July 2nd, 2025 at 20:24 UTC



**Plagiarism Detection Report** 

JustDone Al

# **Plagiarism Detection**



Plagiarism Types	Text Coverage	Words
<b>O</b> Identical	0%	0
Minor Changes	0%	0
Excluded		
Omitted Words		0

**Total Pages** 

Total Words

2627









# **Plagiarism**

0%

# Results (0)



Plagiarism Types	Text Coverage	Words
<b>O</b> Identical	0%	0
Minor Changes	0%	0
Excluded		
Omitted Words		0

#### About Plagiarism Detection

Our Al-powered plagiarism scans offer three layers of text similarity detection: Identical, Minor Changes, and Paraphrased. Based on your scan settings we also provide insight on how much of the text you are not scanning for plagiarism (Omitted words).

Identical

One to one exact word matches. Learn more

# Minor Changes

Words that hold nearly the same meaning but have a change to their form (e.g. "large" becomes "largely"). <u>Learn more</u>

# Omitted Words

The portion of text that is not being scanned for plagiarism based on the scan settings. (e.g. the 'Ignore quotations' setting is enabled and the document is 20% quotations making the omitted words percentage 20%) <u>Learn more</u>

## Copyleaks Shared Data Hub

Our Shared Data Hub is a collection of millions of user-submitted documents that you can utilize as a scan resource and choose whether or not you would like to submit the file you are scanning into the Shared Data Hub. Learn more

#### Filtered and Excluded Results

The report will generate a complete list of results. There is always the option to exclude specific results that are not relevant. Note, by unchecking certain results, the similarity percentage may change. <u>Learn more</u>

# **Current Batch Results**

These are the results displayed from the collection, or batch, of files uploaded for a scan at the same time. Learn more

#### 1.MOTIVO DA ESCOLHA DO OBJETO DE ESTUDO

O Poupatempo representa um avanço significativo no atendimento ao público paulista, centralizando serviços de diversos órgãos públicos, tais quais a Sabesp e o Detran SP, passando por mais de 400 serviços relacionados à saúde, moradia, documentos pessoais, atendimento a consumidores e demais áreas. Porém, apesar de sua grandiosa utilidade, ainda enfrenta problemas comuns quando falamos de sistemas: instabilidades, problemas técnicos e por vezes, a inoperabilidade e impossibilidade de realização de alguns serviços pontuais, conforme o sistema específico utilizado pelo serviço.

Esses problemas afetam diretamente os usuários finais, que apenas descobrem sobre a inoperabilidade do sistema e impossibilidade da realização de seus serviços no polo de atendimento presencial, após terem se locomovido, gerenciado sua agenda e adiado outras demandas.

Logo, o motivo da escolha deste objeto de estudos vai muito além de sua natureza óbvia de exercitação do pensamento crítico, da criatividade e da aplicação dos conhecimentos obtidos através das matérias referentes à banco de dados, mas também é feito pensando na resolução parcial dessas problemáticas já citadas, com objetivo de idealizar um sistema de banco de dados capaz de armazenar informações de agendamentos, relacionar com os sistemas necessários e seus status. Também, estendemos brevemente a pesquisa para abranger outras questões úteis para resolução do problema raiz.

### 2.ESTRATÉGIAS DE ANÁLISE DO OBJETO

A principal origem da análise se dá em um episódio pessoal, em determinado momento em que me organizei para locomoção ao Poupatempo de Campinas, esperei pelo meu atendimento com os documentos em mãos e fui surpreendido pela atendente que me informou sobre a indisponibilidade do sistema (SEI), impossibilitando a realização do serviço pretendido. A partir desse evento, busquei por outros relatos semelhantes na internet, encontrando reclamações registradas no site Reclame Aqui (RECLAME AQUI, 2024) e uma reportagem publicada pelo G1 (RODRIGUES, 2025), evidenciando que essa falha não se trata apenas de um caso isolado. Então, ao entender a questão como uma problemática que atinge milhares de paulistanos a muito tempo, o objeto de estudos mostra-se realmente digno e necessário de ser avaliado. Agora, o foco torna-se técnico: O problema é existente e pertinente, mas como resolvê-lo? Partimos para pesquisas mais profundas em materiais bibliográficos e técnicos que tratam sobre diferentes conceitos que se mostravam úteis para a resolução de conflitos como esse. Autores como Ayooluwa Isaías e instituições como Amazon Web Services (2025) e Microsoft (2025) foram fundamentais para entender boas práticas de verificação de integridade de sistemas, arquitetura baseada em eventos e uso de APIs RESTful. Infelizmente, não consegui contato direto com representantes do Poupatempo ou da Prodesp (responsável pelo sistema atual do Poupatempo) devido ao prazo de resposta institucional (30 dias, prorrogáveis), foi possível deduzir a estrutura do sistema atual com base em conhecimento previamente adquirido em disciplinas como Banco de Dados e Estrutura de Dados. Também, envolvemos nessa análise a comparação de práticas de engenharia de confiabilidade de sites (SRE), amplamente utilizadas em plataformas como a AWS (2025), e o estudo de possíveis soluções para o monitoramento contínuo. Assim, apesar de concentrarmos a pesquisa nos conceitos de Banco de Dados, podemos prever ou traçar caminhos aos quais as outras diferentes frentes seriam importantes para a completude do objeto inicialmente proposto para essa solução.

# 3.CONSIDERAÇÕES CRÍTICAS E CRIATIVAS

Sabemos que não é de hoje que sistemas enfrentam problemas relacionados a instabilidades; com certeza todos já enfrentaram algum momento em que nos organizamos para resolver algum problema, mas não conseguimos realizar este por conta de problemas relacionados ao sistema que gerencia esses atendimentos.

O Poupatempo, apesar de eficaz, não fica de fora desse risco, passando por quedas, infelizmente de forma mais comum que o esperado.

Usuários, claro, reclamam, pois esses problemas só são constatados na grande maioria das vezes já no balcão de atendimento, o que representa o cancelamento de compromissos, perda de horas em seu dia e estresses psicológicos.

No Reclame Aqui, uma usuária reclama: "Poupa tempo vive sem sistema, 3a¬ vez que gasto meu tempo e dinheiro para ir até o local e está sempre sem sistema e nunca tem previsão de volta." (RECLAME AQUI, 2024).

Essa problemática, já existente há algum tempo, foi reportada inclusive pelo G1, que entrevistou diferentes pessoas que passaram por situações parecidas, como a assistente comercial Vânia Rodrigues, que contou à equipe de reportagem: "Estive lá no dia 6 e não consegui atendimento. Remarquei para o dia 7 e novamente não consegui. Só fui atendida ao voltar em um sábado. Era um documento simples que eu precisava emitir e me deu uma enorme dor de cabeça. Me parece que esses problemas têm acontecido principalmente no período da tarde' (RODRIGUES, Vânia, G1, 2024).

Levando em consideração situações como essa, pensamos na possibilidade de desenvolver um sistema de banco de dados que além de armazenar os agendamentos, é capaz de relacionar os serviços oferecidos com os diferentes sistemas necessários para realizar devidas operações, também permitindo automatizações relacionadas à checagem de status de serviços e ao basear-se na inoperabilidade, também o cancelamento de agendamentos, antes da locomoção do usuário ou ao menos apresentando essa informação já no balcão de distribuição de senhas, reduzindo significativamente o tempo "desperdiçado" na espera.

A resolução dessa problemática pode ser um assunto delicado e complicado, mas que deve ser inicialmente considerado uma extensão ao que já funciona, isto é, o sistema de agendamento que o Poupatempo já implementa. Infelizmente, devido ao curto espaço de tempo que operamos essa pesquisa, não podemos dizer com clareza sobre a estrutura e o funcionamento deste - parte por serem informações sigilosas, em razão de segurança, parte pelo prazo de atendimento (tanto da Prodesp, responsável pelo sistema, quanto do Poupatempo em si) ser de 30 dias (prorrogáveis).

Porém, a partir dos conhecimentos obtidos a partir dos cursos de Banco e Estruturas de Dados, podemos deduzir essa estrutura e iniciar a implementação a partir dessa dedução. Também, apesar de concentrarmos nossa pesquisa na área de banco de dados, nos arriscamos em apresentar uma conceituação um pouco mais robusta e mais completa, pensando em outros campos da solução, justamente para podermos demonstrar que é possível realizar e implementar essa a solução.

Para isso, é necessário ampliar o campo de estudos para além da estruturação de um banco de dados relacional, mas pensar na implementação de monitoramento de integridade de sistemas, na comunicação via API RESTful e em estratégias de contato com o usuário final.

Para a estrutura de nosso banco de dados, é fundamental utilizarmos um banco relacional, isto por conta da necessidade que temos de manter integridade e consistência em um sistema de agendamentos, algo que um SGBD relacional consegue entregar com mais eficiência. Também é de suma importância nos lembrarmos dos conceitos ACID (Atomicidade, Consistência, Isolamento e Durabilidade), presentes em sistemas relacionais, garantindo que as transações (como agendamentos ou cancelamentos) possam ocorrer de maneira confiável. Além, claro, de notarmos que os dados que planejamos armazenar possuem forte relação entre si, algo que por si só já é uma característica idealmente ligada com o modelo relacional.

O monitoramento de integridade de sistemas já é algo comum e muito aplicado, consistindo em avaliações periódicas de status, disponibilidade e desempenho de nós ou serviços individuais, retornando ao consultante informações referentes aos resultados dessas verificações, afim de identificar falhas antes que se tornem problemas mais graves. Na nossa solução, poderíamos focar em verificações

simples no início, apenas nos certificando da disponibilidade dos servidores e sistemas necessários para determinado serviço.

Utilizaríamos também uma API RESTful - uma arquitetura cliente-servidor extremamente difundida e utilizada, que se conceitua como uma comunicação sem permanência de estado - como uma ponta de comunicação segura entre as diferentes necessidades do sistema com nosso banco de dados, oferecendo rotas que receberiam requisições HTTP simples, porém bem definidas, para realizar diferentes funções dentro da solução.

Como estratégia para contato com usuário final, poderíamos pensar em um sistema de notificação, baseado em uma arquitetura baseada em eventos. Dentro dessa estrutura, uma ação significativa no sistema (como a alteração de um status de disponibilidade), geraria um evento, que por sua vez acionaria o fluxo de trabalho automatizado que seria o responsável por identificar os serviços afetados e como efeito colateral os usuários que teriam seu agendamento comprometido, podendo lhes enviar notificações informando dessa indisponibilidade antes da chegada ao polo.

Também, para mantermos todas essas "áreas" no mesmo ambiente, podemos utilizar o Docker, uma tecnologia de conteinerização, que pode empacotar todas essas dependências em uma mesma unidade isolada chamada de contêiner, garantindo que tudo funcionará perfeitamente, de maneira isolada do resto do sistema e com uma portabilidade interessante, já que ao funcionar como uma tecnologia de virtualização, podemos "subir" o ambiente independente do sistema operacional utilizado.

Tendo revisado as funcionalidades e outras frentes relevantes, podemos agora destinar nossa atenção para a estrutura pretendida ao nosso banco de dados. Para isso, utilizaremos um diagrama ER com as tabelas relevantes em nossa base de dados.

Dentre essas tabelas, podemos citar as principais: Cidadão, com os dados referentes aos usuários do Poupatempo; Polo Atendimento, com os dados dos polos de atendimento presencial do Poupatempo; Sistema, com as informações dos sistemas utilizados para realização de serviços; Serviço, com a centralização das informações referentes aos serviços prestados; Horário Atendimento, para manter armazenado os diferentes horários de atendimento disponíveis conforme polo de atendimento; Agendamento, guardando os diferentes horários marcados pelos usuários para atendimento presencial;

Também, faz sentido a criação de tabelas auxiliares, ou com função de separar informações, ou Endereço, com os endereços de usuários ou polos; Orgão, com informações dos órgãos públicos atendidos nos Poupatempos; Categoria, com as categorias de serviços prestados; Período Exceção, para guardar os dias e horários que polos específicos (ou o Poupatempo em geral) estará indisponível, seja por manutenção, feriados ou qualquer outras razões; Status Sistema, para mantermos os diferentes estados possíveis em que um sistema pode estar (Online, Offline, Em Manutenção); Status Agendamento, para os estados de um agendamento (Confirmado, Cancelado por Usuário, Cancelado Pelo Sistema, Concluído ou Não Compareceu); Categoria Endereço, para as 2 opções de endereço (De Polo ou De Cidadão). Para melhor visualização dessas tabelas e dos dados que manteremos em cada uma, podemos visualizar abaixo o diagrama ER:

Figura 1 - Diagrama ER

Fonte: O autor (2025)

Após termos feito esse diagrama, partimos para o desenvolvimento prático desse banco de dados. Primeiro, para instanciar o mesmo, foi necessário pensar em uma maneira de isolar esse banco, inicializando o banco de maneira isolada, pensando justamente em um crescimento necessário da solução. Para isso, resolvemos utilizar o Docker, já citado anteriormente como uma tecnologia de conteinerização, podendo inicializar todas as outras dependências em uma mesma unidade isolada (contêiner).

Assim, criamos o docker-compose.yml, que irá definir a maneira que o contêiner será configurado; por desenvolvermos esse trabalho acerca do banco de dados, nos concentraremos nessa fração da solução proposta. O docker-compose pode ser consultado no apêndice

Α.

Um dos dados incluídos nesse docker-compose é uma referência ao script de criação do banco de dados, tabelas e etc. Ela está no

segundo item do bloco "volumes". Este script pode ser visualizado com mais detalhes no apêndice B.

Após a conclusão dessas etapas, apenas um simples comando ("docker-compose up -d") é capaz de executar e configurar o banco de dados em sua forma final, sendo facilmente manipulado e acessado para podermos verificar se a solução é realmente funcional, o que foi o caso.

Agora, após confirmar que a estrutura é funcional, podemos simular a utilização desse banco. Primeiro, pensaremos no processo inicial, por parte da equipe de suporte do sistema. Para disponibilizar o sistema aos cidadãos, é necessário cadastrar no sistema os diferentes polos e seus respectivos horários de atendimento e endereços, também os órgãos e categorias aos quais os serviços atenderão, bem como os serviços em si e os sistemas utilizados por eles.

A partir disso, seria necessário disponibilizar uma aplicação web ou mobile que disponibilize aos cidadãos a possibilidade de utilização da conta Gov (como já ocorre com os outros sistemas do Poupatempo ou outros sistemas governamentais) para realizar o agendamento. O procedimento de agendamento ocorreria assim: Primeiramente, o usuário poderia usar órgãos e/ou categorias para filtrar por serviços específicos. Após selecionar o serviço e polo de atendimento escolhido, o sistema utilizaria as regras de atendimento desse polo para retornar ao usuário o "calendário" para escolha do dia e posteriormente, através de um cálculo utilizando hora\_inicio, hora\_fim e duração\_slot\_minutos, o horário de atendimento (exemplo: 09:00, 09:30, 10:00...). Levando em consideração que essa lista de dias e horários disponíveis já passou por uma validação, que remove os períodos presentes na tabela "Periodo\_excecao", além de utilizar o número de agendamentos feitos para garantir que não ultrapasse o número de mesas disponíveis cadastradas no polo. O processo de automação (comentado futuramente) garantirá que em caso de o sistema não estar disponível no momento em que o agendamento é feito, um período de tempo já é declarado como indisponível, levando em consideração um tempo para recuperação e em caso de o usuário decidir marcar para após esse período ou para o dia seguinte, receberá um aviso prévio avisando a provável indisponibilidade do sistema.

Agora, podemos descrever como funcionaria o processo de monitoramento e automação de cancelamento: Um dos "tentáculos" da nossa solução seria responsável por verificar a saúde dos sistemas. Este processo, conhecido como Health Check, seria um serviço automatizado que, a intervalos regulares, enviaria requisições para os sistemas externos dos quais os serviços do Poupatempo dependem. Ao detectar uma falha - seja por um erro de resposta ou tempo de requisição esgotado - este serviço não agiria diretamente no banco de dados. Em vez disso, seguindo uma arquitetura de eventos, ele publicaria uma mensagem em um tópico de um message broker como o Apache Kafka. Essa mensagem, ou 'evento', conteria os dados essenciais, como o sistema\_id que apresentou falha e seu novo status\_sistema\_id. A partir daí, um segundo serviço, um 'consumidor', estaria inscrito neste tópico do Kafka, aguardando por novos eventos. Ao receber a mensagem de falha, sua primeira responsabilidade seria atualizar o status\_sistema\_id na tabela "Sistema" do nosso banco de dados. Imediatamente após, este mesmo serviço iniciaria o fluxo de cancelamento: ele consultaria a tabela "Agendamento" para encontrar todos os agendamentos futuros em um período de tempo pré-estabelecido pós checagem, ainda com status "Confirmado", que dependem dos serviços afetados pela queda do sistema.

Para cada agendamento encontrado, o status seria atualizado para "Cancelado pelo Sistema". A cada atualização bem-sucedida, um novo evento seria publicado em um segundo tópico do Kafka, dedicado exclusivamente às notificações. Um terceiro serviço, o 'notificador', consumiria essas mensagens e, utilizando os dados do cidadão e do agendamento, se encarregaria de formatar e enviar o aviso final por email, SMS ou notificação push. Essa arquitetura desacoplada garante que o sistema seja resiliente, escalável e que o usuário final seja notificado com eficiência, mitigando a frustração que deu origem a este estudo.

#### **4.REFLEXÕES FINAIS**

Com a conclusão dessa pesquisa, podemos afirmar que o objetivo inicial e principal, de idealizar um banco de dados completo para o

sistema do Poupatempo, foi plenamente alcançado, principalmente evidenciado através do script disponível no apêndice B. Acredito que a principal dificuldade encontrada foi a ausência de dados técnicos oficiais, o que exigiu a utilização de uma abordagem mais dedutiva, baseada em boas práticas do mercado. Posso dizer que o impacto dessa pesquisa em minha vida profissional foi ótimo, especialmente pela necessidade do pensamento crítico e criativo, pensando em todas as frentes que entrariam em conjunto com esse banco de da dados para a resolução total da problemática apresentada.

Ainda mais importante que meu desenvolvimento pessoal, é a comprovação que esse sistema é possível e aplicável, não sendo de um desenvolvimento tão extenso e complexo quanto pensado inicialmente. Logo, podemos concluir que o Poupatempo poderia sim se adaptar e resolver essa questão tão complicada e triste para com os cidadãos paulistas

# **SCAN SETTINGS**

These features were chosen to create this report

mitsettings	Private Cloud Hubs	
References:	Off	
Quotes:	Off	
Citations:	Off	
itles:	Off	
HTML Templates:	Off	
able Of Contents:	Off	
Code Comments:	Off	

#### **Plagiarism Detection Settings**

Security Measures	
Safe Search:	Off
Hide Sensitive Data:	Off
Character Manipulation:	Off
Similarity Level	
Identical:	On
Minor Changes:	On
Al Source Match:	Off

Results Calibration	
Focused Results	Sensitivity: 3