

**CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER
CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

FELIPE CRISPIM DE MATTOS – 4193976

**PROPOSTA DE ATIVIDADE PARA AUTOMATIZAR A COMUNICAÇÃO ENTRE
INSTITUIÇÕES DE ENSINO E RESPONSÁVEIS.**

MOGI-GUAÇU

2023

FELIPE CRISPIM DE MATTOS – 4193976

**PROPOSTA DE ATIVIDADE PARA AUTOMATIZAR A COMUNICAÇÃO ENTRE
INSTITUIÇÕES DE ENSINO E RESPONSÁVEIS.**

Pesquisa apresentada ao curso de análise e desenvolvimento de sistemas, apresentado na disciplina de Atividade Extensionista I do Centro Universitário Internacional UNINTER.

MOGI-GUAÇU

2023

SUMÁRIO

1	INTRODUÇÃO.....	3
2	DESENVOLVIMENTO.....	4
3	METODOLOGIA	5
4	RESULTADOS.....	8
5	CRONOGRAMA.....	10
6	CONCLUSÃO.....	11
	REFERÊNCIAS.....	12
	ANEXO.....	13

1 INTRODUÇÃO

Este estudo tem como proposta aprimorar a qualidade e a eficiência da comunicação feita entre as escolas de cursos livres (Ex: idiomas, informática) e os responsáveis pelos alunos de menor idade que as frequentam, através da automatização e digitalização de processos, resultando em uma melhor infraestrutura, educação de qualidade e um trabalho decente com um possível crescimento econômico.

Foi observado que muitas escolas não conseguem realizar uma comunicação de qualidade presencialmente com os responsáveis de seus alunos. Seja por dificuldades de deslocamento geográfico fora do horário de aula, ou mesmo por falta de tempo hábil para estar presente. Hoje, parte deste processo de comunicação digital é realizado através do aplicativo WhatsApp, porém esta atividade se apoia muito no input manual de informações (notas, comportamento, ou mesmo atividades complementares) do setor administrativo da escola, visto que o setor pedagógico possui pouco tempo para enviar mensagens para cada responsável, mas o mesmo setor por vezes acaba não possuindo o mesmo tempo e informações acabam sendo perdidas ou por muitas vezes não enviadas.

Este projeto se propõe em organizar as informações de alunos e responsáveis que as escolas já possuem em novas planilhas organizacionais, criando um pequeno banco de dados, e automatizar o envio de informações de cada aluno via WhatsApp para os responsáveis utilizando-se da automação web realizada com a linguagem python.

2 DESENVOLVIMENTO

Iniciamos o projeto com a análise do banco de dados da escola de idiomas Wizkids, na cidade de Mogi-Guaçu, SP. Hoje a escola não possui ao menos uma planilha de cadastro de responsáveis, apoiando-se apenas no cadastro do número telefônico dos alunos em uma agenda telefônica através do celular.

Constatado a não existência de nenhum meio de armazenamento de dados eletrônicos que não fosse a agenda eletrônica do celular administrativo, foi proposto que a escola passasse a adotar o programa Microsoft Excel como sua base de dados mãe através de planilhas de apoio, afim de melhorar o controle de seus cadastros e facilitar a pesquisa dos mesmos.

Tendo uma base de cadastros consolidada e de fácil acesso, podemos dar prosseguimento com o objetivo principal, que é a automatização de envio de mensagens para os responsáveis através da automação web, diminuindo o tempo necessário para comunicação entre escola-alunos, e possibilitando uma maior personalização durante o contato.

3 METODOLOGIA

Foi proposto a criação de um modelo de banco de dados simples para armazenar os cadastros dos responsáveis realizados. O modelo de banco de dados se baseia em uma planilha de Excel que deve conter apenas 5 colunas:

- Nome Responsável - Este campo deve ser preenchido com o nome completo do responsável cadastrado (Cada aluno pode possuir mais de 1) no ato da matrícula pelos responsáveis pela matrícula (time de vendas/administrativo).
- Número para contato – Este campo deve ser preenchido com o número telefônico completo do responsável no ato da matrícula pelos responsáveis pela matrícula (time de vendas/administrativo). O número deve conter o código telefônico nacional (55) e o DDD regional conforme figura abaixo.
- Nome Aluno – Este campo deve ser preenchido com o nome do aluno no ato da matrícula pelos responsáveis pela matrícula (time de vendas/administrativo).
- Nome Professor – Este campo deve ser preenchido com o nome do professor responsável pela turma onde o aluno foi cadastrado no ato da matrícula pelos responsáveis pela matrícula (time de vendas/administrativo). Este campo foi adicionado afim de facilitar a pesquisa do corpo pedagógico (Professores) através da ferramenta de filtro na planilha, caso o professor queira que uma mensagem personalizada seja enviada a seu aluno.
- Mensagem – Este campo deve permanecer em branco no ato da matrícula, e será preenchido apenas quando o corpo pedagógico queira se comunicar com os responsáveis dos alunos.

Responsável	Número	Aluno	Professor	Mensagem
Beatriz Viana	5519996356931	Nicolas	Felipe	
Osmir Mattos	5519993561912	Bruna	Livia	
Ana Paula	5519995530021	Victoria	Felipe	
Mateus Machado	5519998231923	Rafael	Maria	

Com a planilha de banco de dados já criada e preenchida, avançamos para a parte de automatização das mensagens.

Para trabalharmos com a automatização web e realizarmos o envio das mensagens através do aplicativo WhatsApp, utilizamos a linguagem python, visto a existência da biblioteca selenium que facilita a nossa comunicação com o ambiente web.

Para desenvolvermos nosso código, utilizamos a plataforma Jupyter Notebook, e nos baseamos em 3 passos para a automatização:

- Leitura do banco de dados: Para leitura do banco de dados, nos utilizamos da biblioteca pandas.

```
#Ler banco de dados
import pandas as pd

contatos_df = pd.read_excel("Cadastro_Alunos.xlsx")
display(contatos_df)
```

	Responsável	Número	Aluno	Professor	Mensagem
0	Beatriz Viana	5519996356931	Nicolas	Felipe	NaN
1	Osmir Mattos	5519993561912	Bruna	Livia	NaN
2	Ana Paula	5519995530021	Victoria	Felipe	NaN
3	Mateus Machado	5519998231923	Rafael	Maria	NaN

- Login web WhatsApp: Para realizarmos o desenvolvimento do passo de login no web WhatsApp, nos utilizamos da biblioteca selenium. Para manipulação da url utilizamos o módulo URLLIB, assim como também utilizamos o módulo time para adicionarmos pausas entre alguns passos.

```
#fazer login no whatsapp web

navegador = webdriver.Chrome()
navegador.get("https://web.whatsapp.com/")

while len(navegador.find_elements(By.ID,"pane-side")) < 1:
    time.sleep(1)

#login feito
```

- Envio da mensagem: Com o login feito, precisamos criar um loop onde lemos as mensagens armazenadas no nosso banco de dados e selecionamos seu respectivo número de contato presente no cadastro. O código deve abrir um novo link, dessa vez iniciando uma conversa com o número selecionado, colar a mensagem lida no banco de dados e a enviar. Após o envio, o código deve esperar 5 segundos para prosseguir para a próxima mensagem.

#Ler mensagem, Armazenar Números, e enviar a mensagem

```
for i, mensagem in enumerate(contatos_df['Mensagem']):
    pessoa = contatos_df.loc[i, "Responsável"]
    numero = contatos_df.loc[i, "Número"]
    texto = urllib.parse.quote(f"Oi {pessoa}! {mensagem}")
    link = f"https://web.whatsapp.com/send?phone={numero}&text={texto}"
    navegador.get(link)
    while len(navegador.find_elements(By.ID, "pane-side")) < 1:
        time.sleep(1)
    time.sleep(2)
    navegador.find_element(By.XPATH, '//*[@id="main"]/footer/div[1]/div/span[2]/div/div[2]/div[2]/button/span').click()
    time.sleep(10)
```


4 RESULTADOS

Foi se apurado que a escola wizzkids possuía 304 alunos matriculados no dia 31/01/2023. Após levantamento realizado com a secretaria da escola, chegou-se a uma conclusão que a escola perderia aproximadamente 30 segundos por aluno para o envio de mensagens padrões (ex: comunicados de feriados) onde a mensagem pode ser copiada mas havia a necessidade de procurar manualmente pelos contatos na agenda do WhatsApp. E aproximadamente 3 minutos (180 segundos) para mensagens específicas, onde a secretaria precisava entrar em contato com o professor responsável, copiar a mensagem que o mesmo enviou a secretaria e só assim podendo procurar pelo contato na agenda.

ALUNOS MATRICULADOS	TEMPO GASTO POR ENVIO DE MENSAGEM (EM SEGUNDOS)
304	210

FIGURA 1: DESCRIÇÃO DE ALUNOS MATRICULADOS E DE TEMPO MÉDIO DE ENVIO DE MENSAGENS PADRÕES MAIS TEMPO MÉDIO DE ENVIO DE MENSAGENS ESPECÍFICAS

Tendo o tempo médio gasto por envio de mensagem na medida de segundos, a transformamos para descobrir quanto tempo levaríamos para enviar as mensagens para cada aluno em horas, utilizando um cálculo básico de número de aluno matriculados x tempo médio gasto por envio de mensagem.

ALUNOS MATRICULADOS	TEMPO GASTO POR ENVIO DE MENSAGEM (EM SEGUNDOS)	TEMPO GASTO x ALUNOS (HORAS)
304	210	18

FIGURA 2: DESCRIÇÃO DE CÁLCULO EM HORAS GASTAS PARA ENVIAR MENSAGENS PARA TODOS OS ALUNOS DURANTE 1 SEMANA

A frequência de envio de mensagens da escola é de apenas 1 vez por semana, portanto, para mensurar o tempo perdido mensalmente, partimos do cálculo de tempo gasto por envio x frequência semanal.

TEMPO GASTO x ALUNOS (HORAS)	FREQUENCIA DE ENVIO POR SEMANA	TEMPO GASTO POR MÊS (HORAS)
18	1	71

FIGURA 3: DESCRIÇÃO DE CÁLCULO EM HORAS GASTAS PARA ENVIAR MENSAGENS PARA TODOS OS ALUNOS DURANTE 1 MÊS

Por fim, também mensuramos o tempo gasto anualmente em horas, e quantos dias de trabalho seriam equivalentes a estas horas levando em consideração a jornada de trabalho de 8 horas.

TEMPO GASTO POR MÊS (HORAS)	TEMPO GASTO POR ANO (HORAS)	TEMPO GASTO EM DIAS DE TRABALHO (8hs)
71	851	106

FIGURA 3: DESCRIÇÃO DE CÁLCULO EM HORAS GASTAS PARA ENVIAR MENSAGENS PARA TODOS OS ALUNOS DURANTE 1 ANO.

Com a automatização do processo de envio de mensagens, observou-se que a escola passou não apenas a ter um controle maior e maior possibilidade de personalização de suas mensagens, como também passou a ganhar aproximadamente 851 horas anuais em otimização de trabalhos. Horas estas que podem ser alocadas em outros esforços como melhoria de processos.

5 CRONOGRAMA

Descrição	Data Inicial	Data Final
Etapa 1 - Levantamento e análise do modelo de banco de dados atual	10/01/2023	14/01/2023
Etapa 2 – Aplicação de novo modelo de armazenagem de dados	15/01/2023	01/02/2023
Etapa 3 - Desenvolvimento do código de automação do envio de mensagens	01/02/2023	09/02/2023
Etapa 4 - Aplicação do código no dia a dia escolar	10/02/2023	27/02/2023
Etapa 5 – Levantamento de resultados	28/02/2023	28/02/2023

Fonte: O autor (2023)

6 CONCLUSÃO

Após análise dos resultados, podemos concluir que mesmo apesar de simples, o processo de automatização do envio de mensagens no ambiente escolar traz uma otimização de tempo considerável (no caso desta escola, de aproximadamente 100 dias de trabalho ao fim do ano), resultando não só em uma melhor, mais rápida e mais eficiente comunicação entre escola e responsáveis, como eventualmente podendo trazer um impacto financeiro positivo para a empresa, visto que ao reduzir a carga de trabalho, a empresa pode realocar seu funcionário para uma outra função que talvez fosse necessária ser preenchida por outra pessoa.

REFERÊNCIAS

SELENIUM WITH PYTHON. Disponível em: <https://selenium-python.readthedocs.io/index.html>. Acesso em: 01 fev. 2023.

PANDAS USER GUIDE. Disponível em: https://pandas.pydata.org/docs/user_guide/index.html. Acesso em: 29 jan. 2023.

GETTING STARTED WITH WEB AUTOMATION USING SELENIUM WITH PYTHON. Disponível em: <https://www.section.io/engineering-education/web-automation-with-selenium-using-python/>. Acesso em: 10 jan. 2023.

ANEXO

Script considerado para abrir o navegador Chrome, logar no WhatsApp, fazer a leitura da base de dados e enviar as mensagens:

```
#ler banco de dados
import pandas as pd

contatos_df = pd.read_excel("Cadastro_Alunos.xlsx")
display(contatos_df)

from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
import time
import urllib

#fazer login no whatsapp web

navegador = webdriver.Chrome()
navegador.get("https://web.whatsapp.com/")

while len(navegador.find_elements(By.ID,"pane-side")) < 1:
    time.sleep(1)

#login feito

#Ler mensagem, Armazenar Números, e enviar a mensagem

for i, mensagem in enumerate(contatos_df['Mensagem']):
    pessoa = contatos_df.loc[i, "Responsável"]
```

```
numero = contatos_df.loc[i, "Número"]
texto = urllib.parse.quote(f"Oi {pessoa}! {mensagem}")
link = f"https://web.whatsapp.com/send?phone={numero}&text={texto}"
navegador.get(link)
while len(navegador.find_elements(By.ID,"pane-side")) < 1:
    time.sleep(1)
time.sleep(2)

navegador.find_element(By.XPATH,'//*[@id="main"]/footer/div[1]/div/span[2]/div/div[2]
]/div[2]/button/span').click()
time.sleep(10)
```