ser capazes de criar as tabelas iniciais do aplicativo e então rodar o aplicativo de fato.

Criando as tabelas do aplicativo de atendimento

Para que os atendimentos possam ser registrados, vamos criar suas tabelas para armazenamento, depois disto podemos de fato iniciar o aplicativo de atendimento.

Para criar as tabelas existe **um assistente dentro do diretório do projeto chamado Instalacao.py**, ele é responsável por fazer a criação das tabelas usando as informações contidas no **config.py** (por isto ele é configurado primeiro).

Executando o seguinte comando no terminal você consegue pedir para o Instalador criar as tabelas do aplicativo de atendimento

python3 Instalacao.py instalar

Se bem sucedido uma mensagem de *** TABELAS CRIADAS COM SUCESSO** será exibida, a estrutura de tabelas já estará criada. Agora necessita apenas iniciar o projeto.

Iniciando o aplicativo de atendimento

O aplicativo de atendimento pode ser iniciado na linha de comando estando dentro do diretório do projeto, executando o seguinte comando:

python3 controlador.py

Depois disto o aplicativo vai estar sendo executado se todos os passos anteriores foram feitos corretamente. Na porta **8080** da máquina onde ele foi instalado pode-se ser acessado pelo navegador digitando o endereço **http://<IP_DO_SERVIDOR>:8080**. Você verá uma tela como esta se tudo ocorreu corretamente.

**USUARIO:****SENHA:****Acessar**

Para acessar digite o seguinte usuario e senha:

Usuario: admin

Senha: \$ODamemodb

Se conseguiu fazer o acesso com sucesso isto indica que o aplicativo foi instalado corretamente e vai estar disponivel pelo servidor que foi configurado na porta 8080. Agora será necessário criar usuarios de loja, cadastrar vendedores, divulgadores e outros ajustes mais informados em outros tutoriais. Caso queira desativar o aplicativo (coloca-lo offline novamente) no terminal segure o **CTRL e pressione a tecla C** isto vai tirar o aplicativo de ação e devolverá o terminal para você.

Colocando o aplicativo em segundo plano

Para colocar o aplicativo como serviço é necessario um comando parecido com o que foi digitado acima para faze-lo funcionar, mas neste caso você coloca ele para executar em segundo plano liberando o terminal e deixando-o ativo para uso. Digite o comando abaixo:

python3 controlador.py 2&>1 > /dev/null &

Um número será retornado, este é o número do processo, caso precise encerrar o aplicativo digite o comando **kill** seguido do **numero do processo** que foi exibido, como segue o comando abaixo

sudo kill 12584

Autor: Marcos Felipe da Silva Jardim