



Programação de Sistemas para Internet

Prof. Romerito Campos

Plano de Aula

- Objetivo:
 - Compreender o envio de email em aplicações Flask

Conteúdos

- Configurações de Email
 - Mailtrap
 - Gmail
 - Outlook
- Emails com a biblioteca smtplib
 - Configurações
- Emails com a extensão Flask-Email
 - COnfigurações

Email

Email

- O email é uma forma de comunicação bastante empregada na Internet e existe desde um período muito anterior a existência da Web.
- Muitas aplicações na Web utilizam serviços de email para diversos tipos de notificação.
- Um exemplo notório é o de confirmação de cadastro de usuários
- Notificações em geral são enviadas também por emails

Email

- O envio de email consiste de três protocolos da web que são SMTP, POP3 e IMAP ([veja detalhes](#)).
- O SMTP age como transportador de email do seu servidor de email para outros servidores de emails, que podem estar no mesmo domínio ou domínios diferentes.
 - Se você usa conta do gmail, o domínio do seu servidor de email é `smtp.gmail.com`
- O IMAP e o POP3 faram o papel de entrega dos emails enviados.

Email

- É comum encontrar nas mais variadas linguagens de propósito geral como Python o suporte ao envio de emails.
- A linguagem de um pacote embardado (built-in) na instalação padrão para o envio de emails. Trata-se do [smtplib](#)
- Desta maneira, é possível enviar emails usando um módulo nativo do python

Email

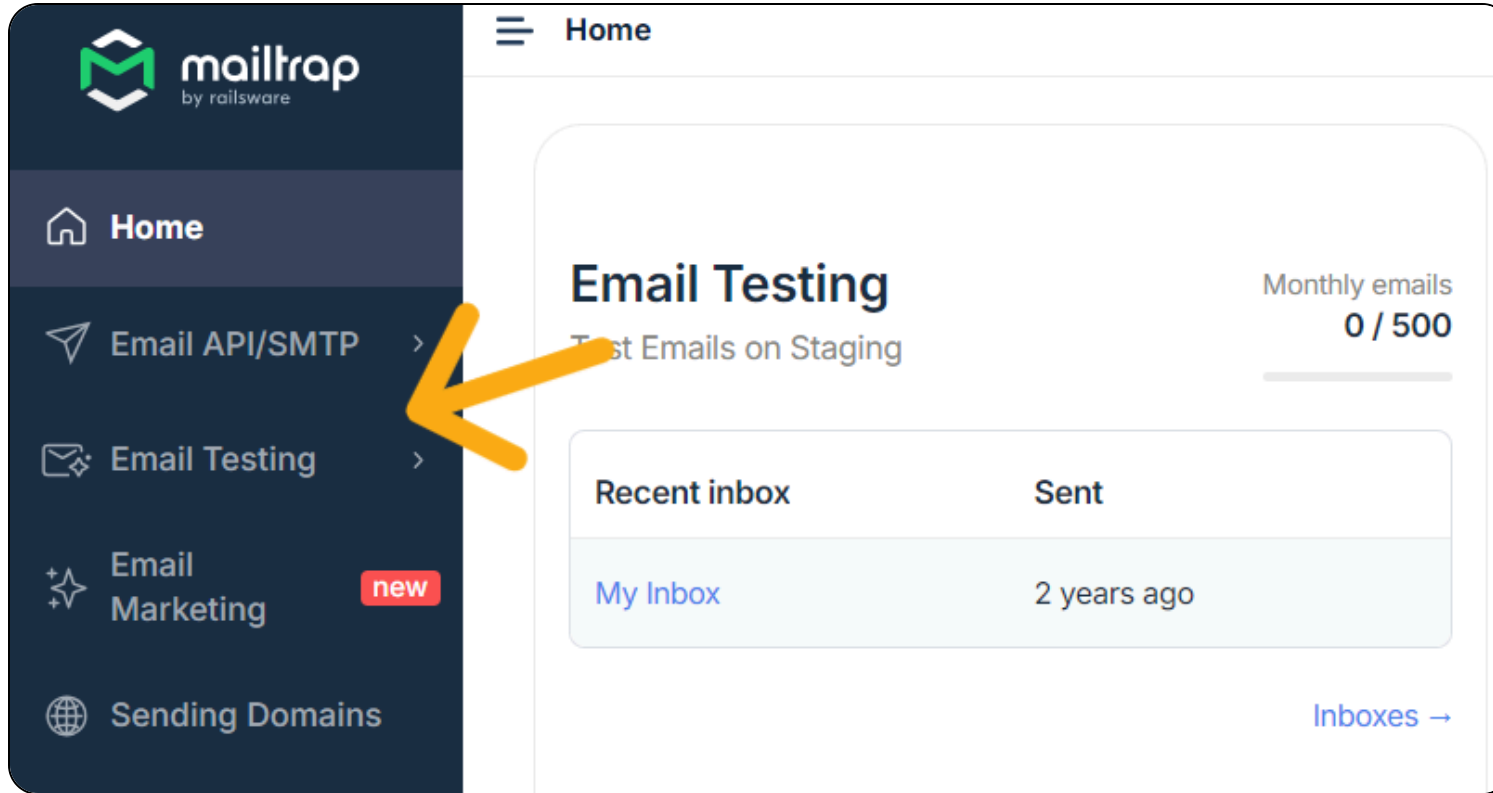
- Ainda sobre as conexões SMTP, devemos levar em consideração que é necessário adicionar uma camada de segurança o envio de emails
- É possível proteger o envio de emails usando. Dois protocolos são utilizados para tal finalidade: SSL (Secure Socket Layer) e TLS (Transport Layer Security)

Servidor de Testes - Mailtrap

Servidor de Testes - Mailtrap

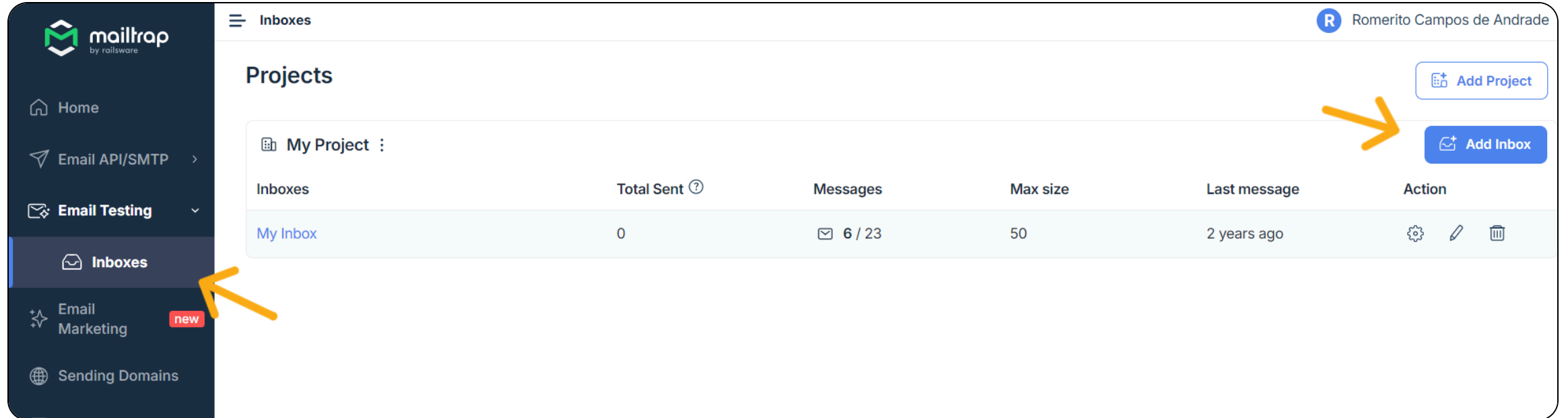
- Para realizar o envio de emails é necessário ter uma conta de algum serviço de email (contas gmail, outlook etc)
- Entretanto, podemos utilizar serviços na web que oferecem serviços de SMTP de modo que podemos testar o envio de emails para uma conta fictícia gerenciada pelo serviço
- Vamos testar o envio de emails com o [Mailtrap](#)

- Ao criar uma conta teremos acesso ao painel de controle:



- **A seta indica a opção de teste de emails**

- Temos uma caixa de entrada (**inbox**). Podemos adicionar outras.



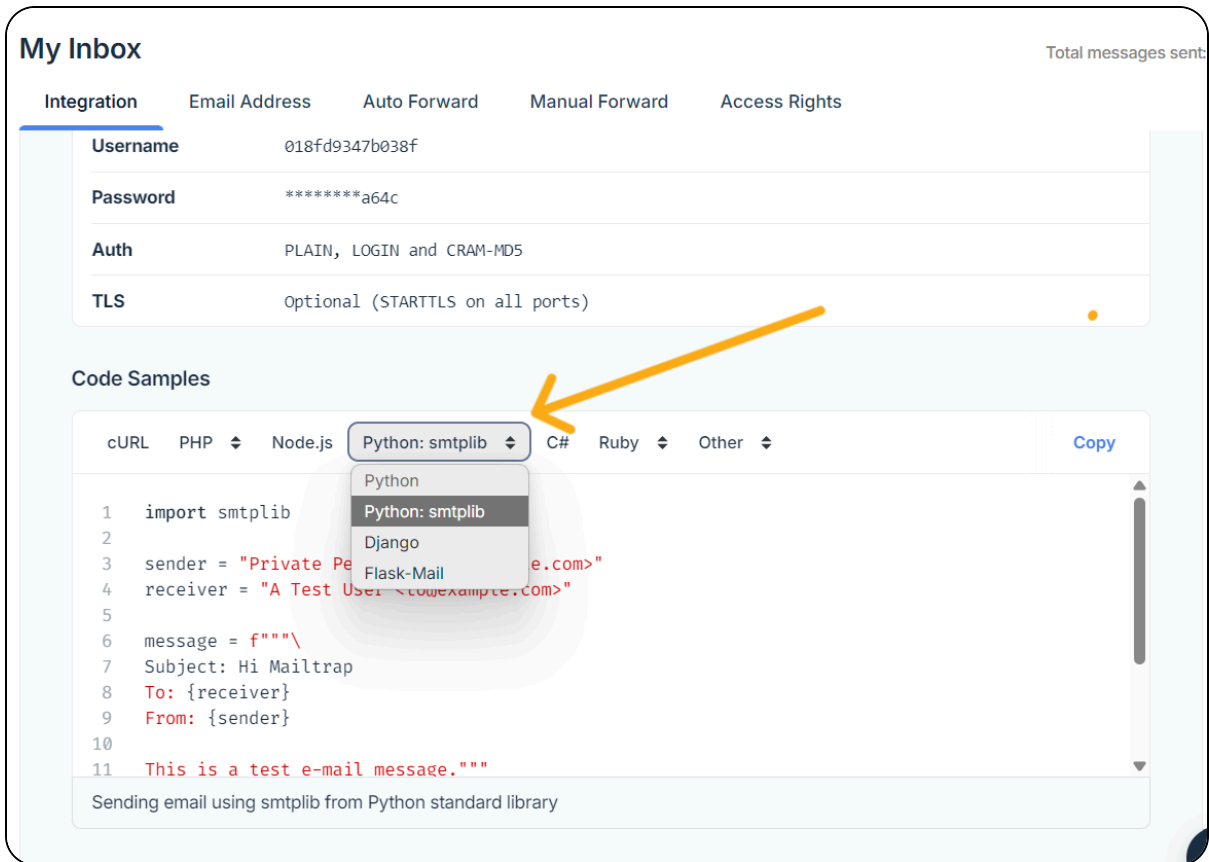
- Estas caixas de email serão utilizadas para vermos os emails enviados pela aplicação ou scripts de testes.

- Veja exemplos de email na caixa e também os dados da configuração SMTP (podem ser redefinidos)

The screenshot displays the Mailtrap web interface. On the left is a dark sidebar with navigation links: Home, Email API/SMTP, Email Testing, Inboxes (selected), Email Marketing, Sending Domains, Templates, Billing, and Settings. The main content area is titled 'My Inbox' and shows a list of emails, all with the subject 'Tarefa Criada'. An orange arrow points to the first email in the list. Below the inbox list, the 'Integration' section is visible, with tabs for SMTP, API, and POP3. The SMTP tab is active, showing a table of credentials. An orange arrow points to the 'SMTP' tab. The credentials table includes fields for Host, Port, Username, Password, Auth, and TLS.

Credentials	
Host	sandbox.smtp.mailtrap.io
Port	25, 465, 587 or 2525
Username	018fd9347b038f
Password	*****a64c
Auth	PLAIN, LOGIN and CRAM-MD5
TLS	Optional (STARTTLS on all ports)

- Na mesma página do slide anterior, temos as configurações aplicadas a diversas tecnologias:



- É importante ressaltar que podemos configurar contas do Gmail e outlook para envio de emails e criar contas de email para uso profissional
- Vejamos os primeiros testes com a biblioteca padrão smtplib do python

Exemplo 1

Envio de Emails SMTPLIB

Exemplo 1

- O código deste exemplo pode ser acessado [aqui](#).
- Há apenas dois arquivos que são:
 - `sender.py`
 - `requirements.txt`
- Um terceiro arquivo é utilizado no projeto `.env`, mas ele é omitido.
 - Este arquivo guarda as credenciais de acesso ao email do mailtrap

- Abaixo segue um exemplo do arquivo `.env` que não está no github.

```
SMTP_SERVER=sandbox.smtp.mailtrap.io  
PORT=2525  
SENDER=casdsauijk  
PASSWORD=*****dsds
```

- Os dados devem ser substituídos pelos que estão presentes na sua conta do mailtrap.
- Não adicionar aspas no valor atribuído as chaves.

- Além disso, é necessário instalar o módulo [python-dotenv](#) para ler as variáveis presentes em `.env`

```
pip install python-dotenv
```

- Levando em consideração essas definições iniciais, vejamos como poderemos enviar os nossos emails utilizando um script simples.

1. Importe os módulos que vamos precisar:

```
from smtplib import SMTP  
import dotenv  
import os
```

2. Carregando as variáveis de ambiente

```
dotenv.load_env()
```

3. Definimos na variável `message` uma mensagem a ser enviada por email

```
message = f"""\n
Subject: Olá\n
To: {os.getenv('SENDER')}\n
From: {os.getenv('SENDER')}\n
\n
Alguma coisa escrita aqui."""
```

4. A próxima etapa é enviar de fato o email

```
with SMTP(host=os.getenv('SMTP_SERVER'), port=os.getenv('PORT')) as server:  
    server.starttls()  
    server.login(os.getenv('SENDER'), os.getenv('PASSWORD'))  
    server.sendmail(os.getenv('SENDER'), os.getenv('SENDER'), msg=message)
```

- Criamos uma instância do servidor SMTP: `server`.
- Em seguida, adicionamos a função `starttls` é chamada para criptografar a conexão.
- O usuário é logado usando as credenciais no arquivo `env`.
- O email é enviado usando `sendmail`.

- O resultado obtido é:

