

Documentação automação de testes SiaPdv

Ferramentas utilizadas

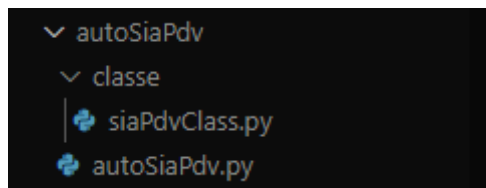
- Python – 3.12
- Flet – 0.22.0
- Pyautogui – 0.9.54
- Pywinauto – 0.6.8
- Time(Já acompanha o Python)

OBS: Todos as ferramentas adicionais foram instaladas com o pip

Exemplo: pip install Biblioteca==versão (Em um cmd)

Estrutura da pasta do projeto

Foi utilizado uma estrutura mais simples, usando apenas uma pasta "classe" para separar os arquivos de back e front-end, sendo o de front mantido na pasta raiz e o back na pasta "classe".



Arquivo siaPdvClass

O código se baseia nessas linhas de código, foram usados os mesmos comandos em todas as automações.

- A importação das ferramentas que foram utilizadas na automação.

```
from pywinauto import Desktop, Application
import pyautogui
import time
```

- Utilização do pywinauto, importante para evitar erros durante a execução dos testes, a biblioteca está sendo usada para localizar janelas abertas no computador e coloca-las em foco, assim evitando que os comandos sejam efetuados na janela errada, foi utilizado esse método para todas as janelas.

```
def selecionaSiaPdv():
    try:
        app = Application(backend="win32").connect(title="SIA PDV - Criare Tecnologia") # Conecta na janela pelo título
        main_window = app.window(title="SIA PDV - Criare Tecnologia") # Obtém a referência para a janela principal
        main_window.set_focus() # Traga a janela para o foco na tela
        pyautogui.click()
        # Verifica se a janela está realmente ativa após a conexão
        if main_window.is_visible():
            return True # Retorno indicando sucesso
        else:
            print("Tela não visível")
            return False
    except Exception as e:
        # Tratamento de exceção caso não consiga conectar
        print(f"Erro ao tentar conectar a tela: {e}")
        return False
```

- Utilização do pywinauto para localizar alerts e dialogs na tela, pywinauto também foi usado para localizar mensagens de alerta ou informações na tela.

```
def travaDialog():
    app = Application(backend="win32").connect(title="SIA PDV - Criare Tecnologia") # Conecta na janela pelo título
    main_window = app.window(title="SIA PDV - Criare Tecnologia") # Obtém a referência para a janela principal
    main_window.set_focus() # Traga a janela para o foco na tela
    pyautogui.click()
    textoTela = app.top_window()
    texto_desejado = "Informação"
    try:
        all_text = textoTela.window_text()
        print(all_text)
        if texto_desejado in all_text:
            print(f"O texto '{texto_desejado}' foi encontrado na tela.")
            return True
        else:
            print(f"O texto '{texto_desejado}' não foi encontrado na tela.")
            return False
    except Exception as e:
        print(f"Erro ao tentar encontrar o texto: {str(e)}")
```

- Método utilizado com o pyautogui para fazer todas as automações, a biblioteca pyautogui não faz ações diretamente nos elementos do siaPdv, ele apenas simula o pressionamento de uma tecla no teclado por exemplo, indiferente de qual janela esteja em foco ou de qual caixa de texto esteja selecionado.

```
def vendaPadrão(self, intervaloTecla, intervaloExec, pararLoop, contador):
    try:
        formaPag = 1
        venda = 2
        while not pararLoop:
            contaVenda = 1
            if formaPag > 13:
                formaPag = 1
            if contador == 1:
                pdvTela = SiaPdvClass.selecionaSiaPdv()
                if pdvTela == False: # Chama a função selecionaMultiPdv para verificar a visibilidade da janela
                    err = "Erro ao tentar encontrar a janela do sia PDV"
                    yield "err", err
                    break
                time.sleep(intervaloTecla)
                pyautogui.press("F2") # abre o caixa
                time.sleep(intervaloTecla)
                travaCaixa = SiaPdvClass.travaDialog()
                if travaCaixa:
                    pyautogui.press("1") # confirma a abertura
                    time.sleep(intervaloTecla)
                    pyautogui.press("escape") # pula o alert de de suprimento
                    time.sleep(intervaloTecla)
                    pyautogui.press("enter") # seleciona a forma de pagamento do suprimento(por enquanto seleciona a primeira)
                    time.sleep(intervaloTecla)
                    pyautogui.typewrite("200") # digita o valor do suprimento
                    time.sleep(intervaloTecla)
                    pyautogui.press("F3") # confirma o valor
                    time.sleep(intervaloTecla)
                    pyautogui.press("1") # confirma de vez o suprimento
                travaAbre = SiaPdvClass.travaDialog()
                while travaAbre == True: # teste para esperar a operação de suprimento
                    travaAbre = SiaPdvClass.travaDialog()
```

- pyautogui.press(tecla que deseja ser apertada) *Comando usado para pressionar uma tecla do teclado*
- pyautogui.typewrite(Texto que deseja ser digitado) *Comando usado para escrever um texto*

Arquivo autoSiaPdv

Resumindo um pouco a forma de retorno entre a back e o front, a explicação do código completo consegue ser estudado pela documentação do flet(Versões futuras do flet terão mudanças).

```
match testeEscolhido:
    case "Venda padrão":
        contador = multiPdvInstancia.vendaPadrão(intervaloTeclas, intervaloExec, pararLoop, contadorAtual)
        for dados in contador:
            if pararLoop:
                break
            if dados[0] == "contador":
                contadorAtual = dados[1]
                atualizarTesteIniciado()
            else:
                erroTesteIniciado(dados[0], dados[1])
```

O código verifica o que retornou da função que vem do siaPdvClass e com base nisso ele chama outra função retornando as informações corretas para a tela.

- Informações retornadas da siaPdvClass para a autoSiaPdv.

```
yield "contador", contador# envia o valor do contador para a tela
```

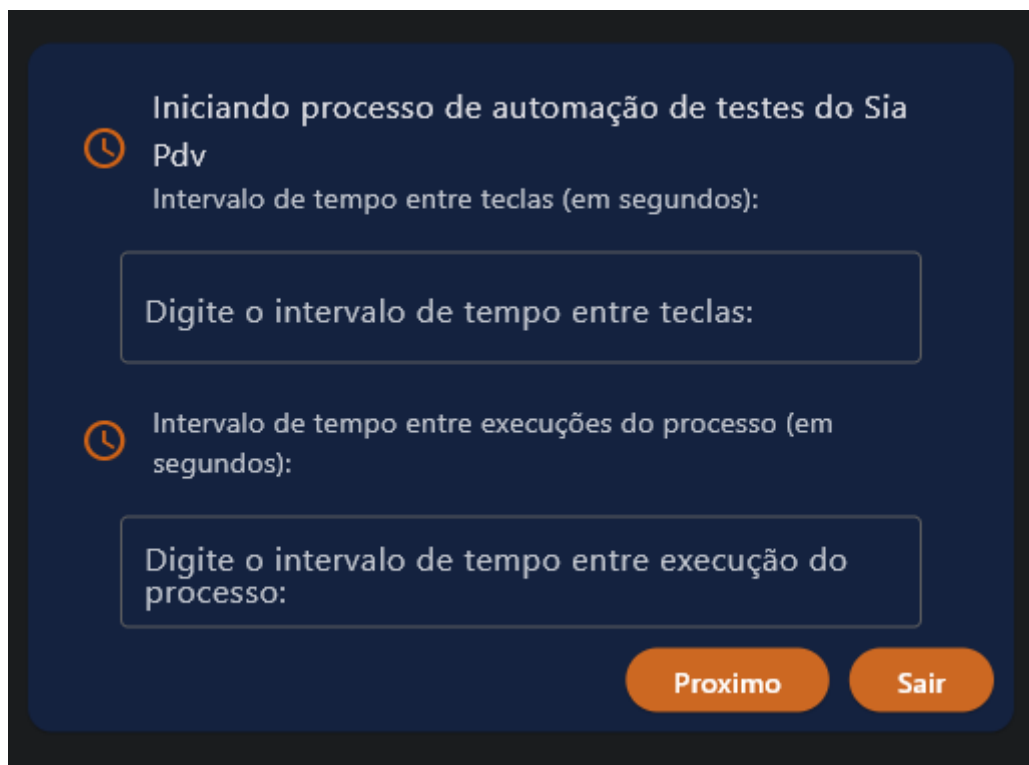
```
yield "err",type(e), e
```

Funcionamento da automação

Descompactar a pasta que foi baixada como arquivo rar ou zip.

Ao executar o .exe chamado de autoSiaPdv a interface da automação irá abrir.

OBS: Todas as automações serão feitas na janela do SiaPdv, a interface é feita apenas para controle e retorno.

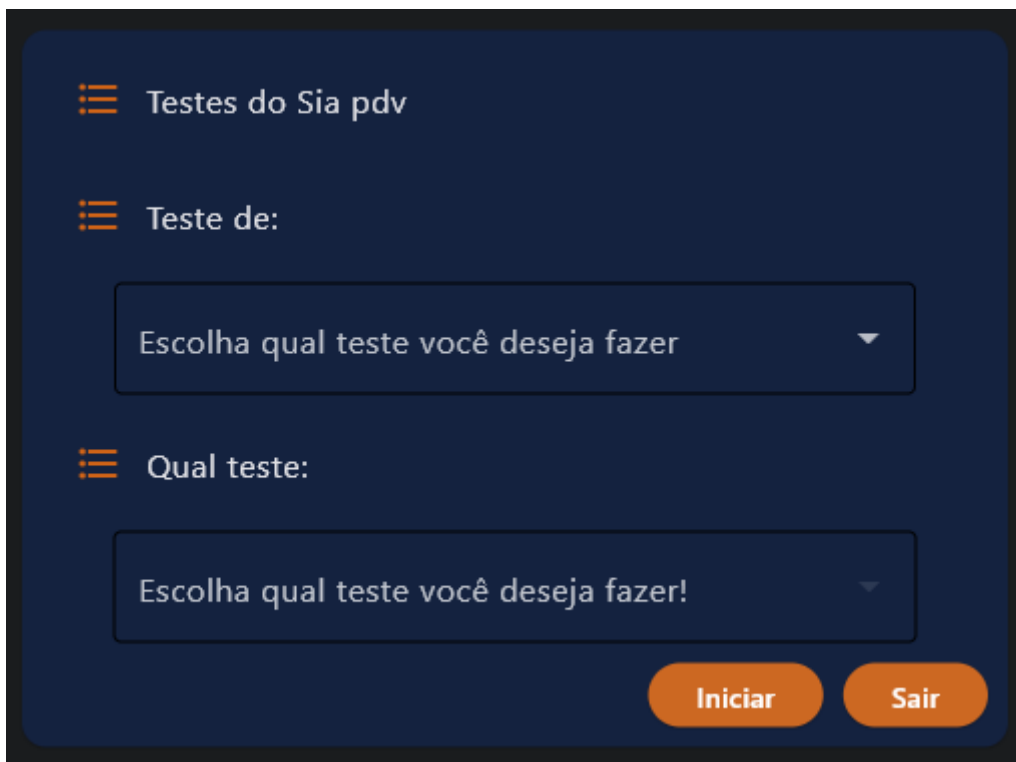


A interface de configuração da automação do SiaPdv é exibida em uma janela com fundo escuro. No topo, há um título "Iniciando processo de automação de testes do Sia Pdv" precedido por um ícone de relógio. Abaixo, há um campo de texto para "Intervalo de tempo entre teclas (em segundos):" com o placeholder "Digite o intervalo de tempo entre teclas:". Segue-se outro campo de texto para "Intervalo de tempo entre execuções do processo (em segundos):" com o placeholder "Digite o intervalo de tempo entre execução do processo:". Na base da interface, há dois botões laranja: "Proximo" e "Sair".

- Tempo de intervalo entre teclas: Tempo entre um comando e outro.
Por exemplo: Tempo entre a digitação entre um campo de texto e outro.
- Tempo entre execuções do processo: Tempo entre uma execução e outra.
Por exemplo: Tempo após salvar um cadastro para iniciar outro.

OBS: Esse tempo varia de computador para computador, a ideia é manter um tempo entre cada comando para evitar erros na execução, caso o computador esteja mais lento o intervalo entre teclas maior é recomendado.

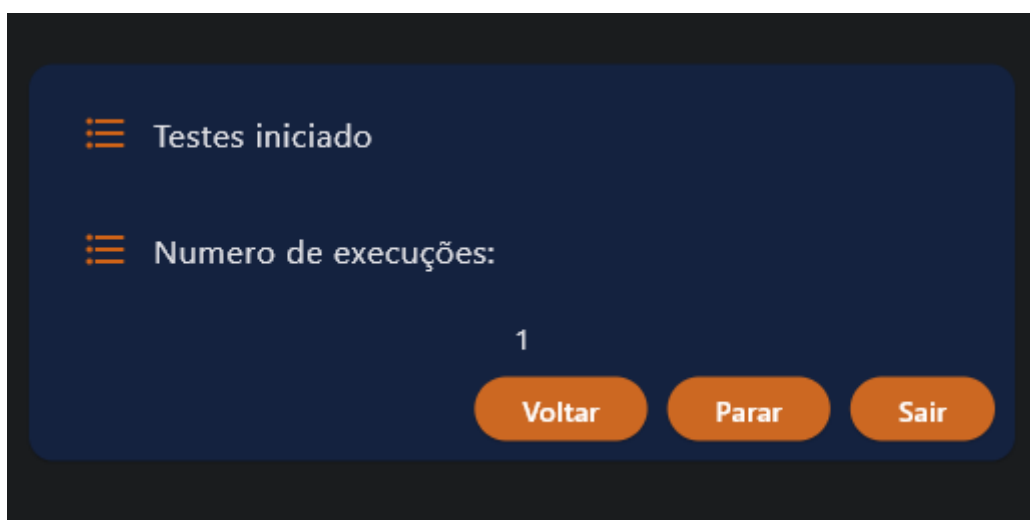
Após inserido os intervalos, na interface irá abrir uma tela com 2 campos a serem selecionados.



A primeira opção vai alterar a aba que o teste será feito, por exemplo: Venda, e a segunda opção irá alternar entre os tipos de teste, por exemplo: Selecionado venda a segunda opção irá ficar disponível com as opções de venda padrão, venda de produtos e etc. Essas opções são o que vão dizer qual teste você deseja fazer.

Após selecionada as opções e apertado o botão iniciar, o teste escolhido irá iniciar automaticamente e uma tela será mostrada na tela.

Importante: Durante a execução do teste é de extrema importância que a tela do siaPdv se mantenha em foco, então durante o teste não será possível trocar para outra janela, isso pode causar problemas em outras aplicações e também irá causar no teste que está sendo realizado.



O valor retornado no meio da tela, é o número de execuções que o programa fez, sendo mostrado o que está sendo executado, por exemplo: se o programa se encontra no número 5 ele está no momento executando o teste pela 5 vez.

O botão de sair irá fechar a interface e irá finalizar o teste após as ações dentro do teste acabarem, por exemplo na execução de uma venda o teste só termina após o programa finalizar a venda e voltar para a tela de frente de caixa

OBS: Caso o teste esteja demorando muito para acabar você pode colocar o mouse na parte superior esquerda da tela até o teste se encerrar.

O botão de parar é utilizado para parar o teste, **lembrando** que o teste só irá parar após concluir **todas as ações**, por exemplo: quando uma venda for finalizada.

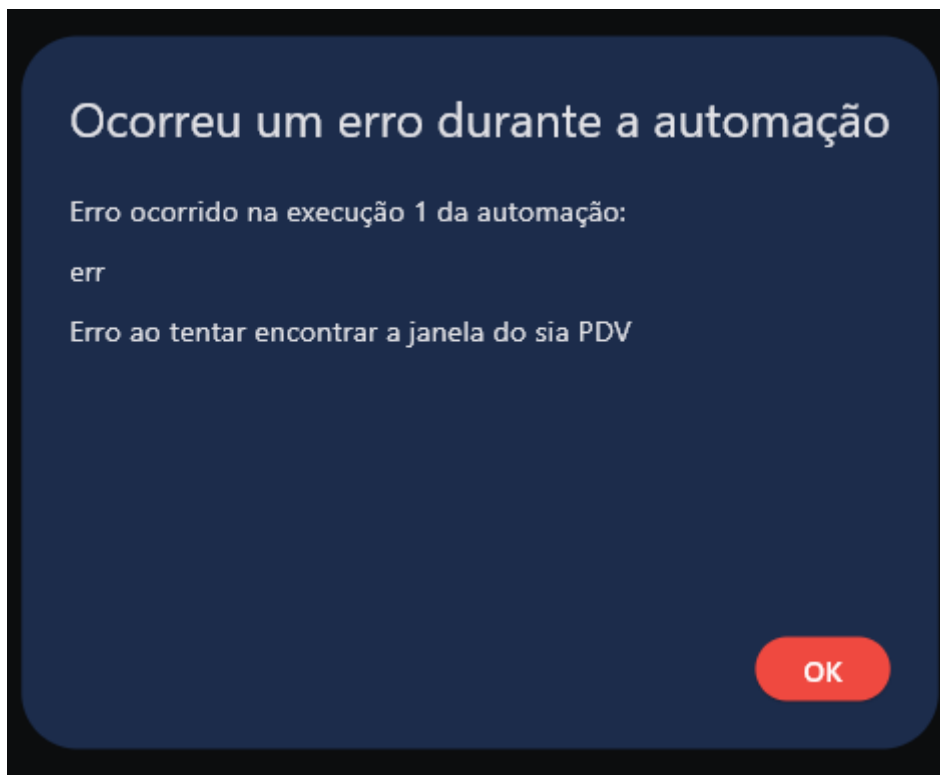
Após apertar o botão de parar ele será substituído pelo botão continuar, ele retornará o teste na última execução realizada, é recomendado apertar o botão de continuar após o teste realmente estiver interrompido, ou seja, já com todas as ações realizadas.

Outra opção para parar o teste que é o mais recomendado nesse caso é de mover o mouse para a parte superior esquerda da tela, até o teste ser parado.

OBS: O final de um teste varia de teste para teste, então em caso de dúvida espere nenhuma ação estar sendo feita.

O botão de voltar irá voltar para a tela de seleção de testes, para caso a seleção de outro teste seja necessário, o teste que estiver ocorrendo no momento será interrompido automaticamente, seguindo as mesmas especificações do botão de parar, que o teste só será interrompido após concluir todas as ações, é recomendado que espere o teste atual ser concluído antes de iniciar um novo teste para evitar erros na execução e possível travamento do programa.

Em caso de um erro que irá interromper a execução uma mensagem será exibida na tela, como essa.



Após apertar em ok, a interface irá se redirecionar para a tela de teste iniciado já com o botão de continuar.

Em casos de o teste sair da sequência correta de execução, ocorrer algum erro e nada ser retornado pare o teste imediatamente, usando o botão na interface e aguardando o teste ser finalizado, ou

movendo o mouse para a parte superior da tela, e reinicie o teste, caso se manter contate quem estiver cuidando das automações.

Não se assuste com os erros, em muitos casos foram apenas erros ocorridos na execução do teste, tente reiniciar o teste, ou usando o botão de volta e escolhendo o teste novamente ou usando o botão de continuar, em caso o erro persista, contate quem estiver cuidando das automações no momento.

Anexos

Caso tente para o teste usando canto superior esquerdo da tela essa mensagem irá aparecer, é apenas uma confirmação que o teste foi interrompido.

