



Universidade de Brasília (UnB)

Faculdade de Economia, Administração, Contabilidade e Gestão de Políticas Públicas (FACE)

Programa de Pós-Graduação em Governança e Inovação em Políticas Públicas (PPGIPP)

PAULO TIBÉRIO MURADAS BULHÕES

Avaliando o Balcão Virtual do STJ como fonte de valor público

Brasília – DF

2025

PAULO TIBÉRIO MURADAS BULHÕES

Avaliando o Balcão Virtual do STJ como fonte de valor público

Trabalho de Conclusão de Curso apresentado ao Programa de Pós-Graduação em Governança e Inovação em Políticas Públicas (PPG-GIPP), da Faculdade de Economia, Administração, Contabilidade e Gestão de Políticas Públicas, como requisito parcial para obtenção do título de Mestre em Governança e Inovação em Políticas Públicas.

Orientador(a): Christiana Soares de Freitas

Brasília – DF

2025

PAULO TIBÉRIO MURADAS BULHÕES

Avaliando o Balcão Virtual do STJ como fonte de valor público

A Comissão Examinadora, abaixo identificada, aprova o Trabalho de Conclusão
do Curso de Gestão de Políticas Públicas da Universidade de Brasília do(a)
aluno(a)

PAULO TIBÉRIO MURADAS BULHÕES

Data da defesa: 25/03/2025

Comissão Examinadora:

Professora Doutora Christiana Soares de Freitas

Orientadora PPG-GIPP/UnB

Professora Doutora Elisabeth Aparecida Correa Menezes

Professora PPG-GIPP/UnB

Professor Doutor Montgomery Wellington Muniz

Examinador Externo/STJ

Ficha catalográfica elaborada automaticamente,
com os dados fornecidos pelo(a) autor(a)

BB933a Bulhões, Paulo Tibério
Avaliando o Balcão Virtual do STJ como fonte de valor público / Paulo Tibério Bulhões; orientador Christiana Soares de Freitas. Brasília, 2025.
138 p.

Dissertação(Doutorado em Administração) Universidade de Brasília, 2025.

1. Valor público. 2. Balcão Virtual. 3. Superior Tribunal de Justiça. 4. Videoconferência. 5. Abordagem da cadeia meio-fim. I. Soares de Freitas, Christiana, orient. II. Título.

RESUMO

Criação de valor público é a razão da existência de serviços públicos. Governos que falham em dar retorno adequado à sociedade (como contrapartida aos impostos pagos) têm dificuldade em se manter no poder. E, no Poder Judiciário, não é diferente — há uma grande pressão para se manter a produtividade, que é uma das mais altas do mundo¹, e justificar o gasto de 1,6% do PIB empregado no Judiciário². Diante desse contexto, os dados de atendimento do Balcão Virtual do Superior Tribunal de Justiça (STJ) são analisados a fim de se encontrar evidências de criação de valor público para o Tribunal e derivar informações que orientem a tomada de decisão baseada em evidências para aprimorar a geração de valor para o STJ. A abordagem desta pesquisa é a análise quantitativa dos registros de atendimento e das informações correlatas do Balcão Virtual. A finalidade é comparar com os componentes do modelo de qualidade de serviço o valor percebido e a intenção de uso continuado (Li; Shang, 2020) de acordo com as abordagens de uso continuado de sucesso de sistemas de informação (DeLone; McLean, 1992) e do modelo cadeia meio-fim (Gutman, 1982) (Zeithaml, 1988). Após a análise dos dados, encontraram-se indicações de que o Balcão Virtual gera valor público para o STJ pela intenção de uso continuado, evidenciado pela crescente demanda do serviço, aliado à avaliação positiva dos usuários. A comparação dos dados existentes com os elementos do modelo também serve como ferramenta de gestão, apontando lacunas nas informações de acompanhamento do serviço como oportunidades de melhoria. Os resultados dessa pesquisa sugerem a expansão de serviço de Balcão Virtual para os outros setores do Tribunal ainda não contemplados com essa modalidade de acesso.

Palavras-chave: Valor público; Balcão Virtual; Superior Tribunal de Justiça; abordagem de uso continuado; abordagem da cadeia meio-fim; videoconferência.

¹ Dado retirado do site do Conselho Nacional de Justiça (CNJ, 2024) e disponível em: <https://www.cnj.jus.br/justica-em-numeros-2024-barroso-destaca-aumento-de-95-em-novos-processos/>. Acesso em: 03 mar. 2025.

² Informação retirada do site Consultor Jurídico (Conjur, 2024a) e disponível em: <https://www.conjur.com.br/2024-fev-25/afinal-quanto-vale-o-poder-judiciario/>. Acesso em: 03 mar. 2025.

ABSTRACT

Creation of public value is the reason for the existence of public services. Governments that fail to provide adequate returns to society (as a counterpart to the taxes paid) struggle to stay in power. The same applies to the Judiciary — there is significant pressure to maintain productivity, which is among the highest in the world, and to justify the expenditure of 1.6% of GDP allocated to the Judiciary. In this context, data from the Virtual Desk of the Superior Court of Justice (STJ) is analyzed to find evidence of public value creation for the Court and to derive information that guides evidence-based decision-making to enhance value generation for the STJ. The research approach involves quantitative analysis of service records and correlated information from the Virtual Desk. The aim is to compare these with components of the service quality model, perceived value, and the intention for continued use by Li and Shang (Li; Shang, 2020), which is based on the information systems success model (DeLone; McLean, 1992) and the means-end chain model (Gutman, 1982) (Zeithaml, 1988).

Data analysis revealed indications that the Virtual Desk creates public value for the STJ through the intention for continued use, evidenced by the growing demand for the service, coupled with positive user evaluations. The comparison of existing data with model elements also serves as a management tool, highlighting gaps in service monitoring information as opportunities for improvement. The results of this research suggest expanding the Virtual Desk service to other sectors of the Court that are not yet included in this access modality.

Keywords: Public Value; Virtual Desk; Superior Tribunal de Justiça; information systems success model; means-end model; videoconferencing.

LISTA DE FIGURAS

Figura 1 – Triângulo estratégico de Moore (1997).....	22
Figura 2 – Modelo de uso continuado de serviços digitais Bhattacharjee (2001), após o uso inicial	24
Figura 3 – Modelo de cadeia meio-fim relacionando preço, qualidade e valor (Zeithaml, 1988).....	24
Figura 4 – Modelo de DeLone e McLean, revisado em 2003	25
Figura 5 – Representação do modelo Li e Shang (2020) e dos modelos dos quais foi derivado	26
Figura 6 – Modelo representando as variáveis e os atributos com as evidências relativas ao Balcão Virtual	42
Figura 7 – Modelo de Li e Shang (2020).....	62

LISTA DE GRÁFICOS

Gráfico 1 – Histograma do tempo de atendimento do Balcão Virtual.....	36
Gráfico 2 – Demanda do Balcão Virtual 2021-2023	41
Gráfico 3 – Tipos de atendimento do Balcão Virtual	44
Gráfico 4 – Atendimento diários de "Falar no Gabinete"	45
Gráfico 5 – Demanda por Gabinete (anonimizado).....	45
Gráfico 6 – Demanda por seção.....	46
Gráfico 7 – Tipos de processo	46
Gráfico 8 – Atendimento diários (Balcão Virtual vs Telefone).....	47
Gráfico 9 – Atendimento diários (telefone)	48
Gráfico 10 – Histograma do tempo de atendimento em segundos	49
Gráfico 11 – Tipos de atendimento do Balcão Virtual.....	49
Gráfico 12 – Tipos de atendimento (telefone).....	50
Gráfico 13 – Tipos de atendimento do Balcão Virtual	65
Gráfico 14 – Atendimento diários de "Falar no Gabinete"	66
Gráfico 15 – Demanda por Gabinete (anonimizado).....	66
Gráfico 16 – Demanda por seção.....	67
Gráfico 17 – Tipos de processo	67
Gráfico 18 – Tempo médio entre atendimentos por Gabinete (anonimizados).....	69
Gráfico 19 – Tempo de atendimento nos Gabinetes.....	70
Gráfico 20 – Atendimento diários (Balcão Virtual vs Telefone).....	74
Gráfico 21 – Atendimento diários (telefone)	75
Gráfico 22 – Histograma do tempo de atendimento em segundos	76
Gráfico 23 – Tipos de atendimento do Balcão Virtual	76
Gráfico 24 – Tipos de atendimento (telefone).....	77
Gráfico 25 – Contagem diária de atendimentos do Balcão Virtual	79
Gráfico 26 – Contagem diária de atendimentos por e-mail.....	80
Gráfico 27 – Contagem diária de atendimentos por telefone	80
Gráfico 28 – Contagem diária de atendimentos do Espaço do Advogado	80

LISTA DE QUADROS

Quadro 1 – Atributos e variáveis do modelo Li e Shang (2020).....	26
Quadro 2 – Quadro-resumo da presença/ausência dos atributos de qualidade do serviço.....	33
Quadro 3 – Quadro-resumo da presença/ausência das variáveis dependentes.....	39
Quadro 4 – Quadro-resumo da presença/ausência das variáveis	42
Quadro 5 – Questões do modelo Li e Shang (2020)	92

LISTA DE TABELAS

Tabela 1 – Ferramenta de videochamada (Zoom)	32
Tabela 2 – Qualidade das informações prestadas	33
Tabela 3 – Duração do atendimento	34
Tabela 4 – Tempo médio de atendimento do Balcão Virtual	34
Tabela 5 – Perfis de acesso ao Balcão Virtual	36
Tabela 6 – Dispositivos de acesso ao Balcão Virtual	37
Tabela 7 – Atendimento de pessoas com deficiência	38
Tabela 8 – Facilidade de acesso ao Balcão Virtual	38
Tabela 9 – Satisfação com o Balcão Virtual	39
Tabela 10 – Recomendação do Balcão Virtual para outros usuários	40
Tabela 11 – Volume de chamados do Balcão Virtual	41
Tabela 12 – Número de chamados por usuário	42
Tabela 13 – atendimentos diários (Balcão Virtual vs Telefone)	47
Tabela 14 – Estatísticas de tempo de atendimento telefônico	48
Tabela 15 – Tribunais de origem	68
Tabela 16 – Perguntas do questionário da Ouvidoria	73
Tabela 17 – atendimentos diários (Balcão Virtual vs Telefone)	74
Tabela 18 – Estatísticas de tempo de atendimento telefônico	75
Tabela 19 – Extrato anonimizado com diferentes identificações para um mesmo usuário	78
Tabela 20 – Relação dos tipos de atendimento e assuntos	96
Tabela 21 – Exemplo de problema de identificação única de um usuário	98
Tabela 22 – Lista detalhada dos 40 tribunais que mais figuram nos processos referenciados no atendimento do Balcão Virtual	99
Tabela 23 – Exemplo de registro de atendimento anonimizado	101

SUMÁRIO

1 INTRODUÇÃO	14
2 DESCRIÇÃO DA PESQUISA TEÓRICO-EMPÍRICA	20
2.1 Introdução	20
2.2 Referencial teórico	21
2.2.1 <i>Valor público</i>	21
2.2.2 <i>Modelo de avaliação de valor público de serviços digitais</i>	23
2.3 Métodos e técnicas	27
2.3.1 <i>Abordagem de modelo como mapa</i>	27
2.3.2 <i>Tratamento de dados</i>	28
2.4 Resultados e discussões.....	30
2.4.1 <i>Fontes de informação</i>	30
2.4.2 <i>Análise das variáveis do modelo: qualidade do serviço</i>	30
2.4.3 <i>Análise das variáveis do modelo: valores</i>	34
2.4.4 <i>Análise das variáveis do modelo: variáveis dependentes</i>	39
2.4.5 <i>Objetivo geral.....</i>	43
2.4.6 <i>Objetivo específico: demanda por expansão do Balcão Virtual para os Gabinetes de Ministros.....</i>	43
2.4.7 <i>Objetivo específico: comparação com o serviço de atendimento telefônico</i>	47
2.4.8 <i>Objetivo específico: recomendações de melhoria.....</i>	50
2.4.9 <i>Resposta à questão de pesquisa</i>	52
2.5 Conclusões e recomendações.....	52
3 PRODUTO TÉCNICO-TECNOLÓGICO: PROJETO DE ASSESSORIA PARA OS GESTORES DO BALCÃO VIRTUAL NO STJ.....	54
3.1 Introdução	54
3.2 Descrição geral do produto	54
3.3 Base teórica utilizada.....	56
3.4 Relevância do produto.....	56

3.5 Relatório técnico conclusivo de Assessoria	57
3.5.1 <i>Introdução</i>	61
3.5.2 <i>Recomendações</i>	63
3.5.2.1 Recomendação 1: expandir o Balcão Virtual para os Gabinetes	63
3.5.2.1.1 Justificativa.....	63
3.5.2.1.2 Estratégia de adoção	63
3.5.2.1.3 Evidência de suporte à recomendação.....	64
3.5.2.2 Recomendação 2: incluir questão de percepção de valor público do Balcão Virtual no questionário de controle de qualidade da Ouvidoria	71
3.5.2.2.1 Justificativa.....	71
3.5.2.2.2 Estratégia de adoção	71
3.5.2.2.3 Evidência de suporte à recomendação.....	71
3.5.2.3 Recomendação 3: incluir questão sobre percepção de segurança da informação no questionário de controle de qualidade da Ouvidoria	71
3.5.2.3.1 Justificativa.....	71
3.5.2.3.2 Estratégia de adoção	72
3.5.2.3.3 Evidência de suporte à recomendação.....	72
3.5.2.4 Recomendação 4: aplicar questionário de controle de qualidade para outros canais	72
3.5.2.4.1 Justificativa.....	72
3.5.2.4.2 Estratégia de adoção	72
3.5.2.4.3 Evidência de suporte à recomendação.....	73
3.5.2.5 Recomendação 5: identificar os usuários de maneira uniforme.....	77
3.5.2.5.1 Justificativa.....	77
3.5.2.5.2 Estratégia de adoção	77
3.5.2.5.3 Evidência de suporte à recomendação.....	78
3.5.2.6 Recomendação 6: melhorar o registro dos atendimentos nos canais de e-mail, telefone e Espaço do Advogado.....	78

3.5.2.6.1 Justificativa.....	78
3.5.2.6.2 Estratégia de adoção	79
3.5.2.6.3 Evidência de suporte à recomendação.....	79
3.5.2.7 Recomendação 7: mensurar o tempo de espera por atendimento.....	81
3.5.2.7.1 Justificativa.....	81
3.5.2.7.2 Estratégia de adoção	81
3.5.2.7.3 Evidência de suporte à recomendação.....	81
3.5.2.8 Recomendação 8: registrar os tempos de indisponibilidade do serviço	81
3.5.2.8.1 Justificativa.....	81
3.5.2.8.2 Estratégia de adoção	81
3.5.2.8.3 Evidência de suporte à recomendação.....	82
3.5.2.9 Recomendação 9: melhorar a validação dos registros de atendimentos do Balcão Virtual	82
3.5.2.9.1 Justificativa.....	82
3.5.2.9.2 Estratégia de adoção	82
3.5.2.9.3 Evidência de suporte à recomendação.....	82
3.5.2.10 Recomendação 10: adaptar e aplicar o questionário do modelo de Li e Shang (2020) nos canais de atendimento do STJ	83
3.5.2.10.1 Justificativa.....	83
3.5.2.10.2 Estratégia de adoção	83
3.5.2.10.3 Evidência de suporte à recomendação.....	83
3.5.3 CONCLUSÃO.....	84
3.5.4 REFERÊNCIAS.....	85
REFERÊNCIAS	86
ANEXO A – QUESTÕES DO MODELO DE LI E SHANG (2020)	92
APÊNDICE A – QUESTIONÁRIO DA OUVIDORIA	94
APÊNDICE B – EXEMPLO DE DADOS DE CONSULTA PROCESSUAL	95
APÊNDICE C – TIPOS DE ATENDIMENTO.....	96
APÊNDICE D – EXEMPLO DE PROBLEMA DE IDENTIFICAÇÃO DE USUÁRIO	98

APÊNDICE E – TRIBUNAIS DE ORIGEM.....	99
APÊNDICE F – EXEMPLO DE REGISTRO DE ATENDIMENTO ANONIMIZADO	101
APÊNDICE G – SOFTWARE DE TRATAMENTO E ANÁLISE DE DADOS	103

1 INTRODUÇÃO

O valor público é definido como as expectativas coletivas dos cidadãos em relação ao Governo e aos serviços públicos (Moore, 1997). De modo geral, um serviço público mais ágil e menos custoso está correlacionado a maior valor público. Isso significa que o aumento de valor público é imperativo ao Poder Público, e a falta dele (como serviços ruins ou gastos maiores que as entregas) causa efeitos políticos e sociais sérios e afeta diretamente o desenvolvimento econômico e social das nações.

É importante observar que a criação de valor público no contexto do Judiciário não é sem controvérsias. A atividade de solução de conflitos da sociedade e a garantia de direitos dos cidadãos³ implicam que uma ou mais partes em conflito tenham suas demandas negadas em um processo judicial. Ou seja, o resultado de uma resolução de conflito pode gerar valor para uma parte e diminuir para outra. Mesmo a celeridade processual, art. 5º LXXVII (Brasil, 1988), é vista de maneira relativa pelas partes, como observado por Sadek: “Muitos empresários admitiram que a morosidade é, por vezes, benéfica, principalmente na área trabalhista” (Sadek, 2004, p. 86):

Dito isso, o foco deste artigo é avaliar se um dos canais de atendimento cria valor público para o Superior Tribunal de Justiça (STJ), que não sofre dessa controvérsia, porque é relacionado ao atendimento dos jurisdicionados, e não à resolução de conflitos *per se*.

Segundo Kearns, o valor público tem três fontes (Kearns, 2004):

1. Entrega de serviços de alta qualidade: a percepção desses serviços é relacionada a fatores como disponibilidade, satisfação dos usuários, importância percebida desses serviços e, por fim, custo;
2. Resultados (*outcomes*): são vistos como desejados pela população, como melhoria na saúde, redução de pobreza ou preservação do meio ambiente; e
3. Confiança (imagem) nas instituições públicas: é uma importante fonte de valor, fazendo os cidadãos mais propensos a aceitar ações governamentais e se sentirem associados ao Governo.

Essas fontes estão diretamente ligadas aos seguintes objetivos estratégicos do Plano Estratégico do Superior Tribunal de Justiça (2021-2026): (I) “fortalecer a imagem e a transparência institucional” e (II) “intensificar o uso de tecnologias da informação” (Superior Tribunal de Justiça, 2021, p. 19). O objetivo aqui delimitado (I) é definido como: “Atuar com foco no cidadão, em parceria com outras instituições, oferecendo *serviços de qualidade*,

³ As atribuições do Supremo Tribunal de Justiça podem ser consultadas neste link, disponível em: <https://www.stj.jus.br/sites/portalp/Institucional/Atribuicoes>. Acesso em: 03 de mar. 2025.

comunicação efetiva e *resultados* transparentes, de modo a *fortalecer a imagem*⁴ do STJ perante a sociedade” (Superior Tribunal de Justiça, 2021b, p. 21) (Os termos grifados correspondem aos pilares do valor público e são prioridade para as gestões do STJ no período 2021–2026). Já o objetivo estratégico institucional (II) está relacionado ao valor público com o foco em tecnologia e é definido como: “Aperfeiçoar os serviços e a infraestrutura de tecnologia da informação e comunicação – TIC, bem como a segurança das informações, incentivando a *transformação digital*, com foco na *satisfação dos usuários*⁵ e no cumprimento da missão institucional” (Superior Tribunal de Justiça, 2021, p. 23).

O foco em transformação digital dos serviços não é por acaso; o Governo brasileiro tem avançado nisso por meio do processo de evolução de e-Governo para Governo Digital (Souza de, Nunes, 2021; OECD, 2018; Figueiras, Palotti, Cireno, 2018). Essa transformação, iniciada na primeira década do século XXI, tem progredido bem, atingindo uma boa colocação (*Very High EGDI*) no *ranking* de EGDI das Nações Unidas (UN Department of Economic and Social Affairs, 2022), especialmente no componente de *Online Services Index* (OSI), com valor de 0,8964 — um dos mais altos das Américas — relacionado à prestação de serviços digitais.

No âmbito do Judiciário, a principal transformação digital foi a migração do processo físico para o processo eletrônico, também iniciada na primeira década do século XXI (Brasil, 2006)⁶, e está quase concluída⁷ com apenas três de 92 tribunais com acervos significativos de processos físicos no fim de 2021. A previsão é de que todos os acervos de processos físicos estejam digitalizados até o fim de 2025.

Em 2024, as principais ações de transformação digital no Judiciário foram agrupadas no programa Justiça 4.0⁸ do Conselho Nacional de Justiça (CNJ), o que inclui o Balcão Virtual. A adoção do Balcão Virtual foi determinada pela Resolução CNJ n.º 372, de 12 de fevereiro de março de 2021, e esse serviço foi implementado com urgência no Judiciário brasileiro devido à pandemia de covid-19 como uma maneira de estimular o distanciamento social e combater o coronavírus. Salienta-se que o Balcão Virtual do STJ foi meticulosamente implantado a fim de

⁴ Grifos do autor.

⁵ Grifos do autor.

⁶ A Lei nº 11.419, de 19 de dezembro de 2006, foi a responsável por dispor sobre a informatização do processo judicial. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2006/lei/11419.htm. Acesso em: 03 mar. 2025.

⁷ Matéria retirada do site do CNJ, de 22 de setembro de 2021, informando que o Judiciário vai receber apenas processos eletrônicos a partir de março de 2022. Disponível em: <https://www.cnj.jus.br/judiciario-vai-receber-apenas-processos-eletronicos-a-partir-de-marco-de-2022/>. Acesso em: 03 mar. 2025.

⁸ O Programa Justiça 4.0 moderniza o sistema judiciário brasileiro com novas tecnologias e inteligência artificial, tornando-o mais ágil, eficaz e acessível. Automatiza atividades dos tribunais, aumentando a produtividade e transparência, além de otimizar o trabalho de magistrados, servidores e advogados. Informação disponível em: <https://www.cnj.jus.br/tecnologia-da-informacao-e-comunicacao/justica-4-0/>. Acesso em: 03 mar. 2025.

ser um canal de atendimento de excelência, em harmonia com a imagem da Corte, e não somente para atender a uma resolução do CNJ. No STJ, além do Balcão Virtual, o atendimento ao público dá-se em modo presencial (Espaço do Advogado⁹), por e-mail e telefone¹⁰.

Sendo uma inovação, o Balcão Virtual é um novo olhar sobre o atendimento da Corte, implementado como um serviço que permite uma nova relação entre o Tribunal e os jurisdicionados. Portanto, este estudo está alinhado aos objetivos da linha de pesquisa de “Ação pública e inovação” do Programa de Pós-Graduação em Governança e Inovação em Políticas Públicas da Universidade de Brasília (UnB).

O Balcão Virtual foi escolhido como objeto desta pesquisa por ser uma inovação recente nos serviços do Tribunal ligada à transformação digital do serviço público. Além disso, a geração de valor a ele associada serve para avaliar se o investimento nessa inovação foi justificado, se esse novo serviço tem auxiliado a população e quais lições podem ser extraídas para amparar a tomada de decisão na gestão do Balcão Virtual. Logo, os achados deste estudo podem trazer benefícios e servir de base de comparação a outras implementações do Balcão Virtual no Judiciário brasileiro. Outro fator importante foi a disponibilidade dos registros de cada chamado ao Balcão. Por fim, há um interesse dos gestores do Balcão Virtual em expandir o serviço para mais áreas do Tribunal e acredita-se que este estudo vai prover informações e recomendações úteis para esse fim.

Valor — a diferença entre o preço de um serviço, em comparação com o benefício percebido (Chang; Dibb, 2012), mesmo no contexto da iniciativa privada — é, muitas vezes, subjetivo e situacional. Os consumidores de um serviço percebem um benefício de, no mínimo, o preço do serviço quando efetuam a sua compra. Entretanto, os benefícios percebidos (reais ou imaginários) variam: alguns percebem a qualidade como sendo melhor que os concorrentes; alguns, por razões regulatórias (*e.g.* revisão de carros novos devem ocorrer em concessionárias autorizadas pelo fabricante) ou mesmo pelo símbolo de *status*. Além disso, o preço do serviço representa um tempo de trabalho maior ou menor para obter o dinheiro necessário para adquiri-lo, dependendo da renda de um consumidor. Para um consumidor que recebe salário-mínimo,

⁹ É um espaço de atendimento do STJ que reúne serviços essenciais para operadores do direito, oferecendo conforto e praticidade. Disponibiliza estações de trabalho, sala de reunião, videoconferência, impressora e café. Conta com equipe treinada para esclarecer assuntos processuais e facilitar a rotina dos advogados. (Superior Tribunal Federal, 2021a). Disponível em: <https://cartadeservicos.stj.jus.br/atendimento-judicial/atendimento-judicial-presencial-espaco-do-advogado/>. Acesso em: 03 mar. 2025.

¹⁰ Notícia retirada do site do STJ, de 25 de março de 2021, e disponível em: <https://www.stj.jus.br/sites/porta/p/Paginas/Comunicacao/Noticias/25032021-Balcao-Virtual-atende-advogados-e-publico-em-geral-por-videoconferencia.aspx>. Acesso em: 03 mar. 2025.

o preço do serviço corresponde a muito mais horas de trabalho do que de um consumidor de alta renda.

Por mais que a quantificação de valor de um serviço seja um tema em estudo (Zauner, Koller, Hatak, Walla, 2015), é relativamente fácil identificar o valor de um serviço — a demanda por um serviço oferecido em um mercado competitivo e se os clientes estão usando o serviço mais de uma vez indicam que os consumidores percebem benefícios superiores ao preço do serviço, o que significa afirmar que esse serviço tem valor.

Em serviços oferecidos pelo Governo, é mais difícil mensurar o valor público percebido, visto que o preço de um dado serviço é socializado: o usuário do serviço público, em geral, não paga pelos serviços diretamente. Na percepção do usuário, o serviço público é “grátis”. A título de exemplificação, pode-se imaginar o seguinte cenário: o orçamento do STJ aprovado na LOA 2024 foi de R\$ 2.104.718.497 (Superior Tribunal Federal, 2024a)¹¹. Contando que o Brasil tenha 203.080.756 habitantes¹², o STJ “custa” aproximadamente R\$ 10,36 por habitante anualmente — um valor irrisório para seus usuários, que, por vezes, têm ações bilionárias sendo julgadas (Superior Tribunal Federal, 2023b)¹³. Além disso, a demanda por bens e serviços em um mercado competitivo, usada para identificar a criação de valor na iniciativa privada, por si só, não se aplica a serviços públicos. Isso porque esses serviços, muitas vezes, são monopólios estatais, obrigando os cidadãos a utilizá-los, ainda que sejam de baixa qualidade e que o benefício percebido seja negativo.

No caso do STJ, além do Balcão Virtual, há outros canais de atendimento¹⁴: telefônico, e-mail e presencial (Espaço do Advogado). Assim sendo, os usuários não ficam “presos” em um único canal, pois há liberdade de escolha. Isso facilita a identificação de criação de valor público, já que a demanda de um canal tem concorrência de outros canais. Caso o Balcão Virtual não seja satisfatório, o usuário pode optar por outro canal de atendimento. Esse aspecto de concorrência entre os canais como fator de avaliação de criação de valor público de um serviço é explorado de maneira limitada neste estudo porque apenas existem dados parciais do atendimento por telefone para se comparar com os dados do Balcão Virtual.

¹¹ Dado disponível em: https://transparencia.stj.jus.br/wp-content/uploads/2024_Lei-Orcamentaria-LOA-.pdf. Acesso em: 17 nov. 2024.

¹² Dado disponível em: <https://censo2022.ibge.gov.br/panorama/>. Acesso em: 26 fev. 2024.

¹³ Notícia retirada do site do STJ, de 16 de março de 2023, e disponível em: <https://www.stj.jus.br/sites/portalp/Paginas/Comunicacao/Noticias/2023/16032023-Corte-rejeita-recorso-da-J-F-e-mantem-tramitacao-de-acao-que-discute-multa-bilionaria-em-acordo-de-leniencia.aspx>. Acesso em: 26 fev. 2024.

¹⁴ Lista dos canais de atendimento aos jurisdicionados do STJ disponível em: <https://www.stj.jus.br/sites/portalp/Contato-e-ajuda/Fale-conosco/Atendimento-judicial>. Acesso em: 29 dez. 2024.

Por essas razões, a pesquisa foca na identificação de geração de valor do Balcão Virtual, e não na sua quantificação. Assim, a questão central de pesquisa é: “É possível identificar a criação de valor público para o STJ pelo Balcão Virtual através de uma combinação de registros de atendimento, pesquisas de satisfação e dados públicos?”. A partir dessa questão, deriva-se o objetivo geral, que é identificar se o Balcão Virtual do STJ cria valor público para a Corte com base nos dados existentes (registros de atendimento, questionário de controle de qualidade do Balcão Virtual e indicadores). Partindo dos achados, da análise feita para atingir o objetivo geral e das sugestões dos gestores do Balcão Virtual, os objetivos específicos são: buscar evidências de demanda de expansão do Balcão Virtual para os Gabinetes de Ministros; comparar, se possível, o atendimento do Balcão Virtual com o telefônico; e sugerir melhorias aos gestores do Balcão Virtual e de outros canais de atendimento, a fim de se aumentar o valor público do STJ.

O Capítulo 2 apresenta as bases teóricas para identificação da criação de valor público, a seleção e a aplicação do modelo de avaliação adotado, a metodologia de tratamento e a análise dos dados. Por fim, são discutidos os resultados e as evidências de valor público gerado pelo Balcão Virtual e a conclusão.

No Capítulo 3, é apresentado o Produto Técnico-Tecnológico, que consiste em um projeto de assessoria para os gestores do Balcão Virtual no STJ, com recomendações acionáveis baseadas na análise dos dados, visando aprimorar o serviço e ampliar seu valor público. O relatório técnico conclusivo está apresentado na Seção 3.5. São indicadas melhorias nos métodos de registro de informação e no questionário de controle de qualidade para corrigir lacunas de informação e sugere-se a expansão do Balcão Virtual para os Gabinetes de Ministros.

A abordagem apresentada neste trabalho pode ser aplicada em outras áreas do serviço público, desde que haja informações suficientes para derivar as variáveis do modelo adotado: qualidade do serviço digital, registro de atendimentos e pesquisas de satisfação. O uso de um modelo como guia também auxilia na identificação e no preenchimento de lacunas informacionais. Por exemplo, se o modelo tem um componente de segurança e não há informações disponíveis, pode-se consultar o questionário usado na elaboração do modelo para saber o que perguntar aos usuários e obter informações sobre segurança.

No contexto do Poder Judiciário, essa abordagem permite uma comparação entre os serviços de Balcão Virtual implantados nos diversos tribunais do país, possibilitando a análise de tempos de resposta, volume de atendimento e satisfação dos usuários, orientando a gestão dos serviços digitais.

As sugestões apresentadas no Produto Técnico-Tecnológico podem trazer impactos positivos significativos para o atendimento no STJ, tanto na qualidade das informações disponíveis para tomada de decisão quanto na ampliação dos serviços disponíveis aos usuários.

2 DESCRIÇÃO DA PESQUISA TEÓRICO-EMPÍRICA

2.1 Introdução

A Receita Federal arrecadou 2,318 trilhões de reais em 2023 (Ministério da Fazenda, 2024), o que representa, aproximadamente, 11.414,18 reais por brasileiro (considerando a renda média mensal de R\$ 1.848,00 (Secretaria de Comunicação Social, 2024). Esse cenário evidencia a elevada carga tributária e a necessidade de um retorno social proporcional. Caso contrário, a União consumiria mais recursos do que devolve, comprometendo o desenvolvimento nacional e os objetivos do art. 3º da Constituição Brasileira de 1988:

- I - construir uma sociedade livre, justa e solidária;
- II - garantir o desenvolvimento nacional;
- III - erradicar a pobreza e a marginalização e reduzir as desigualdades sociais e regionais;
- IV - promover o bem de todos, sem preconceitos de origem, raça, sexo, cor, idade e quaisquer outras formas de discriminação (Brasil, 1988).

A criação de valor público, ou seja, um retorno social adequado aos recursos empregados, é de fundamental importância para o Estado, e o STJ também tem essa responsabilidade.

Segundo Kearns (2004), valor público decorre da entrega de serviços de qualidade, de resultados desejados pela sociedade e da confiança institucional. Esses elementos se alinham aos objetivos estratégicos do STJ (2021–2026), que visam fortalecer a transparência e intensificar o uso de tecnologia (Superior Tribunal Federal, 2021). O Plano Estratégico do STJ busca atender ao cidadão com serviços eficientes e comunicação transparente, fortalecendo sua imagem institucional. A transformação digital, impulsionada pelo Conselho Nacional de Justiça no programa Justiça 4.0, inclui o Balcão Virtual, regulamentado pela Resolução CNJ n.º 372/2021, criado durante a pandemia de covid-19 para viabilizar o atendimento remoto. Sendo o STJ um Tribunal de abrangência nacional, com a missão de uniformizar a interpretação da legislação federal e promover uma Justiça célere e cidadã, a habilidade de atender aos cidadãos onde quer que estejam, utilizando os recursos disponíveis, é crucial para facilitar o acesso à Justiça. O Balcão Virtual cumpre esse papel e complementa os atendimentos presenciais, por e-mail e telefone. É válido ressaltar que os dados da pesquisa de satisfação da Ouvidoria confirmam que o serviço atendeu usuários em todos os estados, no Distrito Federal e até fora do país, evidenciando seu alcance supranacional.

O Balcão Virtual é um serviço de atendimento aos jurisdicionados do STJ baseado em videoconferência. Para utilizá-lo, os usuários devem acessar o link <https://balcaovirtual.stj.jus.br/>, clicar no botão “Acessar” e preencher a página de identificação com as informações de contato do usuário, tipo de demanda, CPF ou OAB, número de processo

(se houver) e tipo de dispositivo utilizado (computador, celular ou tablet). Após essa etapa, o usuário é encaminhado a uma sala de acolhimento e, em poucos minutos, é conectado por videoconferência a um dos consultores da Coordenadoria de Atendimento e Protocolo Judicial (CATJ). O consultor realiza o atendimento inicial e, se necessário, aciona as áreas técnicas do STJ para solução de demandas específicas. Durante o atendimento, as interações são registradas no sistema de informações do Balcão Virtual, constituindo uma das principais fontes de dados deste estudo. Uma vez encerrado o atendimento, sugere-se que o usuário responda à pesquisa de satisfação da Ouvidoria do STJ, que também integra o conjunto de dados analisado nesta pesquisa.

Nesse contexto, este estudo avalia o Balcão Virtual do STJ como ferramenta de criação de valor público, propondo melhorias com base em evidências. A questão de pesquisa investiga se é possível identificar esse valor por meio de registros de atendimento, pesquisas de satisfação e dados públicos. O objetivo geral é verificar se o Balcão Virtual gera valor público para o STJ, enquanto o objetivo específico visa sugerir aprimoramentos nos serviços de atendimento para ampliar esse impacto.

2.2 Referencial teórico

2.2.1 Valor público

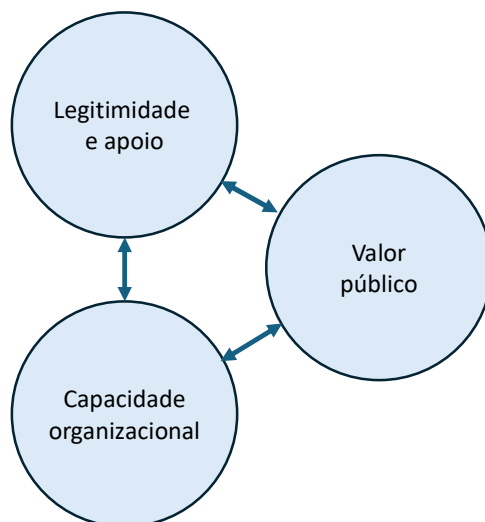
Para abordar o problema de identificação de geração de valor público por serviços digitais governamentais, este estudo parte dos textos seminais de valor público de Moore (1997). O autor adapta o conceito de *shareholder value* (valor para os acionistas) para o setor público, defendendo que gestores públicos devem tomar decisões visando o benefício da sociedade como um todo, indo além da missão específica de suas organizações.

Moore (1997) exemplifica esse conceito com o caso de uma gerente de biblioteca que pode negar ou aprovar o uso do espaço por crianças cujos pais trabalham (*latchkey children*¹⁵). No sentido mais burocrático e estrito da missão da biblioteca, a resposta seria “não”, com o argumento de que biblioteca não é creche. Porém, na abordagem de valor público, o uso da biblioteca pelas crianças pode ser aceito porque gera valor para a sociedade em geral e pode contribuir para a consolidação do hábito de leitura nas crianças. Entretanto, a gerente da biblioteca teria que ter certos cuidados: é necessário assegurar de que a biblioteca teria capacidade de acolher as crianças e legitimar esse escopo maior de atuação com a sociedade,

¹⁵ “Uma criança que tem a chave de sua casa e muitas vezes fica sozinha em casa depois que a escola termina porque seus pais estão trabalhando”. Trecho traduzido de Dictionary Cambridge e disponível em: <https://dictionary.cambridge.org/dictionary/english/latchkey-kid>. Acesso em: 3 mar. 2025.

visto que o espaço da biblioteca é público. Moore (1997) sintetiza essa abordagem no triângulo estratégico, que equilibra três pilares: valor público, legitimidade e capacidade organizacional (Figura 1).

Figura 1 – Triângulo estratégico de Moore (1997)



Fonte: Elaborado pelo autor segundo Moore (1997, p. 70).

No STJ, essa visão de escopo maior de atuação em prol da sociedade já foi empregada com sucesso no acordo de desjudicialização do STJ com a Advocacia-Geral da União (AGU). Essa iniciativa, na qual a AGU deixa de recorrer em casos com pouca chance de sucesso, vai além da missão do STJ de “Uniformizar a interpretação da legislação federal e oferecer justiça ágil e cidadã” (Superior Tribunal Federal, s.d, n.p)¹⁶ e gerou grandes benefícios para a sociedade, com a prevenção de mais de 2 milhões de processos “não só no âmbito do STJ, mas também dos tribunais de origem e dos juízos de primeiro grau” (Superior Tribunal Federal, 2023a)¹⁷.

Moore (1997) também discute conceitos mais operacionais de reconhecimento de valor público, sendo eles: ‘atingindo objetivos obrigatórios eficientemente e eficazmente’, ‘competência politicamente neutra’, ‘técnicas analíticas para averiguar valor público’ e, mais recentemente, ‘foco no atendimento ao cliente e satisfação dos usuários’. Este último conceito é o mais próximo deste estudo, pois o Balcão Virtual é avaliado como criador de valor público por qualidade no atendimento e satisfação dos usuários.

¹⁶ Informação retirada do site do STJ na Seção Missão, Visão e Valores. Disponível em: <https://transparencia.stj.jus.br/gestao/objetivos-estrategicos/missao-visao-valores-do-stj/>. Acesso em: 9 fev. 2025.

¹⁷ Dado retirado de notícia do site do STJ “Acordo com AGU intensifica desjudicialização e alcança mais de dois milhões de processos, de 18 de agosto de 2023. Disponível em: <https://www.stj.jus.br/sites/porta1p/Paginas/Comunicacao/Noticias/2023/17082023-Acordo-com-AGU-intensifica-desjudicializacao.aspx>. Acesso em: 9 fev. 2025.

Kearns (2004) aplica o conceito de valor público para o Governo eletrônico e define três fontes principais:

1. Entrega de serviços de alta qualidade, associada à disponibilidade, satisfação dos usuários, importância percebida desses serviços e custo.
2. Resultados (*outcomes*) desejados pela população, como melhorias na saúde, redução de pobreza ou preservação do meio ambiente.
3. Confiança (imagem) nas instituições públicas, essencial para a aceitação das ações governamentais.

Este estudo adota essas fontes de Kearns (2004) supracitadas como ponto de partida para avaliação de criação de valor público do Balcão Virtual. O foco está na entrega de serviços de qualidade, porque o Balcão Virtual é um serviço digital, e nos resultados (*outcomes*), porque se espera poder identificar resultados positivos ou negativos do Balcão Virtual nos dados existentes. Já a confiança e a imagem, por abrangerem o STJ como um todo, estão fora do escopo da pesquisa.

2.2.2 Modelo de avaliação de valor público de serviços digitais

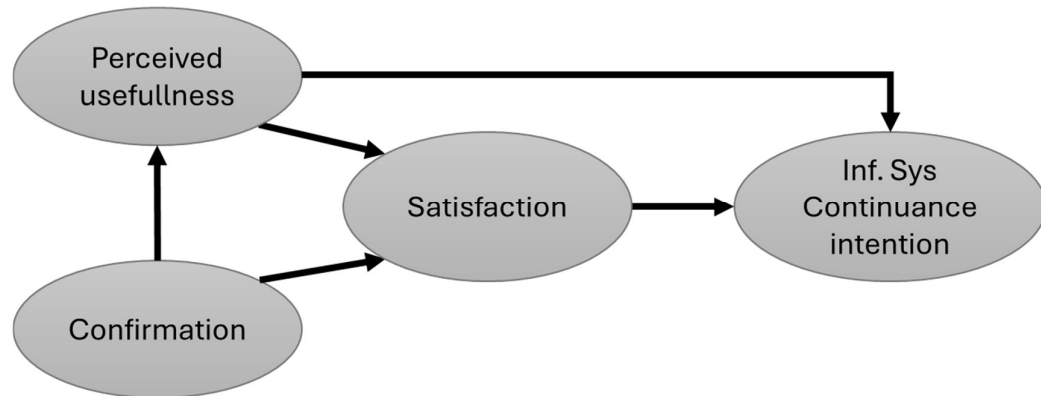
Este trabalho é um estudo explicativo que procura identificar as variáveis que contribuem na criação de valor do Balcão Virtual, tendo como ponto de partida um modelo de avaliação de criação de valor de serviços digitais existentes. A busca pelo modelo de avaliação iniciou-se com Twizeyimana e Andersson (2019), que expandiram o *framework* de valor público de Ndou (2004) e incluíram indicadores de desempenho relacionados a valor público (*Key Performance Indicators – KPIs*). Também foi investigado o *framework* de indicadores de valor público de estudo recente na China (Mensah, Zeng, Mwakapesa, 2022) baseado no *framework* de Twizeyimana e Andersson (2019). No entanto, esses *frameworks* são mais adequados à avaliação global dos serviços digitais de um Governo, e o foco deste trabalho é a avaliação de um único serviço dentro de uma organização governamental.

Prosseguindo na avaliação de modelos existentes, chegou-se ao modelo de Li e Shang (2020), que se baseia nos modelos de expectativa-confirmação em sistemas de informação (Bhattacharjee, 2001), no modelo de comportamento dos usuários de Zeithaml (1988) e em uma contribuição do modelo de sucesso de sistemas de informação de DeLone e McLean (1992). O modelo de Li e Shang (Anexo A) foi considerado o mais apropriado porque associa criação de valor, sucesso de serviços digitais e satisfação dos usuários.

O modelo de uso continuado de sistemas de informação de Bhattacharjee (2001) (Figura 2) aplica a teoria de expectativa e confirmação (Oliver, 1980) ao contexto de serviços digitais.

Nesse modelo, os usuários acessam o serviço com uma expectativa inicial, formam percepções a respeito durante o uso e, ao final, as comparam às expectativas. Se houver confirmação positiva, os usuários formam uma satisfação em relação ao serviço, o que gera o uso contínuo, ao passo que os usuários insatisfeitos abandonam o serviço.

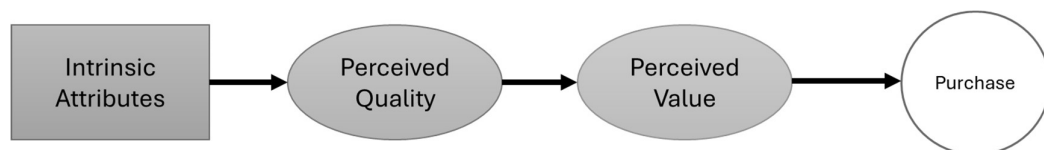
Figura 2 – Modelo de uso continuado de serviços digitais Bhattacharjee (2001), após o uso inicial



Fonte: Li e Shang (2020, p. 6).

O modelo da cadeia meio-fim (*means-end model*), proposto por Zeithaml (1988) (Figura 3), relaciona qualidades de um produto, seu valor percebido pelos consumidores e a compra de um produto (medida de sucesso do produto no mercado) e é uma evolução do modelo de Dodds e Monroe (1985). Amplamente utilizado no marketing da iniciativa privada, esse modelo postula que o comportamento de compra dos consumidores é determinadamente influenciado pela percepção de preço, qualidade e valor. No modelo de Li e Shang (Figura 5), a relevância está na relação entre qualidade e valor, considerando que o preço, por ser socializado, não é um fator determinante.

Figura 3 – Modelo de cadeia meio-fim relacionando preço, qualidade e valor (Zeithaml, 1988)

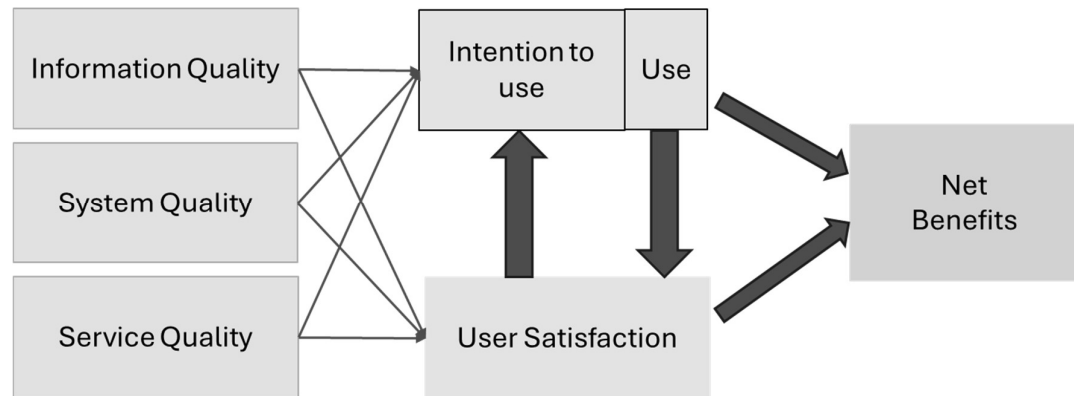


Fonte: Li e Shang (2020, p. 6).

O modelo de DeLone e McLean (2003) (Figura 4), originalmente desenvolvido em 1992 e revisado em 2003, define o sucesso de um sistema a partir da intenção de uso continuado e na satisfação dos usuários. Para sustentar uma avaliação positiva, o sistema deve apresentar atributos de qualidade (agrupados em qualidade de informação, qualidade do sistema e qualidade do serviço). O modelo indica que, se um sistema tiver os atributos de qualidade necessários e os usuários estiverem satisfeitos, se pode medir uma intenção de uso

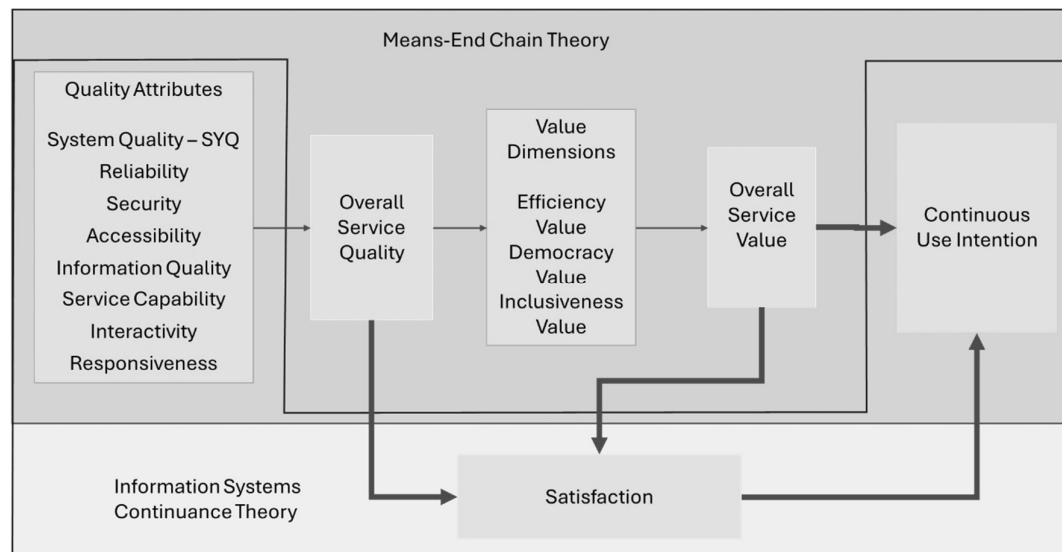
continuado e a adoção efetiva pelos usuários, gerando benefícios para a organização da qual o sistema faz parte. No entanto, se houver queda nos atributos de qualidade ou na satisfação do usuário, a intenção de uso do sistema deve cair.

Figura 4 – Modelo de DeLone e McLean, revisado em 2003



Fonte: DeLone e McLean (2003, p. 24).

O modelo de Li e Shang (2020) combina o modelo de uso continuado de sistemas de informação de Bhattacharjee (2001) com o modelo da cadeia meio-fim (Zeithaml, 1988). Nesse modelo, as qualidades do sistema de informação de DeLone e McLean (2003) substituem a qualidade do produto de Zeithaml (1988). O valor percebido do modelo meio-fim é adaptado para os valores públicos de eficiência, democracia/participação e inclusividade. E, por fim, a intenção de uso continuado de Bhattacharjee (2001) ocupa o lugar da “compra” do modelo de Zeithaml (1988). Esse modelo (ilustrado de forma gráfica na Figura 5) é relevante porque associa qualidades do serviço a valores/benefícios para usuários e a consequente criação de valor público que pode ser identificada pela intenção de uso continuado do serviço.

Figura 5 – Representação do modelo Li e Shang (2020) e dos modelos dos quais foi derivado

Fonte: Li e Shang (2020, p. 7).

Examinando os componentes desse modelo (Figura 5), temos o Quadro 1 a seguir:

Quadro 1 – Atributos e variáveis do modelo Li e Shang (2020)

Atributos de qualidade	Qualidade geral do serviço (SQ)	Variáveis dependentes sobre valor	Valor geral do serviço (PSV)	Intenção de uso continuado (CUI)
Qualidade do sistema (SYQ)		Eficiência (EFV)		
Confiabilidade (REL)		Democracia (DEV)		
Segurança (SCT)		Inclusividade (INV)		
Acessibilidade (ACS)		Satisfação (SAT)		
Qualidade da informação (IQ)				
Capacidade (SEC)				
Interatividade (INT)				
Responsividade (RPS)				

Fonte: Elaborado pelo autor (2024).

Do lado esquerdo do Quadro 1, estão os atributos (variáveis independentes¹⁸) de qualidade do serviço, que são mediados pela variável de Qualidade geral do serviço (SQ). Esta variável influencia tanto as variáveis dependentes de valor quanto a Satisfação dos usuários (SAT). As variáveis dependentes de valor (EFV, DEV e INV) são mediadas pela variável Valor geral do serviço (PSV) que, em conjunto com a variável SAT, impacta a variável dependente Intenção de uso continuado (CUI), evidenciando a geração de valor. Esse é o dado que se busca

¹⁸ Para aprofundamento no conceito de variáveis independentes, consultar a obra *Research Skills Reader*, Roelof Hars (2024). DOI: <https://doi.org/10.21827/67a4ceba2ceca>. Disponível em: <https://opentextbooks.rug.nl/rspremsc/>. Acesso em: 11 fev. 2025.

evidenciar neste artigo. Destaca-se, ainda, que o modelo é recente, validado em 2021 na China, e sem validação no Brasil.

Esse modelo foi criado para avaliar um conjunto de serviços governamentais, mas alguns aspectos não se aplicam ao contexto de avaliação de um único serviço, como é o caso do Balcão Virtual. O valor “Democracia” (DEV), por exemplo, refere-se à capacidade dos usuários de influenciar a instituição pública, e, no contexto de avaliação de um canal de atendimento, isso não se aplica, pois os jurisdicionados não usam o Balcão Virtual para fazer reclamações, sugestões ou demandas institucionais — papel que cabe à Ouvidoria do STJ. Dessa forma, o valor “Democracia” não será considerado.

2.3 Métodos e técnicas

2.3.1 Abordagem de modelo como mapa

Neste estudo, busca-se identificar (nos dados de atendimento, na pesquisa de satisfação dos usuários e em outras fontes) evidências da existência de atributos e variáveis do modelo de Li e Shang (2020) para avaliar se houve a criação de valor público ou não pelo Balcão Virtual. O objetivo deste artigo não é uma validação do modelo de Li e Shang (2020) no contexto brasileiro, e sim usá-lo como referência para identificar qualidades desejadas e áreas a serem aprimoradas.

A utilização de um modelo científico como mapa para tomada de decisão é uma prática conhecida, sendo o exemplo mais visível dessa abordagem o *Intergovernmental Panel on Climate Change* (IPCC)¹⁹, da Organização da Nações Unidas, que usa modelos científicos como base para a tomada de decisões na mitigação e adaptação às mudanças climáticas. Para uma revisão de literatura do assunto e uma apresentação das ferramentas mais importantes, sugere-se o artigo de Coali, Novelli, Sirigiri e Spina (2024).

É importante notar que o paradigma de “modelo como mapa” é útil quando os serviços não têm muitas evidências de criação de valor público, pois permite o direcionamento de ações de gestão. Se um serviço tem baixa intenção de uso continuado (variável mais à direita, evidenciando valor), por exemplo, pode-se observar se falta alguma qualidade das variáveis antecedentes mais à esquerda do modelo: como disponibilidade, capacidade, usabilidade; ou boa qualidade no serviço, mas atendimentos não valorizados (ex.: falta de evidência de geração de valor, atendimentos não eficientes ou que atinjam poucos usuários). Nesses casos, é possível usar os itens do questionário do modelo (Anexo A) nessas áreas específicas para se levantar as

¹⁹ Homepage do IPCC disponível em: <https://www.ipcc.ch/about/>. Acesso em: 6 abr. 2025.

lacunas e tomar ações corretivas. Além disso, os atributos e as variáveis do modelo têm pesos (coeficiente beta²⁰) que indicam quanto o atributo ou a variável influenciam as variáveis dependentes. Assim, os gestores podem investir mais tempo e recursos no aprimoramento de atributos que tenham maior peso.

2.3.2 Tratamento de dados

Os registros de atendimento são a principal fonte de informação. É deles que se deriva dados como: áreas do Tribunal com mais demanda, tempos de atendimento e horários mais requisitados bem como chaves para enriquecimento dos dados com consultas processuais (na Seção 3.5, é possível visualizar um exemplo de dados disponíveis na consulta processual).

A Coordenadoria de Atendimento e Protocolo Judicial (CATJ) disponibilizou os registros de atendimento do Balcão Virtual até dezembro de 2023 e de outros canais de atendimento (Espaço do Advogado, e-mail e telefone). Os dados originais são uma planilha em formato CSV²¹ extraída do banco de dados do sistema do Balcão Virtual. Cada linha da planilha representa um atendimento que contém: a data e a hora de início e fim do atendimento; os dados de identificação do usuário; os dados da pessoa que realizou o atendimento pelo STJ; e até 3 assuntos tratados no atendimento, entre outros dados para uso dos gestores do Balcão Virtual. Uma amostra de alguns registros de atendimento anonimizados está disponível no Apêndice F.

Esses dados foram alimentados em um sistema de programas próprio em *Python*²², e a escolha por essa linguagem se deu não só por experiência própria prévia, mas também por seu *framework* de *Data Science* robusto (*Pandas*²³) e por ser mais eficiente do que o Microsoft Excel® para tratamento complexo de dados. O tratamento envolveu:

1. Eliminar dados inválidos (ex.: datas inconsistentes, tempos de atendimento excessivos, valores nulos);
2. Analisar os textos dos atendimentos para identificar e separar os números dos processos mencionados nos atendimentos;

²⁰ Para consultar conceito de “coeficiente beta”, consultar *The Cambridge Dictionary of Statistics 4th Edition*, Everitt e Skrondal (2010, p. 43).

²¹ CSV é um padrão de internet, definido pelo RFC 4180 m-. Para se aprofundar no conceito, consultar: <https://datatracker.ietf.org/doc/html/rfc4180.html>. Acesso em: 8 mar. 2025.

²² *Python* é uma linguagem de programação de alto nível, interpretada e de sintaxe simples, focada na legibilidade e produtividade. É amplamente usada em diversas áreas, como desenvolvimento web, ciência de dados, automação e inteligência artificial.

²³ *Pandas* é uma biblioteca de *Python* usada para manipulação e análise de dados. Ela fornece estruturas eficientes, como *DataFrame* (tabelas de dados) e *Series* (listas indexadas), facilitando o tratamento, limpeza e análise de grandes conjuntos de dados.

2.1. No caso de intenção de falar com algum Gabinete, identificar qual Ministro foi mencionado.

3. Salvar os dados tratados em formato mais eficiente para posterior manipulação (*Parquet*²⁴ para *Python* e *xlsx* para Microsoft Excel®);

4. Filtrar os registros que mencionam processos e conduzir uma consulta processual na página do STJ para obter mais detalhes (ex.: Relator, Tribunal de origem, área do Direito e partes no processo); e

5. Armazenar de forma eficiente em *Python* os dados extraídos das consultas processuais (armazenamento de cada processo como um objeto em *Python* e, depois, serialização em um arquivo *pickle*²⁵ da lista de todos os objetos).

Devido ao volume de dados (17.035 consultas), o programa de consulta processual foi executado em fins de semana com pausa entre cada consulta para evitar sobrecarga no sistema do STJ e bloqueios de segurança — inclusive, este tipo de atividade pode ser interpretado como um ataque de negação de serviço²⁶. As consultas processuais levaram três dias para serem efetuadas. Os resultados das consultas também foram armazenados em disco e o programa verificou se uma consulta já havia sido gravada antes de consultar o sistema do STJ, o que evitou consultas desnecessárias e acelerou a extração de dados de iterações anteriores.

Após os dados estarem armazenados de forma eficiente, as análises para se encontrar as evidências almejadas foram feitas usando programas específicos em *Python* que produzem saídas em formato Microsoft Excel® para confecção de gráficos e verificação. Por exemplo, para elaborar o histograma do tempo de atendimento, foi escrito um programa²⁷ que lê os dados em formato *Parquet*, varre toda a lista de atendimentos e cria uma tabela de frequência dos tempos de atendimento. Os dados, então, são gravados em formato compatível com Microsoft Excel® para confecção do gráfico e, por fim, os dados do histograma são comparados aos originais por amostragem para verificar se não há erros no programa (ex.: se há 47 atendimentos entre 1.741 segundos e 1.800 segundos, os dados originais são filtrados nessa faixa de tempo de atendimento para confirmar se são exatamente 47 atendimentos).

²⁴ *Parquet* é um formato de arquivo colunar otimizado para armazenamento e processamento eficiente de grandes volumes de dados, amplamente usado em Big Data e ciência de dados por oferecer compressão eficiente e acesso rápido às colunas necessárias, reduzindo o uso de memória e melhorando a performance em comparação com formatos como CSV.

²⁵ *Pickle* é um formato de arquivo para armazenamento eficiente na linguagem *Python*. A documentação oficial do arquivo pode ser consultada em: <https://docs.python.org/3/library/pickle.html>. Acesso em: 8 mar. 2025.

²⁶ Para se aprofundar no assunto, basta consultar notícia da Agência de Defesa Americana, 2021. Disponível em: <https://www.cisa.gov/news-events/news/understanding-denial-service-attacks>. Acesso em: 8 mar. 2025.

²⁷ O código fonte do software pode ser consultado no Apêndice G.

2.4 Resultados e discussões

Para responder à pergunta de pesquisa e atingir o objetivo geral, adotou-se o modelo de Li e Shang (2020) como “mapa”, avaliando-se cada variável desse modelo. A análise parte das variáveis independentes (localizadas mais à direita no modelo), passando pelas variáveis dependentes até se chegar na variável dependente: a intenção de uso continuado, considerada o fator de sucesso do serviço. Os objetivos específicos são abordados ao longo do processo de análise dos dados para o atingimento do objetivo geral e das análises suplementares. Os resultados estão apresentados nesta seção em quatro etapas: (i) descrição das fontes de informação; (ii) evidências da presença ou não das variáveis do modelo de Li e Shang (2020); (iii) discussão do objetivo geral e dos objetivos específicos; e (iv) resposta à questão de pesquisa.

2.4.1 Fontes de informação

As fontes de informação do Balcão Virtual utilizadas são:

1. Registros de atendimento do Balcão Virtual: 22.275 registros originais, entre as datas de 13/05/2021 e 14/12/2023. Os registros com descrições detalhadas dos atendimentos iniciam-se em 24/08/2021, pois os anteriores incluem um período de testes.

2. Respostas do questionário de avaliação do serviço enviados pela Ouvidoria do STJ: 3.953 pesquisas de satisfação respondidas (em média, 1 em cada 5,63 atendimentos tem uma pesquisa de satisfação respondida, ou seja, 17,75% do total de registros). Salienta-se que os resultados de satisfação dos usuários podem conter viés, pois a resposta dos questionários é voluntária e anônima, isto é, não há garantia de que as respostas sejam igualmente distribuídas pelos usuários atendidos pelo Balcão Virtual. Além disso, usuários que não conseguem acessar o serviço não recebem o questionário, o que pode afetar a representatividade das respostas.

3. Média do indicador “Disponibilidade do Portal” do Plano Diretor de Tecnologia da Informação e Comunicação de 2023-204 (Superior Tribunal de Justiça, 2024b, p. 33) para o período dos registros de atendimento (maio de 2021 a dezembro de 2023).

2.4.2 Análise das variáveis do modelo: qualidade do serviço

A seguir, apresenta-se a análise de cada atributo e variável do modelo de Li e Shang (2020) aplicados ao Balcão Virtual e as evidências de sua presença ou não nas fontes de informação usadas.

Dos atributos de qualidade do serviço, temos:

1. Qualidade do sistema (SYQ): refere-se à facilidade de acesso ao Balcão Virtual, qualidade dos links, navegabilidade e usabilidade. Dado que o Balcão Virtual opera via videoconferência, sem necessidade de navegação entre telas, e a identificação do usuário é voluntária (sem login formal), considera-se que esse atributo está presente.

2. Confiabilidade (REL): relaciona-se ao tempo de indisponibilidade do serviço (se o portal está indisponível, então o Balcão Virtual estará indisponível). Embora não haja monitoramento específico do Balcão Virtual, o indicador de disponibilidade do portal do STJ, mantido pela Secretaria de Tecnologia da Informação e Comunicação (STI), serve como referência. Entre 13/05/2021 e 14/12/2023, a disponibilidade média do portal foi próxima de 100% (com exceção de julho de 2021, 99,99%, e janeiro de 2022, 97,55%). No entanto, a disponibilidade do portal não garante a do Balcão Virtual. A fim de se avaliar a confiabilidade do serviço, foram analisados os registros de atendimento com “Conexão Interrompida” — um indicador indireto, tendo em vista que a falha pode ocorrer do lado do usuário também. Dos 22.725 atendimentos, apenas 268 (1,2%) foram interrompidos, resultando em uma disponibilidade de 98,8%, próxima à meta do STJ (99%). Assim, considera-se que o atributo confiabilidade está presente.

3. Segurança (SCT): apesar de não haver relatos de problemas de segurança nem incidentes de segurança pela Coordenadoria de Segurança da Informação e Defesa Cibernética (CSID) no período analisado, as questões do modelo (disponíveis Anexo A) versam sobre a percepção dos usuários quanto à proteção dos dados no STJ, a segurança no tratamento dos dados pessoais enviados ao Balcão Virtual e a existência de medidas de segurança adequadas. No entanto, a pesquisa de satisfação do Balcão Virtual não inclui questões relativas à percepção de segurança. Assim, por falta de evidências, esse atributo é considerado ausente. Ademais, sugere-se a inserção de perguntas específicas no questionário de controle de qualidade para preencher essa lacuna e averiguar essa percepção de segurança e privacidade dos usuários do Balcão Virtual.

4. Acessibilidade (ACS): o questionário avalia três aspectos dessa variável: (1) acesso a informações de como usar o serviço; (2) compatibilidade com o computador do usuário; e (3) necessidade de configurações complexas. Os requisitos técnicos do Balcão Virtual são modestos do ponto de vista computacional (são os mesmos requisitos do Zoom, que é usado para videoconferências, sem necessidade de configurações especiais). Entretanto, a exigência de banda larga para videoconferências pode ser uma barreira em algumas regiões devido ao custo.

Quanto à disponibilidade de informações sobre o uso do Balcão Virtual, o link de acesso está na página inicial do STJ (<https://www.stj.jus.br>) e há vídeos explicativos no YouTube sobre como funciona o procedimento de acesso (Atendimento Judicial por Videoconferência – Balcão Virtual do STJ).

A pesquisa de satisfação inclui o item “Ferramenta de videochamada (Zoom)”, que obteve média de 9,61 em uma escala de -1 a 10, conforme apresentado na Tabela 1.

Tabela 1 – Ferramenta de videochamada (Zoom)

Ferramenta de videochamada (Zoom)	
Contagem	3953
Mínimo	-1
Máximo	10
Média	9,61
Desvio padrão	1,15
Nota 10	81,94%
Nota 9	8,98%
Nota 8	4,78%
Notas 10-8	95,70%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Por essas evidências, esse atributo é considerado presente.

5. Qualidade da informação (IQ): as questões do modelo avaliam websites quanto à qualidade e atualidade das informações. O Balcão Virtual utiliza o mesmo sistema de informação dos Ministros do STJ, garantindo que a informação provida aos usuários seja a mais atualizada possível. Ademais, a pesquisa de satisfação inclui a questão “Qualidade das informações prestadas” (Tabela 2) que tem uma resposta média de 9,76 (na escala de 0-10), indicando uma percepção positiva dos usuários quanto às informações prestadas.

Tabela 2 – Qualidade das informações prestadas

Qualidade das informações prestadas	
Contagem	3.946
Mínimo	-1
Máximo	10
Média	9,76
Desvio padrão	1,06
Nota 10	90,47%
Nota 9	5,09%
Nota 8	2,23%
Notas 10-8	97,80%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Assim sendo, esse atributo é considerado presente.

6. Capacidade (SEC): esse atributo se refere à capacidade do serviço de realizar os atendimentos em um só lugar (*one stop shopping*). No Balcão Virtual, há uma pessoa especializada para atender e auxiliar na busca por informações desejadas nos sistemas do STJ. Por isso, considera-se esse atributo presente.

7. Interatividade (INT): essa variável se aplica a serviços não interativos, como e-mail, em que há um intervalo entre a solicitação do usuário e a resposta. Como o Balcão Virtual é um serviço de videoconferência altamente interativo, esse atributo é considerado presente.

8. Responsividade (RPS): as perguntas originais visam avaliar a responsividade de websites governamentais sem atendentes humanos. O Balcão Virtual, por ser um serviço de videoconferência com atendentes humanos, é um serviço responsivo. Portanto, esse atributo é considerado presente.

Qualidade Geral do Sistema (SQ): variável que agrupa a influência dos atributos de qualidade do sistema. Não tem pergunta correspondente no questionário e Li e Shang. Como sete dos oito atributos de qualidade estão presentes, essa variável é considerada presente. A seguir, o Quadro 2 apresenta o resumo da presença ou ausência dos atributos.

Quadro 2 – Quadro-resumo da presença/ausência dos atributos de qualidade do serviço

Atributo	Presente ou ausente?
Qualidade do sistema (SYQ)	Presente
Confiabilidade (REL)	Presente
Segurança (SCT)	Sem evidência direta
Acessibilidade (ACS)	Presente
Qualidade da informação (IQ)	Presente

Capacidade (SEC)	Presente
Interatividade (INT)	Presente
Responsividade (RPS)	Presente
Qualidade geral do Sistema (SQ)	Presente

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

2.4.3 Análise das variáveis do modelo: valores

Quanto às variáveis dependentes relacionadas a valores do Balcão Virtual, temos:

1. Democracia (DEV): o valor “democrático”, no contexto do modelo de Li e Shang (2020), refere-se à aproximação de assuntos de interesse público, como transparência, prestação de contas (*accountability*) e informações sobre ações governamentais. Como esses assuntos estão fora do escopo do Balcão Virtual (no STJ, são de responsabilidade da Ouvidoria e da página de Transparência), esse atributo não foi considerado na avaliação.

2. Eficiência (EFV): eficiência refere-se ao potencial de economia de tempo, dinheiro e esforço. Para avaliá-lo, utiliza-se o indicador da pesquisa de satisfação da Ouvidoria (Apêndice A). “Duração do atendimento”, que tem média de 9,73 (em uma escala de -1 a 10). A Tabela 3 apresenta os dados estatísticos desse indicador:

Tabela 3 – Duração do atendimento

Duração do atendimento	
Contagem	3.945
Mínimo	-1
Máximo	10
Média	9,73
Desvio padrão	1,08
% Nota 10	88,52%
% Nota 9	6,02%
% Nota 8	2,81%
% Notas 10-8	97,34%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

A fim de se estabelecer quanto é “um bom tempo de atendimento”, realizou-se uma análise dos registros de atendimento para encontrar o tempo médio e a distribuição dos tempos de atendimento. O tempo médio encontrado (Tabela 4) foi de 473,31 segundos (com desvio padrão de 310,48 segundos).

Tabela 4 – Tempo médio de atendimento do Balcão Virtual

Tempo de atendimento	
Contagem	21.441
Mínimo	0

Máximo	1.843
Média	473,31
Moda	300
Desvio padrão (população)	310,48

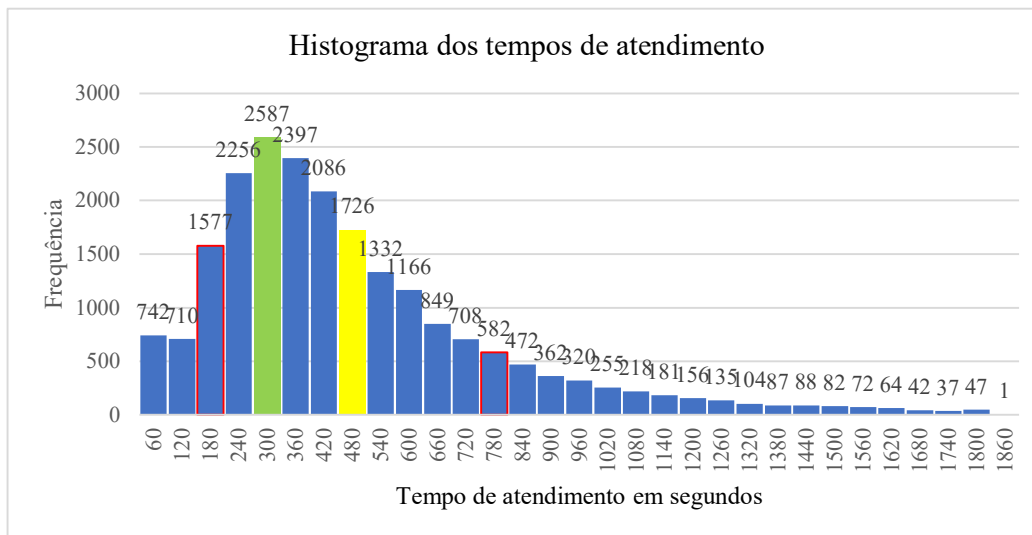
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

A distribuição dos tempos de atendimento é ilustrada no histograma do Gráfico 1:

- A barra em verde indica o tempo de atendimento mais frequente em segundos (em estatística, essa medida é a moda).
- A barra amarela representa a média.
- Os tempos de atendimento são agrupados em barras de 60 segundos de duração.

Os *outliers* (atendimentos muito fora da curva, com mais de 3 desvios padrão, ou 0,3% dos atendimentos²⁸) foram removidos para evitar que a média fosse enviesada por atendimentos incomuns. Por não ser possível vincular as respostas do questionário da Ouvidoria a atendimentos específicos, assume-se que as respostas estão na distribuição normal dos tempos de atendimento. Salienta-se que a complexidade dos atendimentos varia, sendo assim, um tempo adequado é aquele dentro de um desvio padrão da média (68% dos atendimentos). Para o caso do Balcão Virtual, considerando-se os dados disponíveis, o esperado seria entre 162,83 segundos (média 473,31 segundos – desvio padrão 310,48 segundos = 162,83 segundos) e 783,79 segundos (média 473,31 segundos + desvio padrão 310,48 segundos = 783,79 segundos). Esses limites estão assinalados no Gráfico 1 pelas barras com a borda em vermelho.

²⁸ Esse conceito pode ser aprofundado em: *Scientific Research and Methodology: An introduction to quantitative research in science and health* (Dunn, 2021, cap. 13.5). Disponível em: <https://bookdown.org/pkaldunn/Book>. Acesso em: 8 mar. 2025

Gráfico 1 – Histograma do tempo de atendimento do Balcão Virtual

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Com base nessas evidências, essa variável é considerada presente.

3. Inclusão (INV): o valor de inclusão social refere-se a facilitar o acesso a mais cidadãos, garantindo igualdade e aumento de acesso a grupos vulneráveis. A inclusividade é multifacetada: pode-se falar de inclusividade econômica, geográfica, digital e de pessoas com deficiência. A nível institucional, a inclusão pode ser abordada a partir de uma variedade de canais de atendimentos para cobrir segmentos diferentes dos usuários, como é o caso do STJ, que provê atendimento presencial, telefônico, por e-mail e, por fim, o Balcão Virtual (videoconferência).

Antes de se discutir o aspecto de inclusão social do Balcão Virtual, examinaram-se os dados disponíveis quanto ao perfil do usuário e ao dispositivo de acesso mais utilizado (Tabela 5):

Tabela 5 – Perfis de acesso ao Balcão Virtual

Perfil	Contagem	Percentual
Advogado(a)	17.222	78,42%
Estagiário(a) de Direito	2.467	11,23%
Parte em processo	998	4,54%
Cidadã(o)	575	2,62%
Tribunal/Vara	384	1,75%
Outra	283	1,29%
Servidor(a) público(a)	33	0,15%
Contagem total	21.962	100,00%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

A grande maioria (89,65%) dos usuários é advogado e estagiário de Direito, e o tipo de dispositivo mais utilizado é computador (85,81%) (Tabela 6), o que pode ser justificado pelas telas maiores serem mais produtivas para videoconferências, permitindo melhor visualização do operador e a leitura dos textos quando a tela é compartilhada.

Tabela 6 – Dispositivos de acesso ao Balcão Virtual

Dispositivo	Contagem	Percentual
Computador	18.845	85,81%
Celular	3.117	14,19%
Total	21.962	100,00%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Do ponto de vista de inclusão econômica, o Balcão Virtual não é o mais acessível. Ele exige: (1) conexão de internet banda larga (preço mínimo de R\$ 35 no Plano Nacional de Banda Larga²⁹ e média de R\$ 100 para planos de mercado³⁰) — já que a videoconferência demanda uma velocidade alta de transmissão para que a experiência do usuário seja agradável — e (2) computador (laptops mais acessíveis custam cerca de R\$ 1.800,00³¹). Assim, para usuários de menor renda, a opção mais econômica é o atendimento telefônico do STJ, que requer apenas telefone fixo ou celular. Por outro lado, para escritórios de advocacia, o acesso ao Balcão Virtual não incorreria em custos adicionais, tendo em vista que computador e internet banda larga já estão previstos nas despesas de negócio.

Em relação à inclusão geográfica, a cobertura do serviço de banda larga no Brasil ainda tem limitações³², especialmente em áreas rurais. Nesses casos, os usuários dependem do acesso de serviços via satélite, como o Starlink, que tem preços de instalação e mensalidade mais elevados³³ do que as ofertas das operadoras nacionais.

Além disso, o acesso ao Balcão Virtual requer alfabetização (para ler as telas) e letramento digital (o mínimo de conhecimento para navegar o site). Logo, em casos analfabetismo (digital ou não), a opção mais acessível é o atendimento telefônico.

²⁹ O Plano Nacional de Banda Larga, criado em 2020, foi uma iniciativa do Governo Federal para expandir o acesso à internet banda larga, especialmente em áreas carentes. Informação disponível em: <https://www.gov.br/anatel/pt-br/regulado/universalizacao/plano-nacional-de-banda-larga>. Acesso em: 19 fev. 2025.

³⁰ Preço retirado com base em novo Plano Estrutural de Redes de Telecomunicações (PERT) da Anatel. Disponível em: <https://teletime.com.br/22/03/2024/estudo-da-anatel-mostra-queda-de-precos-na-telefonica-e-banda-larga/>. Acesso em: 19 fev. 2025.

³¹ Preço retirado do site Qual notebook comprar. Disponível em: https://quenotebookcomprar.com.br/ofertas/?_sfm_spec_laptop_category=De+entrada&post_types=notebooks&ort_order=_sfm_sale_lowest-price+asc+num. Acesso em: 3 mar. 2025.

³² Segundo matéria da Agência Gov, em 2023, 5,9 milhões de domicílios do país não tinham internet. Disponível em: <https://agenciagov.ebc.com.br/noticias/202408/internet-foi-acessada-em-72-5-milhoes-de-domicilios-do-pais-em-2023>. Acesso em: 19 fev. 2025.

³³ Segundo site da Starlink, o preço de instalação é de R\$ 1.000,00. Disponível em: <https://www.starlink.com/br/residential>. Acesso em: 19 fev. 2025.

Para pessoas com deficiência auditiva, há suporte por meio de Libras e legendagem, além da criação da sala acessível do Balcão Virtual em abril de 2023 (Superior Tribunal Federal, 2023c)³⁴. Examinando os registros de atendimento, 94 usuários com deficiência foram atendidos (Tabela 7). No entanto, mensurar a inclusividade é uma tarefa desafiadora, pois usuários excluídos não são registrados nem nos atendimentos nem nos questionários de controle de qualidade da Ouvidoria.

Tabela 7 – Atendimentos de pessoas com deficiência

Deficiência?	Contagem	Percentual
Não	21.868	99,57%
Sim	94	0,43%
Total	21.962	100,00%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

A percepção de facilidade de acesso ao Balcão Virtual é bastante positiva entre os usuários. O indicador “Facilidade em acessar o Balcão Virtual do STJ” (Tabela 8) tem uma média é 9,60 (em uma escala de -1 a 10). Contudo, essa métrica não diferencia dificuldades relacionadas a deficiências e/ou barreiras tecnológicas.

Tabela 8 – Facilidade de acesso ao Balcão Virtual

Facilidade em acessar o Balcão Virtual do STJ	
Contagem	3.954
Mínimo	-1
Máximo	10
Média	9,60
Desvio Padrão	1,18
% Nota 10	82,04%
% Nota 9	8,73%
% Nota 8	4,63%
% Notas 8-10	95,40%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Por fim, convém lembrar que o objetivo do Balcão Virtual é atender à Resolução CNJ n.º 372 de 2021 dentro de suas limitações tecnológicas e promover a inclusão sempre que possível, e a implementação da sala acessível é um exemplo do compromisso do STJ em ampliar a acessibilidade do serviço.

Dadas as evidências apresentadas acima, essa variável é considerada presente. O Quadro 3 a seguir apresenta um resumo da presença ou ausência das variáveis.

³⁴ Matéria para consulta disponível em: <https://www.stj.jus.br/sites/portalp/Paginas/Comunicacao/Noticias/2023/20042023-Tribunal-lanca-Sala-Acessivel-no-Balcao-Virtual-para-atender-usuarios-com-deficiencia.aspx>. Acesso em: 19 fev. 2025.

Quadro 3 – Quadro-resumo da presença/ausência das variáveis dependentes

Variável	Presente ou ausente?
Democracia (DEV)	Não se aplica
Eficiência (EFV)	Presente
Inclusão (INV)	Presente

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

2.4.4 Análise das variáveis do modelo: variáveis dependentes

Variáveis dependentes:

1. Valor percebido do serviço (PSV): é uma variável de mediação entre os valores DEV, INV e EFV e a tanto a satisfação (SAT) e a intenção de uso continuado (CUI). Dado que as variáveis antecedentes (INV e EFV) e consequentes (SAT e CUI) estão presentes, pode-se assumir a presença dessa variável. No entanto, como não há evidências diretas, sua comprovação é definida como sem evidência direta. Recomenda-se aprimorar o questionário da pesquisa de satisfação incluindo uma questão sobre essa variável.

2. Satisfação (SAT): refere-se ao grau em que os usuários têm suas necessidades atendidas ou não e se se sentem satisfeitos com a experiência. Essa variável é derivada diretamente do indicador da “Satisfação geral – De uma forma geral, como você avaliaria o serviço ‘Balcão Virtual’?”, que apresenta uma média de 9,74 em uma escala de -1 a 10 (Tabela 9).

Tabela 9 – Satisfação com o Balcão Virtual

Satisfação geral - De uma forma geral, como você avaliaria o serviço "Balcão Virtual"?	
Contagem	3.953
Mínimo	-1
Máximo	10
Média	9,74
Desvio Padrão	1,00
% Nota 10	86,95%
% Nota 9	8,73%
% Nota 8	2,30%
% Notas 8-10	97,98%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Dada a evidência disponível, esse atributo é considerado presente.

3. Intenção de uso continuado (CUI): essa é a variável dependente do modelo e indica a geração de valor público em conjunto com SAT. Ela representa a intenção dos usuários de acessar novamente o serviço e recomendá-lo a outros usuários. No caso do Balcão Virtual, essa

variável pode ser identificada pelo indicador de *Net Promoter Score* (Reichheld, 2003), abordada pela questão “Qual a probabilidade de você recomendar este serviço a um amigo ou colega de profissão?” da pesquisa de satisfação. O resultado desse indicador foi de 9,81 em uma escala de -1 a 10 (Tabela 10).

Tabela 10 – Recomendação do Balcão Virtual para outros usuários

Qual a probabilidade de você recomendar este serviço a um amigo ou colega de profissão?	
Contagem	3.944
Mínimo	-1
Máximo	10
Média	9,81
Desvio Padrão	1,00
% Nota 10	92,85%
% Nota 9	4,08%
% Nota 8	1,37%
% Notas 8-10	98,30%

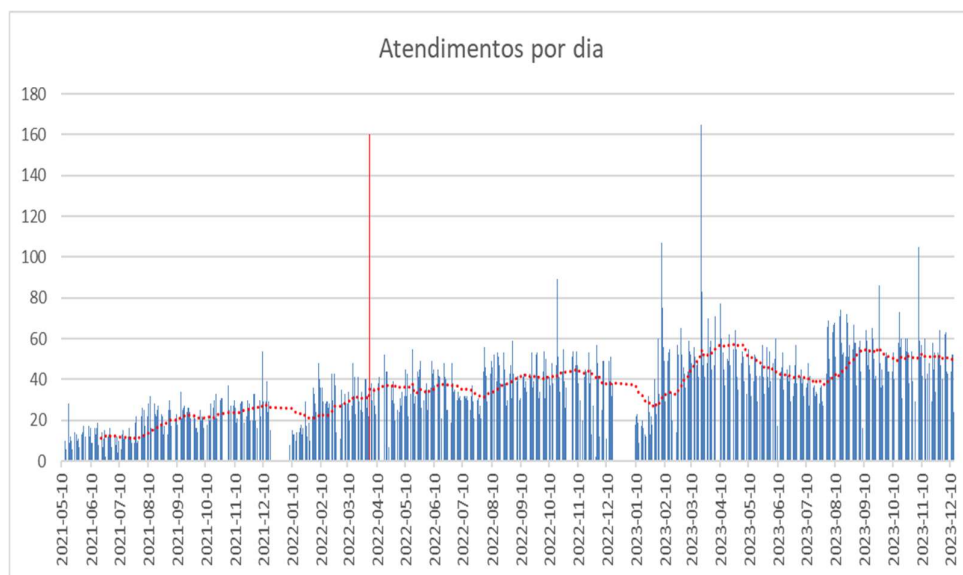
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Além disso, há mais evidências dessa variável na demanda do Balcão Virtual ao longo do tempo e na análise do número de atendimentos recorrentes, ou seja, usuários que retornaram ao serviço.

A demanda continuada é ilustrada no Gráfico 2:

- A linha vertical vermelha indica o retorno ao atendimento presencial no STJ (1º de abril de 2022, Resolução STJ/GP n.º 9 de 2022³⁵), após o fim da pandemia.
- As lacunas no gráfico são os recessos e dias sem atendimento.
- A linha vermelha pontilhada é a média móvel de 30 dias.

³⁵ Resolução disponível em: <https://bdjur.stj.jus.br/items/d82fc044-347e-459e-9de0-d7f18a853f88>. Acesso em: 8 mar. 2025.

Gráfico 2 – Demanda do Balcão Virtual 2021-2023

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Observa-se um crescimento da demanda do Balcão Virtual ao longo do tempo, com aparente estabilização da demanda em 2023. No entanto, é necessário haver mais dados para verificar se essa tendência de estabilidade é temporária ou não. Em termos numéricos, os atendimentos cresceram ao longo do tempo (Tabela 11).

Tabela 11 – Volume de chamados do Balcão Virtual

Medida	2021	2022	2023
Média diária	19,03	34,87	46,59
Total de atendimentos	2.862	8.099	10.697
Crescimento no período		182,98%	32,08%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Cabe ressaltar que a maioria dos processos são decididos em primeira e segunda instância, e apenas 2% dos processos chegam a tribunais superiores como o STJ³⁶. Dessa forma, o número de atendimentos recorrentes por usuário é menor do que na primeira e segunda instâncias do Judiciário. Em uma contagem conservadora³⁷, os dados da Tabela 12 mostram que 37,94% dos atendimentos foram realizados para usuários recorrentes (ou seja, que utilizaram o Balcão Virtual duas ou mais vezes). Considerando a população total de usuários (16.679 usuários), 18,29% deles (3.050 usuários) retornaram ao serviço mais de uma vez. Isso sugere que a intenção de uso continuado se traduziu em uso continuado efetivo do Balcão Virtual.

³⁶ Segundo matéria do Consultor Jurídico (2024b), de cada dez processos, nove terminam na primeira instância. Disponível em: <https://www.conjur.com.br/2024-mai-23/de-cada-dez-processos-nove-terminam-na-primeira-instancia/>. Acesso em: 20 fev. 2025.

³⁷ Como os usuários se identificam de várias formas (nome, CPF ou OAB e e-mail), o mesmo usuário pode ser contado como dois usuários diferentes (por exemplo, usando um nome com abreviatura uma vez e um nome completo outra vez). Logo, o número de usuários recorrentes é subestimado nessa contagem.

Tabela 12 – Número de chamados por usuário

	Contagem	%	Total de chamados	%
Usuários que chamaram uma vez	13.629	81,71%	13.629	62,06%
Usuários que chamaram mais de uma vez	3.050	18,29%	8.332	37,94%
Total	16.679		21.961	100,00%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Dado o alto nível de recomendação do Balcão Virtual, a demanda continua e a frequência do uso continuado, a variável dependente do modelo “Intenção de uso continuado” é considerada presente. O Quadro 4 a seguir apresenta um resumo da presença ou ausência das variáveis.

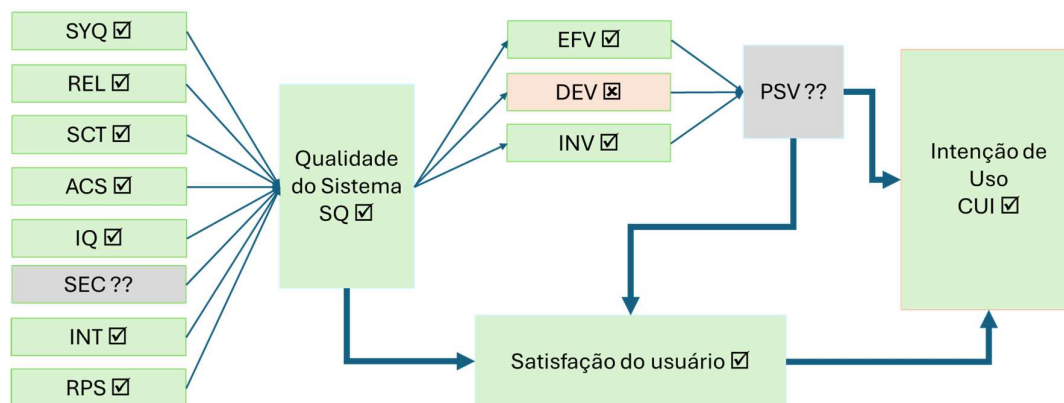
Quadro 4 – Quadro-resumo da presença/ausência das variáveis

Variável	Presente ou ausente?
Valor percebido do serviço (PSV)	Sem evidência direta
Satisfação (SAT)	Presente
Intenção de uso continuado (CUI)	Presente

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

A fim de se recapitular as informações apresentadas, a Figura 6 apresenta um resumo das variáveis e dos atributos encontrados ou não no Balcão Virtual.

Os atributos presentes estão em verde e com o símbolo do *checkmark* (☑); os que não se aplicam estão em vermelho com o símbolo da cruz (☒); e os que não se há evidências diretas suficientes estão em cinza, marcados com pontos de interrogação (??).

Figura 6 – Modelo representando as variáveis e os atributos com as evidências relativas ao Balcão Virtual

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

2.4.5 Objetivo geral

Dado que há evidências das variáveis representando os valores de eficiência (EFV), inclusividade (INV), satisfação dos usuários (SAT) e uma sólida evidência da intenção de uso continuado (CUI), pode-se afirmar que o Balcão Virtual gera valor público, que é o objetivo geral deste estudo.

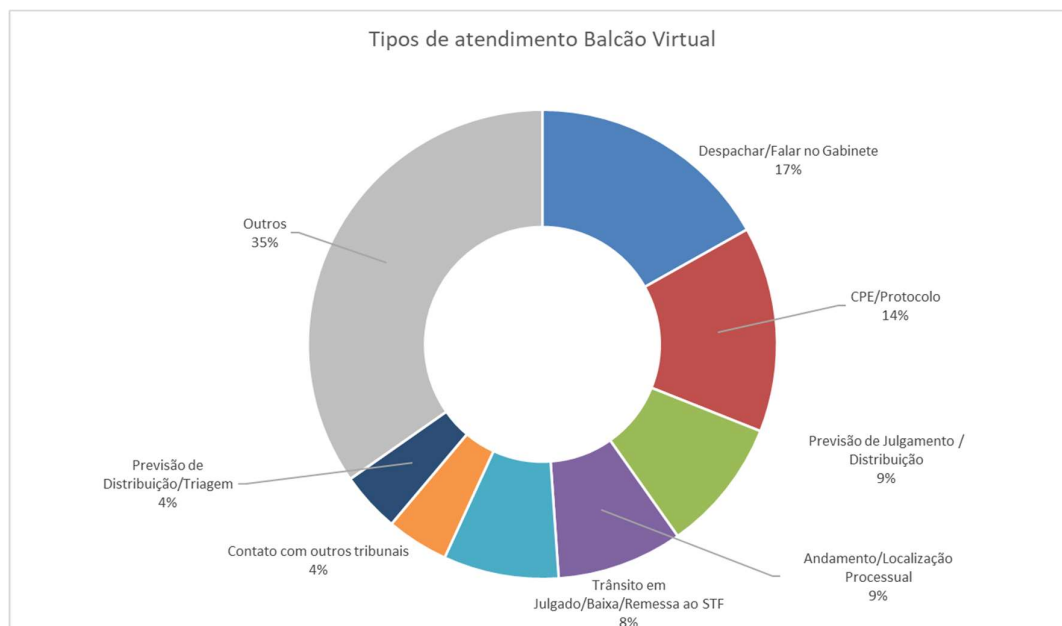
Avaliando os resultados à luz dos modelos que deram origem ao modelo de Li e Shang (2020), observa-se que:

- Pelo modelo de DeLone e McLean (1992), o serviço pode ser considerado um sucesso, pois os atributos de qualidade estão presentes, os usuários demonstram satisfação e isso se reflete no uso continuado.
- Pelo modelo de expectativa/confirmação de Bhattacharjee (2001), os usuários tinham uma expectativa de utilidade do Balcão Virtual que foi confirmada após o uso, que se reflete na satisfação e na intenção de uso continuado.
- Por fim, pelo modelo de meio-fim de Zeithaml (1988), os usuários têm uma percepção de qualidade do Balcão Virtual (atributos de qualidade) e um valor público percebido (eficiência e inclusividade) que culmina na “compra” — no caso, uso, uso continuado e recomendação do serviço a outros usuários.

2.4.6 Objetivo específico: demanda por expansão do Balcão Virtual para os Gabinetes de Ministros

Os gestores do Balcão Virtual do STJ observaram que, frequentemente, os usuários expressavam a vontade de entrar em contato com os Gabinetes de Ministros pelo Balcão Virtual. No entanto, atualmente, o Balcão Virtual está disponível somente para o Gabinete da Presidência do STJ, não abrangendo os demais ministros. Assim, para mensurar essa demanda, foi realizada uma análise quantitativa das solicitações desse tipo de atendimento (outras informações gerenciais mais detalhadas para a expansão do serviço para os Gabinetes foram incluídas no Capítulo 3).

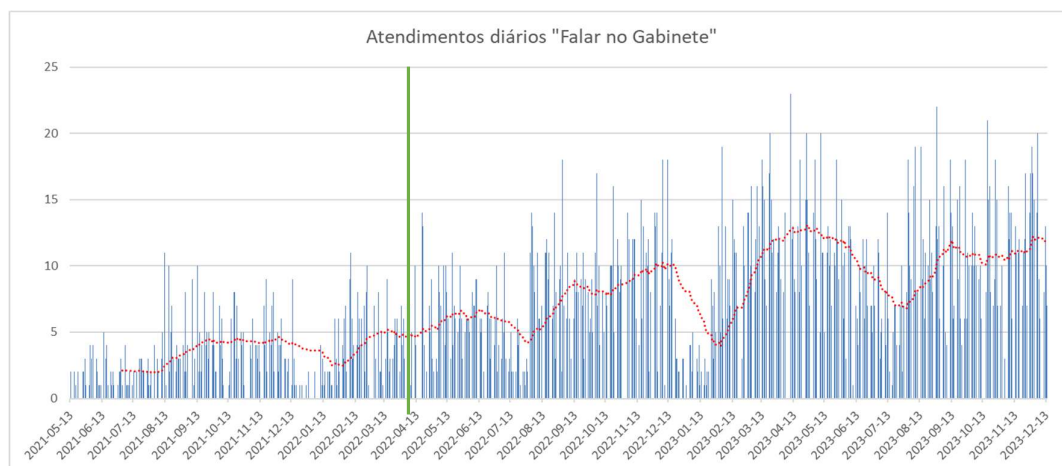
A demanda por atendimentos do tipo “Despachar/Falar no Gabinete” é a mais popular no Balcão Virtual — com 4.389 atendimentos (16,86% do total de atendimentos, arredondado para 17% no Gráfico 3). Esse dado reforça a justificativa de se expandir o Balcão Virtual para os Gabinetes do STJ.

Gráfico 3 – Tipos de atendimento do Balcão Virtual

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

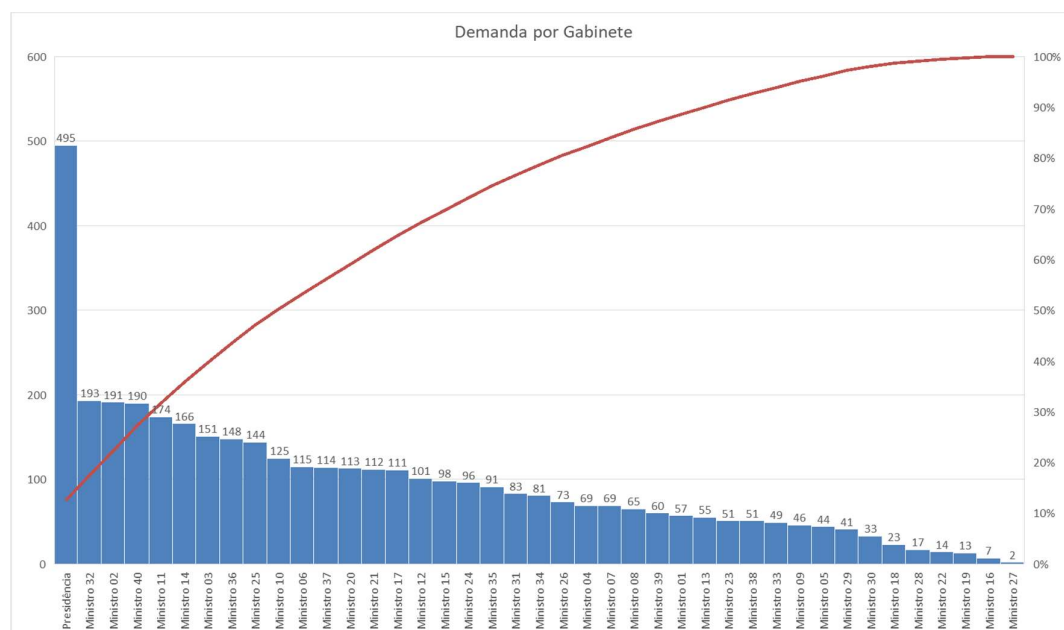
O Gráfico 4 apresenta a demanda por esse tipo de atendimento ao longo do tempo. A linha vermelha é a média móvel de 30 dias e demonstra uma clara tendência de aumento. A linha vertical verde indica o fim da pandemia e do retorno ao presencial³⁸ em 1º de abril de 2022. A demanda continuada é muito importante porque está diretamente relacionada à variável dependente do modelo — intenção de uso continuado (CUI) —, que, reiterando, é o principal indicador de valor público.

³⁸ A Resolução STJ/GP n.º 9, de 25 de março de 2022, estabeleceu o retorno do trabalho presencial no âmbito do Superior Tribunal de Justiça. Disponível em: <https://bdjur.stj.jus.br/items/d82fc044-347e-459e-9de0-d7f18a853f88>. Acesso em: 3 mar. 2025.

Gráfico 4 – Atendimentos diários de "Falar no Gabinete"

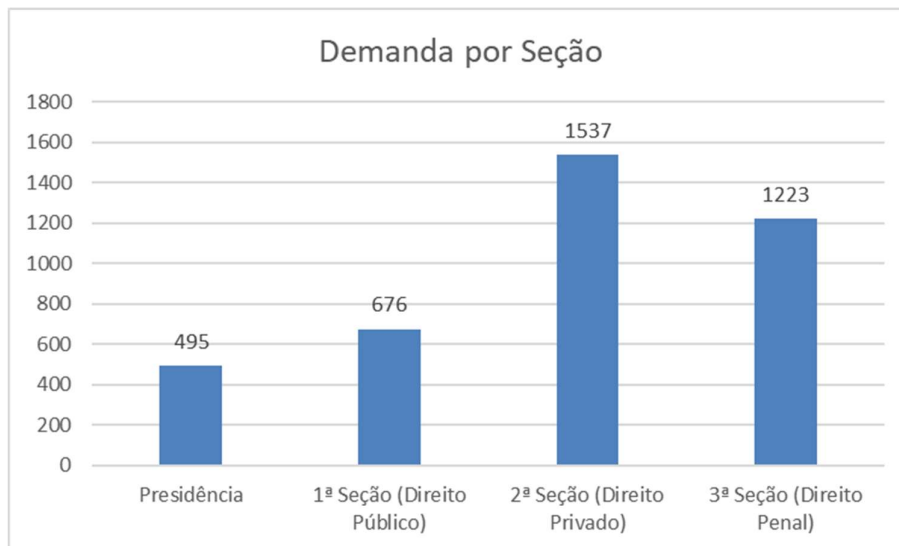
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Dos chamados em que foi possível identificar qual Gabinete o usuário pretendia contactar, foi elaborado o Gráfico 5, que apresenta a demanda por Gabinete. É possível verificar que o Gabinete mais demandado é o da Presidência. No entanto, não é possível afirmar se isso ocorre devido à natureza de suas funções ou, simplesmente, por ser o único Gabinete que já atende pelo Balcão Virtual, ao contrário dos outros, que atendem somente por telefone.

Gráfico 5 – Demanda por Gabinete (anonimizado)

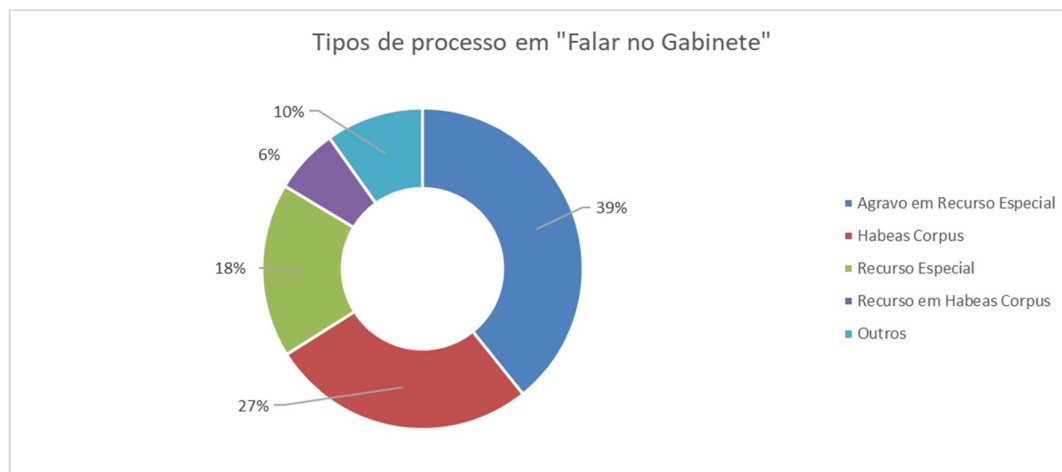
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

No Gráfico 6, está indicada a demanda agregada por seção. A maioria das demandas é para a área de Direito Privado e Direito Penal.

Gráfico 6 – Demanda por seção

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

No Gráfico 7, apresenta-se os tipos de processos referenciados nos atendimentos classificados como “Falar no Gabinete”. Conforme a demanda do STJ, a maior parte dos processos é de Agravos em Recurso Especial (AREsp), Recurso Especial (REsp) e processos relacionados a *Habeas Corpus* (*Habeas Corpus* e Recurso em *Habeas Corpus*).

Gráfico 7 – Tipos de processo

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Dadas as evidências acima, recomenda-se a expansão do serviço do Balcão Virtual para os Gabinetes de Ministros.

2.4.7 Objetivo específico: comparação com o serviço de atendimento telefônico

O atendimento telefônico é o canal mais popular no STJ. Por essa razão, foi realizada uma breve comparação entre o Balcão Virtual e o atendimento telefônico a fim de se evidenciar as diferenças de demanda entre os dois serviços. No entanto, essa análise enfrentou três limitações: (1) o serviço telefônico não é o foco deste trabalho; (2) os registros de atendimento disponíveis têm uma lacuna de quase um ano (321 dias); e (3) não há pesquisa de satisfação detalhada referente ao atendimento telefônico. Dito isso, os dados de atendimento do Balcão Virtual foram selecionados para coincidir com os dados disponíveis do atendimento telefônico (a comparação foi feita em dias que havia registros tanto do Balcão Virtual quanto do atendimento telefônico) e os resultados estão na Tabela 13.

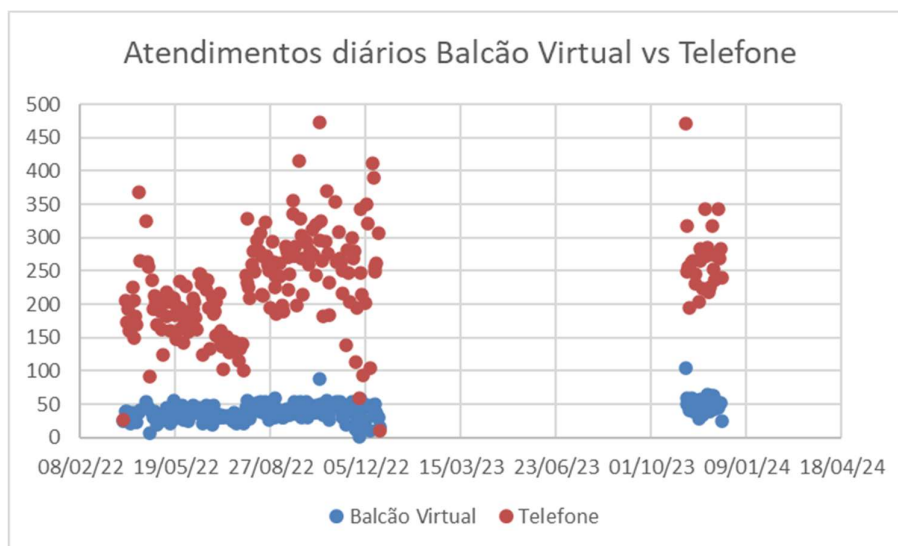
Tabela 13 – atendimentos diários (Balcão Virtual vs Telefone)

Serviço	Total de atendimentos	Média	Desvio padrão	Mínimo	Máximo
Balcão Virtual	8.108	39.17	12.46	2	105
Telefone	47.440	229.18	72.72	10	473

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Como se pode observar, o atendimento por telefone apresenta um volume muito maior do que o Balcão Virtual, na ordem de 5,85 vezes mais atendimentos por dia. O Gráfico 8 ilustra os atendimentos tanto do Balcão Virtual como por telefone, ficando evidente a lacuna de 321 dias nos registros de atendimento por telefone.

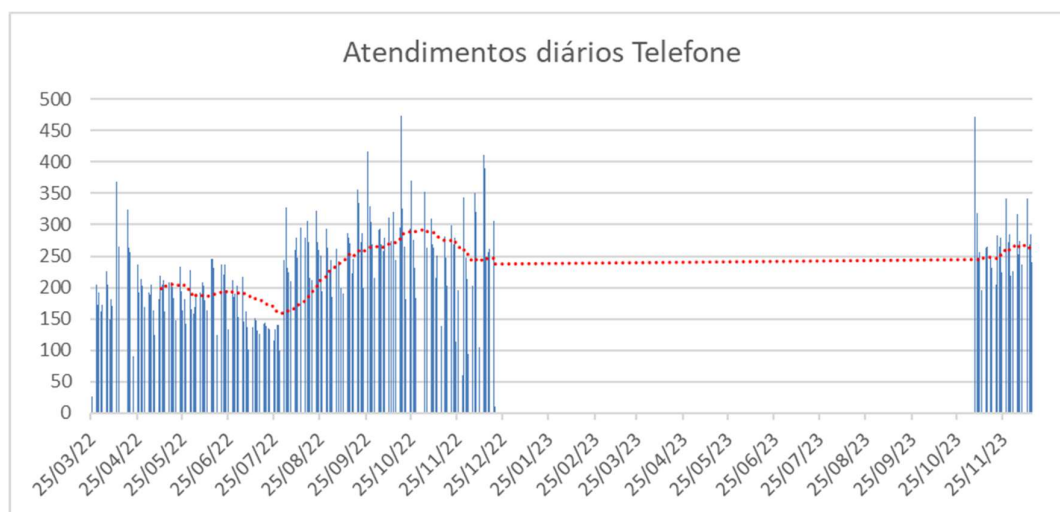
Gráfico 8 – Atendimentos diários (Balcão Virtual vs Telefone)



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

No Gráfico 9, são apresentados os atendimentos por telefone com linha de tendência de média móvel. Nele, pode-se observar um aumento de chamados na segunda metade de 2022, mas não há dados suficientes para determinar se há uma tendência de aumento constante ou se o aumento do segundo semestre de 2022 é um efeito sazonal.

Gráfico 9 – Atendimentos diários (telefone)



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

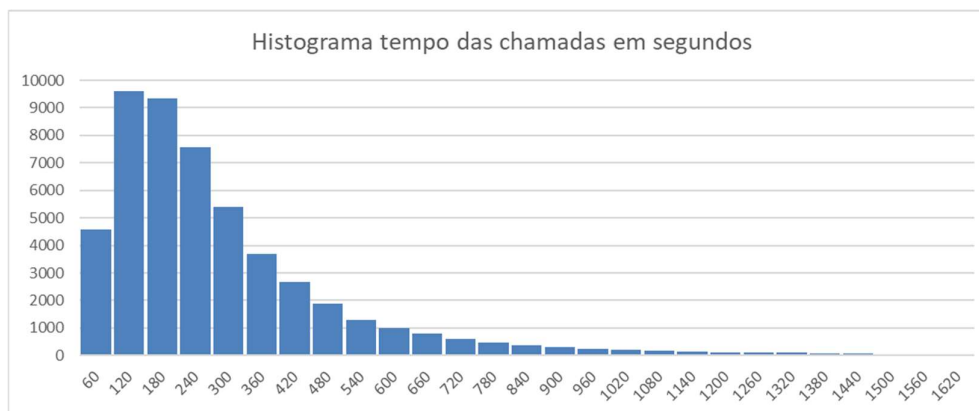
Quanto aos chamados recorrentes, foram registrados 51.979 no total, dos quais 23.301 são dos mesmos usuários, o que equivale a um percentual de 44,82% de recorrência.

As estatísticas do quesito de tempo médio do atendimento telefônico estão na Tabela 14, e no Gráfico 10 consta o histograma de duração do tempo de atendimento.

Tabela 14 – Estatísticas de tempo de atendimento telefônico

Tempo mínimo	Tempo máximo	Média	Desvio padrão	Contagem de atendimentos
4s	1.561s	262.45s	223.48s	50.823

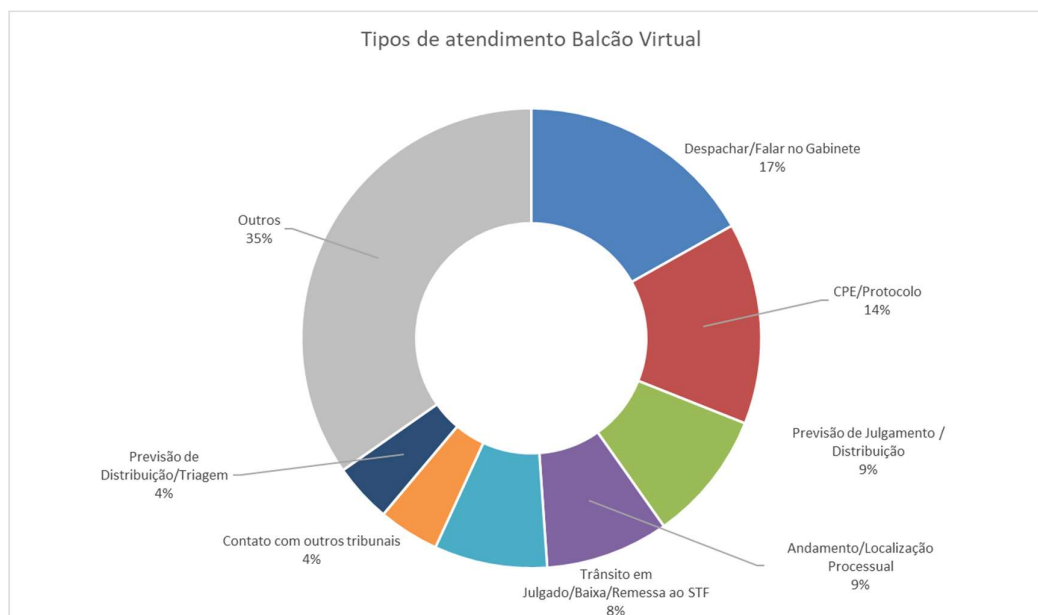
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 10 – Histograma do tempo de atendimento em segundos

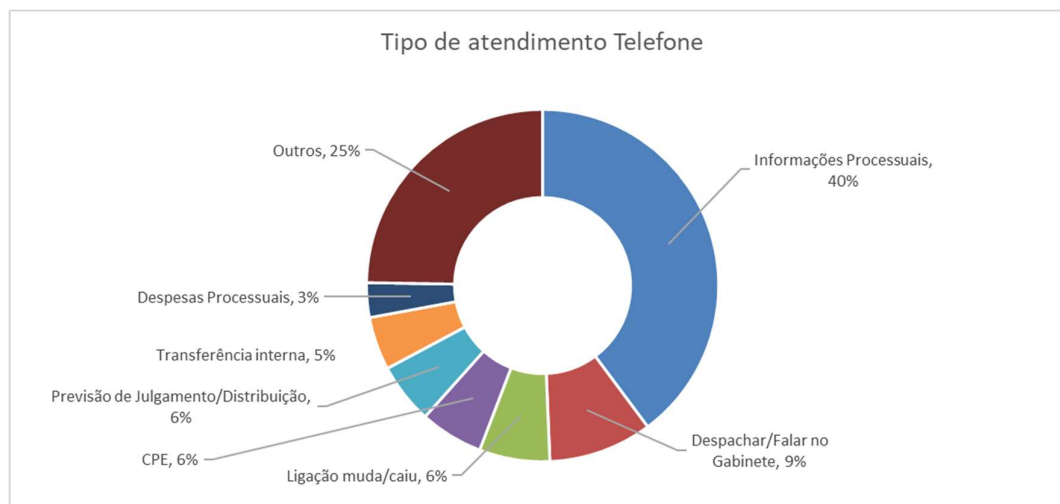
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

A comparação entre o Balcão Virtual e o atendimento telefônico, apresentada nos Gráficos 11 e 12, revela que o tipo de atendimento mais frequente no Balcão Virtual é “Despachar/Falar no Gabinete”, representando 17% do total. Já no atendimento telefônico, a categoria mais comum é “Informações Processuais”, responsável por 40% dos registros.

Contudo, em conversa com os gestores do Balcão Virtual, foi esclarecido que a categoria “Informações Processuais” no atendimento telefônico é pouco precisa — sendo utilizada tanto para consultas sobre processos quanto como categoria genérica quando não há uma classificação mais adequada. Diante disso, não é possível afirmar com confiança que os perfis de atendimento diferem entre os dois canais.

Gráfico 11 – Tipos de atendimento do Balcão Virtual

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 12 – Tipos de atendimento (telefone)

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Dadas as evidências acima, é possível concluir que o atendimento telefônico registra maior número de chamadas recorrentes — o que se relaciona à variável dependente do modelo de Li e Shang (2020), a intenção de uso continuado (CUI). Além disso, esse tipo de atendimento apresenta um tempo mais curto, o que se associa ao atributo de valor eficiência (EFV). Esses fatores indicam uma possível geração de valor público superior ao Balcão Virtual, mas a ausência da pesquisa de satisfação da Ouvidoria para o atendimento telefônico impede uma avaliação completa, especialmente quanto à satisfação dos usuários, elemento essencial para se aferir a qualidade dos serviços prestados.

2.4.8 Objetivo específico: recomendações de melhoria

Na jornada para a consecução do objetivo geral, foram identificadas oportunidades de melhoria, que compõem um dos objetivos específicos deste estudo e estão apresentadas a seguir:

1. Aprimoramento do questionário da Ouvidoria, para suprir lacunas das variáveis do modelo Li e Shang (2020), com questões baseadas no questionário do modelo de Li e Shang, referentes a:

a. Percepção de valor no uso do Balcão Virtual (variável PSV no modelo).

Sugere-se incluir a pergunta: “Levando-se em conta o tempo e esforço, valeu a pena utilizar o Balcão Virtual?”; e

b. Percepção de segurança das informações (variável SEC no modelo). Sugere-se incluir a pergunta: “Acredito que meus dados estão seguros nos sistemas do STJ e serão manipulados de maneira adequada?”.

2. Mensuração do tempo de espera antes do atendimento, para detecção de possíveis problemas de capacidade de atendimento do serviço. Também devem ser monitoradas desistências, isto é, quando um usuário tenta acessar o serviço, mas desiste/desconecta antes do atendimento se iniciar. No momento, só há registro de ligações interrompidas após o início do atendimento. Esse cenário precisa ser levado em consideração, pois:

a. Desistências ou dificuldades de acessar o serviço afetam a confirmação da expectativa de utilidade do serviço do modelo de Bhattacharjee (2002) e podem fazer com que os usuários abandonem o Balcão Virtual por outros canais de atendimento.

b. Para evitar a evasão, gestores precisam conhecer a taxa de desistência e o tempo máximo de espera tolerado.

3. Registro dos tempos de indisponibilidade do serviço, pois isso afeta o valor percebido, a evasão dos usuários e a imagem do Tribunal. Esse tipo de registro permite subsidiar ações de mitigação de instabilidade.

4. Melhoria na validação dos registros de atendimentos do Balcão Virtual. Atualmente, os registros apresentam algumas inconsistências (ex.: data de fim de atendimento no ano de 2142). É importante que os dados sejam validados na hora do registro para evitar que precisem ser corrigidos em um momento posterior. Uma possibilidade seria barrar a entrada de valores inválidos (como um ano diferente do corrente), de tempos de atendimento fora do horário de atendimento do Balcão Virtual e de tempos de atendimento com datas futuras.

5. Uniformização do registro de usuários. A identificação voluntária permite que um mesmo usuário se registre de diferentes formas (ora com nome completo, ora com o primeiro e último ou com os nomes intermediários abreviados, OAB, CPF, e-mails variados — até mesmo corporativos/públicos). Um exemplo dessa situação encontrada nos registros do atendimento por telefone está presente no Apêndice F. Isso prejudica a precisão da análise do uso continuado e a gestão integrada dos usuários. Nesse sentido, sugere-se utilizar o nome completo e OAB/CPF dos usuários validados com base no registro de partes da consulta processual ou de outro sistema do Tribunal.

Isso permite:

a. Medir com maior precisão o uso continuado;

b. Ter uma visão holística dos usuários em todos os canais de atendimento; e

c. Monitorar mudanças no comportamento dos usuários ao longo do tempo, ajudando a identificar tendências e necessidades de reestruturação dos serviços oferecidos. Por exemplo, pode-se questionar se a migração de um usuário de um canal de atendimento para outro foi causada por uma mudança nos serviços do STJ ou por alguma mudança tecnológica ou social, e, diante desse cenário, avaliar a necessidade de alterar os serviços nos canais ou, talvez, migrar recursos de um canal menos utilizado para um canal mais popular.

2.4.9 Resposta à questão de pesquisa

Avaliando os resultados pela ótica de valor público de Kearns (2004), entende-se que o Balcão Virtual é fonte de valor público, pois:

1. Entrega de serviços de alta qualidade, evidenciado nos resultados pela alta satisfação dos usuários (variável SAT).
2. Gera resultados (*outcomes*) positivos, representados pela intenção de uso continuado (variável CUI) e pela percepção de valor do serviço (variáveis DEV e INV).

Quanto à questão de pesquisa “É possível identificar a criação de valor público para o STJ pelo Balcão Virtual através de uma combinação de registros de atendimento, pesquisas de satisfação e dados públicos?”, devido aos resultados, a resposta é um “Sim” qualificado. Para isso, é necessário que a pesquisa de satisfação cubra as variáveis essenciais do modelo adotado. No caso do modelo de Li e Shang (2020), o questionário deve incluir, no mínimo, as variáveis relacionadas a valor público (eficiência e inclusividade), satisfação e intenção de uso continuado. Mas, para um diagnóstico mais completo, recomenda-se abranger todas as variáveis do modelo.

Quanto aos registros de atendimento, eles devem conter, pelo menos: a identificação do usuário (para se obter evidências de uso continuado), a data e hora de início e fim do atendimento (para determinar a tendência de uso do serviço) e o tempo de atendimento.

2.5 Conclusões e recomendações

O objetivo deste trabalho foi encontrar evidências de criação de valor público pelo serviço do Balcão Digital do STJ. Esse objetivo foi atingido, pois foram identificadas evidências da existência dos elementos mais importantes no modelo de valor proposto por Li e Shang (2020), em particular a qualidade geral do serviço, os aspectos do serviço que são valorizados pelos jurisdicionados (eficiência e acessibilidade) e, finalmente, evidências de uso continuado, intenção de uso continuado e recomendação do serviço para outros usuários.

Para alcançar esse resultado, foram fundamentais a disponibilidade de dados de atendimento de qualidade e os resultados de pesquisa de satisfação. A execução do estudo seguiu os seguintes passos: primeiro, reuniões com os gestores do Balcão Virtual para explicar os benefícios e quais dados seriam interessantes para eles, de forma a estabelecer uma relação positiva, que facilitou a execução deste trabalho. Após obtidos os dados brutos, devido ao seu volume e à inviabilidade de tratá-los por planilhas, foi elaborado um software em Ciência de Dados para analisar os registros de atendimento com programas específicos para sanitizar os dados e agregá-los. Esse processo revelou as evidências almejadas e as informações gerenciais que contribuíram para o produto técnico-tecnológico (Capítulo 3). Nessa jornada, foram identificadas lacunas de informação e oportunidades de melhoria, que foram apresentadas como resultado do objetivo específico deste estudo.

O Balcão Virtual é um serviço novo em comparação aos outros canais de atendimento que existem há mais de 20 anos e ainda não está claro se a demanda por esse serviço está estabilizada. Sendo assim, sugere-se investigar fatores relativos à adoção e possíveis resistências a novos serviços digitais, como falta de conhecimento do serviço por parte dos usuários, falta de valor adicional com relação a serviços existentes ou requisitos tecnológicos (computador e internet banda larga), e que não são considerados diretamente pelo modelo de Li e Shang (2020). Também sugere-se fazer um estudo longitudinal com os dados já disponíveis para avaliar o comportamento dos usuários recorrentes do Balcão Virtual. Isto é, se o uso recorrente tem alguma tendência definida (crescimento ou declínio) ou se segue algum tipo de padrão (interações em curto ou longo espaço de tempo).

Considerando que o modelo de Li e Shang (2020) é útil para avaliação de serviços digitais no contexto do governo brasileiro, sugere-se que o modelo seja validado empiricamente. Tal validação aumentaria a aplicabilidade do modelo, pois o Brasil tem diferenças significativas em relação à China — país onde o modelo foi originalmente testado — no que diz respeito à matriz cultural, ao sistema político e à estrutura econômica. Além disso, o processo de validação do modelo permite calcular os coeficientes de variância entre as variáveis do modelo, isto é, mensurar quanto uma variável independente influencia uma variável dependente. Isso permite saber quais variáveis têm mais impacto nos resultados, auxiliando os gestores a obter melhores resultados com os recursos disponíveis, direcionando-os de forma mais eficiente.

Por fim, a abordagem de “modelo como mapa” e o modelo Li e Shang (2020) podem ser aplicados em vários outros contextos, especialmente no Judiciário brasileiro, que possui serviços similares ao STJ. Dessa forma, o modelo pode orientar e acompanhar a inovação e a melhoria contínua dos serviços oferecidos pelos tribunais e beneficiar a sociedade brasileira.

3 PRODUTO TÉCNICO-TECNOLÓGICO: PROJETO DE ASSESSORIA PARA OS GESTORES DO BALCÃO VIRTUAL NO STJ

3.1 Introdução

Os gestores do Balcão Virtual do STJ gostariam de expandir os serviços prestados para outras áreas do STJ, em especial os atendimentos para os Gabinetes de Ministros da Corte. Isso não é uma tarefa fácil porque os Gabinetes de Ministro sofrem uma grande pressão para julgar processos. Só em abril de 2024, foram recebidos 54.127 processos³⁹, que, se fossem distribuídos igualmente, resultariam em uma média de 70,42 processos julgados por dia útil por Gabinete. Dada essa realidade, os Gabinetes são muito resistentes a alterações nos processos de trabalho que não tragam ganhos diretos de produtividade. No entanto, como demonstrado no Capítulo 2, o Balcão Virtual gera valor para o STJ, beneficia a sociedade e contribui para o objetivo estratégico de “Fortalecer a imagem e a transparência institucional”. Assim, este produto técnico-tecnológico de assessoria para os gestores do Balcão Virtual oferece informações que podem apoiar a implementação do Balcão Virtual nos Gabinetes, incluindo estratégias de adoção para melhorar as chances de aceitação e sucesso nessa empreitada.

Outros objetivos deste produto são:

- Ampliar a criação de valor público pelo Balcão Virtual; e
- Sugerir melhorias na gestão dos canais de atendimento do STJ, possibilitando a comparação entre os canais de atendimento e uma visão holística dos usuários atendidos.

3.2 Descrição geral do produto

Este projeto de assessoria foi executado em seis etapas:

1. Reunião com a CATJ para levantar necessidades do STJ e disponibilidade de dados;
2. Estudo teórico para identificar o referencial de avaliação dos dados fornecidos para maximizar o valor desse produto para o STJ;
3. Elaboração de software para sanitização, extração e armazenamento eficiente das informações coletadas;
4. Enriquecimento dos registros de atendimento com os dados de consulta processual;

³⁹ Dado retirado de notícia publicada no site do STJ, “Número de processos recebidos pelo STJ cresce 64% e bate recorde em abril”, em 13 de junho de 2024. Disponível em: <https://www.stj.jus.br/sites/portalp/Paginas/Comunicacao/Noticias/2024/13062024-Numero-de-processos-recebidos-pelo-STJ-cresce-64--e-bate-recorde-em-abril.aspx>. Acesso em: 03 mar. 2025.

5. Comparação do serviço do Balcão Virtual com o atendimento telefônico (a modalidade de acesso mais popular do STJ); e

6. Elaboração do relatório conclusivo de assessoria apresentado na Seção 3.5.

O Relatório Técnico Conclusivo de Assessoria contém as seguintes partes:

- Sumário executivo;
- Introdução;
- Recomendações
 - Recomendação: Expandir o Balcão Virtual para os gabinetes;
 - Recomendação: Incluir questão de percepção de valor do uso do balcão virtual no questionário de controle de qualidade da ouvidoria;
 - Recomendação: Incluir uma questão sobre percepção de segurança da informação no questionário de controle de qualidade da ouvidoria;
 - Recomendação: Mensurar o tempo de espera por atendimento;
 - Recomendação: Registrar os tempos de indisponibilidade do serviço;
 - Recomendação: Melhorar a validação dos registros de atendimentos do balcão virtual;
 - Recomendação: Identificar os usuários de maneira uniforme;
 - Recomendação: Melhorar o registro dos atendimentos nos canais de e-mail, telefone e Espaço do Advogado;
 - Recomendação: Aplicar questionário de controle de qualidade para outros canais;
 - Recomendação: Adaptar e aplicar o questionário do modelo de Li e Shang (2020) nos canais de atendimento do STJ
- Conclusão.

Além do objetivo específico da pesquisa teórico-empírica, este produto tem o objetivo específico adicional de buscar evidências para a expansão do Balcão Virtual no STJ.

3.3 Base teórica utilizada

O modelo teórico deste projeto de assessoria é o modelo de Li e Shang (2020), que avalia serviços digitais sob a ótica de geração de valor e da intenção de uso continuado de serviços, auxiliando gestores do STJ a tomar decisões de qualidade. Mais detalhes sobre o referencial teórico e a metodologia de avaliação do Balcão Virtual estão apresentados na pesquisa teórico-empírica (Capítulo 2).

Este trabalho é bastante original, porque, na literatura científica brasileira, até o ano de 2024, não foram encontrados estudos similares. Isso se deve a dois fatores principais: existência de dados de qualidade, estruturados e detalhados dos atendimentos e a facilidade de acessar esses dados devido a um projeto do mestrado profissional patrocinado pelo STJ e executado por um servidor da instituição. É válido destacar que os dados primários têm informações de identificação pessoal e, dificilmente, o STJ poderia disponibilizá-los a terceiros sem um enorme esforço de anonimização e consentimento dos usuários⁴⁰.

3.4 Relevância do produto

Este projeto de assessoria é bastante complexo, pois a extração de dados e conversão em *insights* gerenciais para o STJ dependem de componentes de software feitos sob medida e interações com sistemas distintos, além de perspicácia gerencial e experiência em gestão de mudanças — é necessário transformar estatísticas em argumentos sólidos para tomada de decisão, considerando fatores humanos, culturais e organizacionais. Ademais, o desenvolvimento do produto demandou tempo e esforços consideráveis em análises exploratórias que, às vezes, não produzem *insights* interessantes e, outras vezes, apenas confirmam o esperado.

Este produto pode contribuir para a melhoria contínua do Balcão Virtual e, dada a visibilidade e liderança do STJ no Poder Judiciário brasileiro, tem o potencial de ajudar outros tribunais a melhorarem a prestação de seus serviços para os jurisdicionados. Além disso, a abordagem apresentada de “modelo como mapa” pode ser usada para qualquer serviço público digital, podendo ser estendida ao Governo brasileiro em geral.

Por fim, espera-se que este produto viabilize a adoção do Balcão Virtual em mais áreas do Tribunal, aumentando a qualidade e variedade dos serviços oferecidos aos jurisdicionados e melhorando a imagem do STJ perante a sociedade brasileira.

⁴⁰ Anonimização não é trivial, mesmo substituído identificadores pessoais (como nome, OAB etc.) por códigos. Se vários processos estão associados a um mesmo código, seria possível inferir a identidade de quem fez o chamado cruzando os nomes das partes com os identificadores anonimizados.

3.5 Relatório técnico conclusivo de Assessoria



Universidade de Brasília

Mestrado Profissional em Governança e Inovação em Políticas Públicas

Paulo Tibério Muradas Bulhões

**Relatório Técnico Conclusivo de Assessoria para os gestores do Balcão
Virtual**

Brasília

2025

RESUMO

Este é o relatório dos resultados do projeto de assessoria aos gestores do balcão virtual, desenvolvido como parte de um projeto de pesquisa do Mestrado Profissional em Governança e Inovação em Políticas Públicas, patrocinado pelo Superior Tribunal de Justiça. O relatório tem uma breve introdução sobre a avaliação do balcão virtual como criador de valor público. Na seção Recomendações são apresentadas dez recomendações de melhoria para o balcão virtual e outros canais de atendimento do STJ. As recomendações estão ordenadas por relevância. Cada recomendação é acompanhada por elementos de tomada de decisão: o porquê da recomendação, uma sugestão de implementação e as evidências que a basearam, a fim de facilitar o processo de decisão quanto a implementá-la ou não. Por fim, tem-se a conclusão que sugere a continuidade deste tipo de estudo.

SUMÁRIO EXECUTIVO

O Balcão Virtual do STJ representa um bom uso de recursos públicos, pois gera valor público para a Corte. Esse foi o resultado do estudo apresentado no Capítulo 2, no qual se aplicou um modelo de avaliação de serviços digitais governamentais aos registros de atendimento e ao questionário de controle de qualidade do Balcão Virtual, à luz do estado da arte de na teoria de governança e políticas públicas. Os principais fatores que demonstraram a criação de valor público foram:

- **Intenção de uso continuado:** os usuários estão satisfeitos com o serviço do Balcão Virtual e o recomendam a outros. 98,30% dos usuários atendidos afirmam que recomendariam o serviço para outros usuários. O uso desse serviço tem uma tendência de crescimento, conforme demonstram os dados de atendimento: 2.862 atendimentos em 2021; 8.099 em 2022; e 10.697 em 2023. Por fim, 37,47% dos atendimentos foram para usuários recorrentes, evidenciando o uso contínuo do serviço;
- **Atendimento de requisitos valorizados pelos usuários:** os usuários consideram que vale a pena o tempo e o esforço para usar o Balcão Virtual. Salienta-se a eficiência do serviço (97,34% indicam que o tempo de atendimento foi adequado e 95,40% acham o Balcão Virtual fácil de usar); e
- **Qualidade do serviço:** os usuários percebem o Balcão Virtual como um serviço de alta qualidade com base em seis atributos presentes: confiabilidade, acessibilidade, qualidade da informação, capacidade, interatividade e responsividade.

Ao longo da execução deste estudo sobre criação de valor público pelo Balcão, foram identificadas dez oportunidades de melhoria. As três principais são:

Recomendações principais:

1. **Expansão do Balcão Virtual para Gabinetes,** visto que esse é o atendimento mais procurado. A procura esperada é de, aproximadamente, dois atendimentos por semana, compatível com a carga de trabalho dos Gabinetes. Assim, sugere-se uma estratégia de adoção gradual, iniciando pelos Gabinetes mais receptivos. Para facilitar a implementação, recomenda-se a parceria com a equipe de atendimento dos Gabinetes, integrando o Balcão Virtual à rotina de trabalho.

2. **Aprimoramento da coleta de dados:** faltam dados quanto ao tempo de espera dos usuários antes do atendimento. É essencial monitorar essa métrica para identificar gargalos e aprimorar a experiência do usuário.

3. Padronização da identificação dos usuários: é necessário sanitizar a identificação dos usuários no momento da chamada (confirmar a identidade de um usuário e registrar a OAB/CPF de maneira uniforme, garantindo maior precisão nos dados e melhor acompanhamento do uso contínuo do serviço).

3.5.1 Introdução

Este estudo versa sobre a identificação de criação de valor público pelo Balcão Virtual do STJ. O conceito de valor público foi introduzido por Moore em 1995, em seu trabalho seminal “*Creating Public Value: Strategic Management in Government*” (Moore, 1997). O autor compara o valor público ao conceito de *shareholder value* (valor para os acionistas, ou donos da empresa). Aqui, há uma diferença fundamental: no caso do setor público, a sociedade assume o papel de “acionista” do Governo. Dessa forma, a gestão pública pode ter uma abordagem mais holística em suas ações, trazendo benefícios para a sociedade em geral, ao invés de se fixar em uma missão específica.

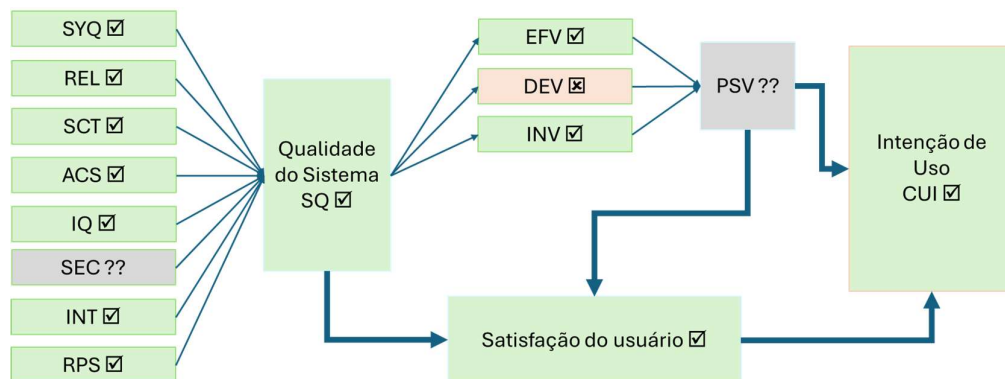
O STJ já aplica essa visão ampliada do valor público em diversas iniciativas, como na desjudicialização. Embora, em uma interpretação mais estrita de sua missão, esse processo pudesse estar fora do escopo de atuação da Corte, ele gera significativos benefícios à sociedade e atende à visão mais ampla de valor público, presente na missão do STJ.

Em 2024, Kearns (2004) aplicou o conceito de valor público ao governo eletrônico, estabelecendo três fontes principais de valor público: serviços de qualidade, resultados e confiança institucional.

O Balcão Virtual é avaliado dentro do contexto de valor público de Kearns (2004). E para avaliar com o rigor científico, foi necessária a utilização de um modelo matemático, definido por Von Neumann (1955). Modelos são um conjunto de variáveis em que o valor de uma variável influencia a outra, isto é, um aumento de uma variável impacta outras variáveis relacionadas.

O modelo adotado no Capítulo 2 foi o de Li e Shang (2020), que relaciona qualidade de serviços digitais (considerando oito atributos de qualidade), três valores percebidos pelo público e dois critérios de sucesso (satisfação com o serviço e intenção de uso continuado). Em seguida, foram buscadas evidências da existência dos elementos desse modelo nos dados disponíveis do Balcão Virtual. O processo detalhado da análise de dados e as evidências coletadas estão descritos no Capítulo 2, com resultado esquematizado na Figura 7. Os atributos ou as variáveis presentes estão em verde e com o símbolo do *checkmark* (☑); os que não se aplicam estão em vermelho com o símbolo da cruz (☒); e os em que não se há evidências diretas suficientes estão em cinza, marcados com pontos de interrogação (?).

Figura 7 – Modelo de Li e Shang (2020)



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Como foram encontradas evidências dos dois critérios de sucesso, de dois valores percebidos pelo público e de sete atributos de qualidade, pode-se concluir que o Balcão Virtual cria valor público para o STJ.

No início do estudo, foram realizadas reuniões com os gestores responsáveis pelo Balcão Virtual da Coordenadoria de Atendimento e Protocolo Judicial (CATJ). O objetivo dessas reuniões foi levantar a disponibilidade de dados que viabilizariam esta pesquisa. Durante as conversas, os gestores manifestaram interesse em saber se haveria evidências que apoiassem a expansão do Balcão Virtual para os Gabinetes, visto que havia demanda por esses atendimentos por parte dos usuários.

No processo de análise dos dados (descrito detalhadamente no Capítulo 2), foram encontradas evidências concretas que apoiam a adoção do Balcão Virtual nos Gabinetes. Além disso, foi possível elaborar uma estimativa de potencial demanda desse serviço, contribuindo para diminuir as incertezas e os riscos de uma possível expansão.

Essa recomendação é a primeira apresentada na seção a seguir. Adicionalmente, nove outras oportunidades de melhoria foram identificadas e estão compiladas na próxima seção.

3.5.2 Recomendações

Nesta seção, são apresentadas as recomendações de melhorias para o Balcão Virtual e demais canais de atendimento do STJ, organizadas por ordem de relevância. Para maior clareza e praticidade, cada recomendação inclui: um título breve, a principal razão para a recomendação, uma estratégia de adoção e evidências completas, que ficaram para o fim, por serem extensas.

3.5.2.1 Recomendação 1: expandir o Balcão Virtual para os Gabinetes

3.5.2.1.1 Justificativa

É o atendimento mais demandado pelos usuários do Balcão Virtual.

3.5.2.1.2 Estratégia de adoção

A evidência indica que os jurisdicionados têm interesse em interagir com os Gabinetes por meio do Balcão Virtual. Atualmente, quando o usuário tenta falar com o Gabinete pelo Balcão Virtual, é aconselhado a entrar em contato por telefone. Isto é, o Gabinete já é contatado de qualquer forma, mas a disponibilidade do Balcão Virtual nos Gabinetes tornaria esse processo mais ágil e eficiente.

Por ser uma videoconferência, o Balcão Virtual tem mais densidade de informação do que uma ligação telefônica: é possível compartilhar telas e perceber a linguagem corporal dos atendentes, o que reforça o sentimento de acolhimento do jurisdicionado.

As evidências apontam para uma demanda amena por Gabinete (menos de dois atendimentos por semana, em média). Esse dado é relevante para convencimento dos Gabinetes em adotar esse canal de atendimento, porque, apesar da pressão interna e da resistência a mudanças que não impactam diretamente na produtividade, essa iniciativa pode melhorar a imagem do Tribunal.

A estratégia de adoção do Balcão Virtual é sugerida nos passos a seguir e, ao fim de cada um, quando possível, está indicado um ou mais artigos da *Harvard Business Review* ou de capítulo de livro disponível na biblioteca do STJ relacionado ao tópico.

1. Identificar Gabinetes mais receptivos para um projeto-piloto: analisar as peculiaridades de cada um e escolher aquele mais aberto à inovação e com demanda moderada. Para convencer um Gabinete a implantar o Balcão Virtual, recomenda-se salientar que a mudança que o Balcão Virtual representa é pequena, visto que esse tipo de atendimento é similar ao atendimento telefônico já existente e traz benefício alinhados aos valores (acessibilidade e inovação), à visão (justiça moderna) e aos objetivos (“Fortalecer a imagem e a transparência

institucional” e “Intensificar o uso de tecnologias da informação”) do STJ⁴¹. (Leitura sugerida [\(Change Management in Government\)](#)).

2. Estudar o fluxo de trabalho do Gabinete piloto: é importante ajustar a operação do Balcão Virtual sem interferir nos demais processos do Gabinete. Uma atenção especial deve ser dada ao processo de atendimento telefônico, pois, sendo o Balcão Virtual uma extensão dele, deve seguir um modo de operação similar. (Leituras sugeridas: [An Agile Approach to Change Management](#) e [Comportamento Organizacional, capítulo 15, Mudança Organizacional](#)).

3. Engajar a chefia do Gabinete: ressaltar para o chefe de Gabinete que o Balcão Virtual é similar ao atendimento que os Gabinetes já proveem e, assim sendo, não é uma mudança de risco elevado. Quaisquer objeções devem ser abordadas atentamente para tranquilizar os participantes do processo de implantação do Balcão Virtual. (Leituras sugeridas: [An Agile Approach to Change Management](#) e [Choosing Strategies for Change](#)).

4. Capacitar os atendentes dos Gabinetes: propor um modelo de atendimento confortável, com o objetivo de reduzir a resistência à adoção do Balcão Virtual. O apoio dos atendentes que farão a linha de frente do Balcão Virtual é fundamental para o sucesso da implantação. (Leitura sugerida: [Change Management in Government](#)).

5. Garantir privacidade no atendimento: pode ser necessário que os atendentes do Balcão virtual trabalhem em um espaço fisicamente reservado do Gabinete. Sugere-se o uso de *headsets* e evitar os microfones embutidos em câmeras para que conversas no ambiente do Gabinete não sejam captadas nos atendimentos do Balcão Virtual.

6. Usar a experiência do projeto-piloto para refinar a adoção em outros Gabinetes: ter um caso de sucesso facilita a expansão e reduz resistências. Além disso, pode ser necessário fazer processos de atendimento específicos para cada Gabinete (ex.: exigir agendamento prévio). Nesses casos, a CATJ pode orientar os usuários quanto ao modelo de atendimento utilizado e fazer o agendamento com data acordada pelo atendente do Balcão Virtual do Gabinete. (Leitura sugerida: [Break Down Change Management into Small Steps](#)).

7. Implantar o Balcão Virtual gradualmente nos outros Gabinetes. (Leitura sugerida: [Break Down Change Management into Small Steps](#)).

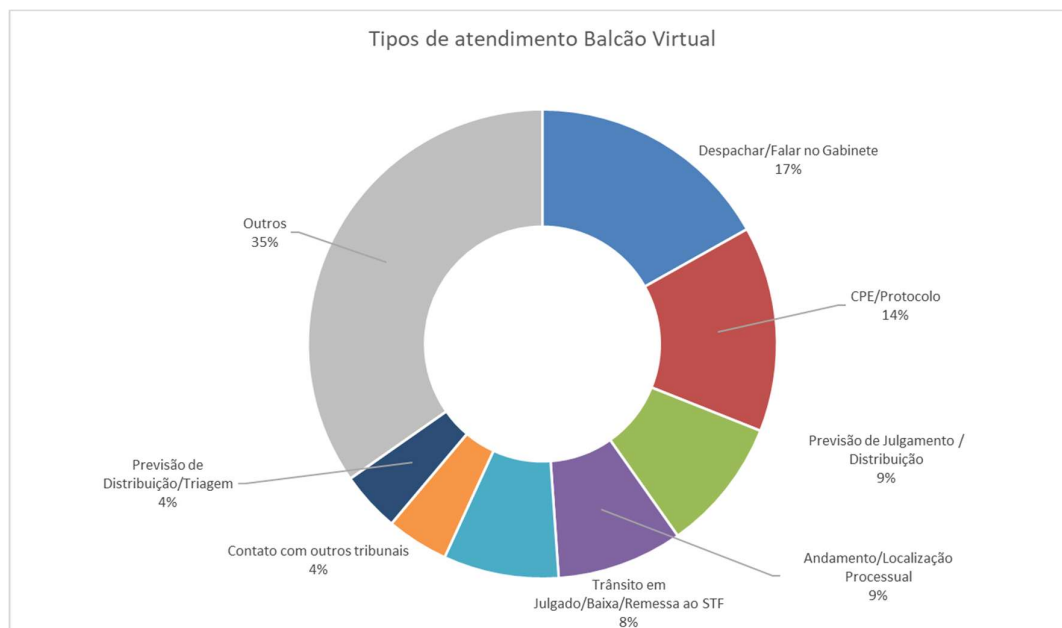
3.5.2.1.3 Evidência de suporte à recomendação

A demanda de atendimentos do tipo “Despachar/Falar no Gabinete” é a mais popular no Balcão Virtual — com 4.389 atendimentos (16,86% do total de atendimentos, arredondado para 17% no Gráfico 13) — e apoia a intenção de expandir o Balcão Virtual para os Gabinetes do

⁴¹ (Superior Tribunal Federal, 2021, p.15, 17, 21 e 23).

STJ. Atualmente, só há Balcão Virtual no Gabinete da Presidência do Tribunal. Ao tentar falar com um Gabinete que não tenha Balcão Virtual, o usuário recebe o número do telefone do Gabinete para um contato telefônico posterior.

Gráfico 13 – Tipos de atendimento do Balcão Virtual



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

O Gráfico 14 apresenta a demanda por este tipo de atendimento ao longo do tempo. A linha vermelha é a média móvel de 30 dias e demonstra uma clara tendência de aumento até abril de 2023 e, depois, parece haver uma estabilização. A demanda continuada é muito importante porque está diretamente relacionada à variável dependente do modelo — intenção de uso continuado (CUI) —, que é o principal indicador de valor público.

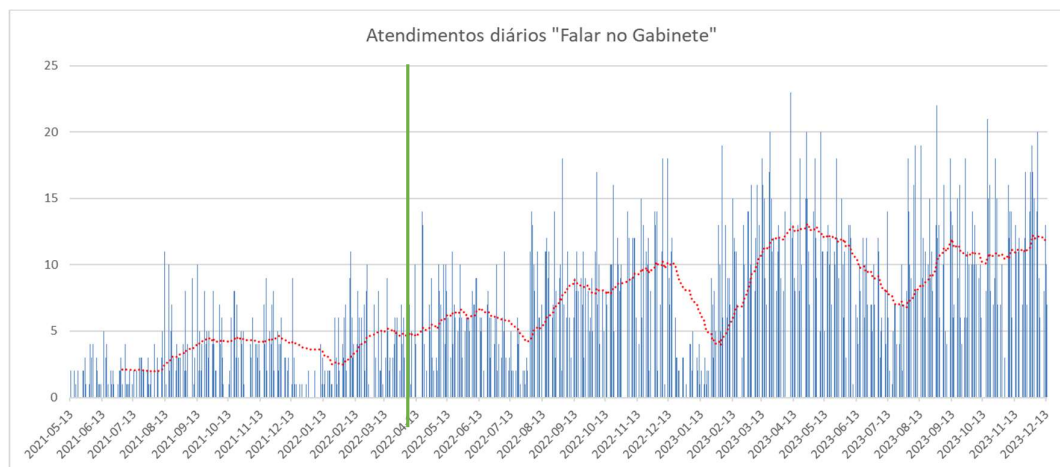
É importante observar que, apesar do fim da pandemia e do retorno ao presencial⁴² em 1º de abril de 2022 (assinalado com a linha verde), a demanda do Balcão Virtual continua crescendo, demonstrando que a necessidade de criação do Balcão Virtual não se devia à impossibilidade de atendimento presencial.

Quanto ao nivelamento da linha de atendimentos no fim de 2023, é preciso ter mais dados para estabelecer se esse nivelamento é permanente ou temporário. Existem possíveis cenários, como: a demanda simplesmente atingir seu máximo, ou usuários que gostariam de

⁴² A Resolução STJ/GP n.º 9, de 25 de março de 2022, estabeleceu o retorno do trabalho presencial no âmbito do Superior Tribunal de Justiça. Disponível em: <https://bdjur.stj.jus.br/items/d82fc044-347e-459e-9de0-d7f18a853f88>. Acesso em: 3 mar. 2025.

entrar em contato com os Gabinetes sendo informados de que ainda não há Balcão nos Gabinetes, desistindo de tentar novamente e usando outros meios, como ligação.

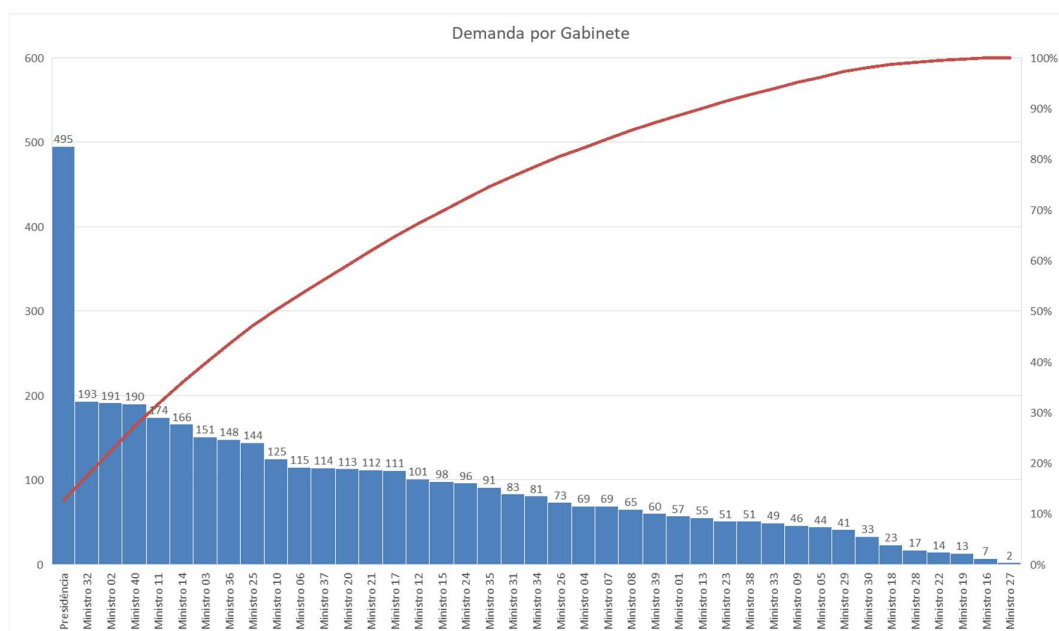
Gráfico 14 – Atendimentos diários de "Falar no Gabinete"



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Dos chamados em que foi possível identificar qual Gabinete o usuário pretendia contactar, foi elaborado o Gráfico 15, que apresenta a demanda por Gabinete. É possível verificar que o Gabinete mais demandado é o da Presidência. No entanto, não é possível afirmar se isso ocorre devido à natureza de suas funções ou, simplesmente, por ser o único Gabinete que já atende pelo Balcão Virtual, ao contrário dos outros, que atendem somente por telefone.

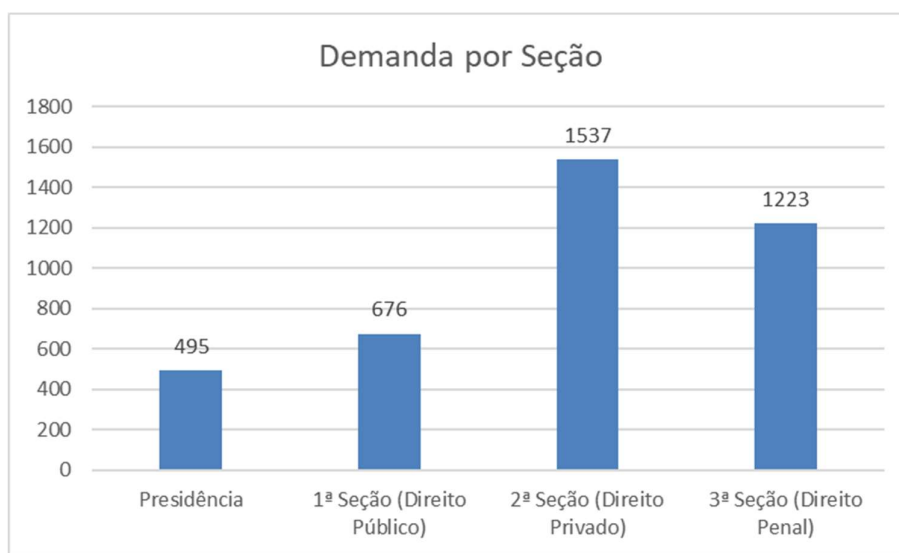
Gráfico 15 – Demanda por Gabinete (anonimizado)



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

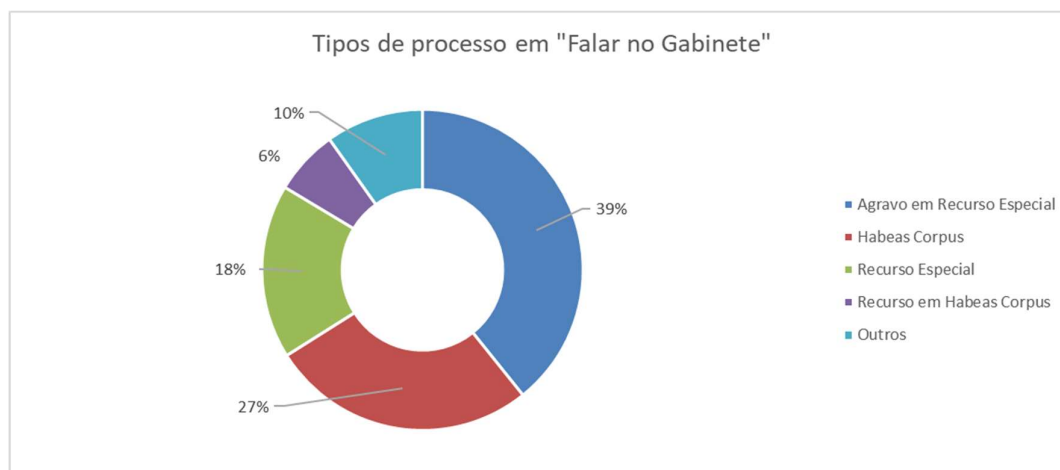
No Gráfico 16, temos a demanda agregada por seção. A maioria das demandas é para a área de Direito Privado e Direito Penal. No Gráfico 17, temos os tipos de processos referenciados nos atendimentos do tipo “Falar no Gabinete” — seguindo a demanda do STJ, a maior parte dos processos são Agravos em Recurso Especial (AREsp), Recurso Especial (REsp) e processos relacionados a *Habeas Corpus* (*Habeas Corpus* e Recurso em *Habeas Corpus*).

Gráfico 16 – Demanda por seção



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 17 – Tipos de processo



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

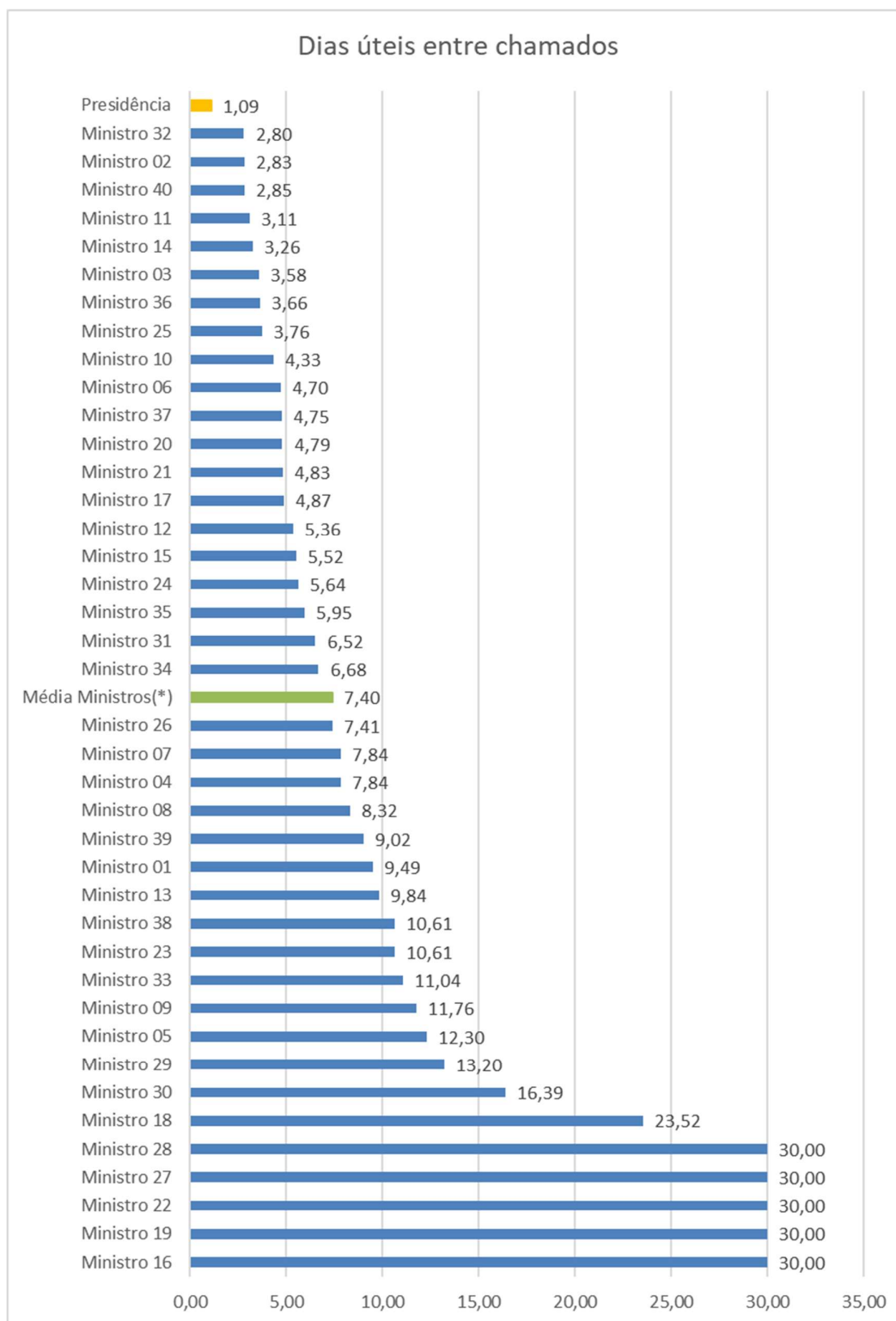
Já a Tabela 15 apresenta os 10 maiores tribunais de origem dos chamados do Balcão Virtual. Nesse aspecto, não há surpresas: a demanda segue o padrão esperado, com maior volume oriundo dos tribunais dos estados mais populosos. No Apêndice E é possível conferir uma lista mais detalhada, com os 40 tribunais de origem que mais figuram nos processos referenciados no atendimento do Balcão Virtual.

Tabela 15 – Tribunais de origem

Tribunal de origem	Contagem	Percentual
TRIBUNAL DE JUSTIÇA DO ESTADO DO RIO DE JANEIRO	2611	15,60%
TRIBUNAL DE JUSTIÇA DO ESTADO DE SÃO PAULO – CONSELHEIRO FURTADO – PÁTIO DO COLÉGIO	1720	10,28%
SUPERIOR TRIBUNAL DE JUSTIÇA	1272	7,60%
TRIBUNAL DE JUSTIÇA DO DISTRITO FEDERAL E DOS TERRITÓRIOS	937	5,60%
TRIBUNAL DE JUSTIÇA DO ESTADO DE SÃO PAULO – RUA DA GLÓRIA	882	5,27%
TRIBUNAL DE JUSTIÇA DO ESTADO DE MINAS GERAIS	841	5,02%
TRIBUNAL DE JUSTIÇA DO ESTADO DA BAHIA	595	3,55%
TRIBUNAL REGIONAL FEDERAL DA 2ª REGIÃO	503	3,01%
TRIBUNAL DE JUSTIÇA DO ESTADO DO RIO GRANDE DO SUL	499	2,98%
TRIBUNAL REGIONAL FEDERAL DA 5ª REGIÃO	480	2,87%
Outros (116 tribunais)	6397	38,22%

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

A seguir, examina-se a demanda para o Balcão Virtual em termos de dias entre chamados. Esse tipo de análise é essencial para auxiliar na tomada de decisão quanto ao esforço e à quantidade de pessoal necessário para um atendimento de qualidade. É possível perceber que o Gabinete mais demandado, o da Presidência, tem, em média, um chamado por dia (Gráfico 18). Dos Gabinetes dos Ministros, o mais demandado tem uma média de 2,80 dias entre chamados, o que representa um pouco menos de duas vezes por semana. A “Média Ministros”, de 7,40 dias, é o intervalo médio de todos os gabinetes de ministro, excluindo a presidência e os com intervalo de 30 dias, para não enviesar a média. Isso significa que não é uma demanda intensa, e pode ajudar na argumentação de adoção do Balcão Virtual nos Gabinetes. Além disso, devido ao modo de operação atual, o Gabinete ainda seria chamado por telefone pelo usuário, logo a implantação do Balcão Virtual não causaria transtornos significativos no processo de trabalho do Gabinete.

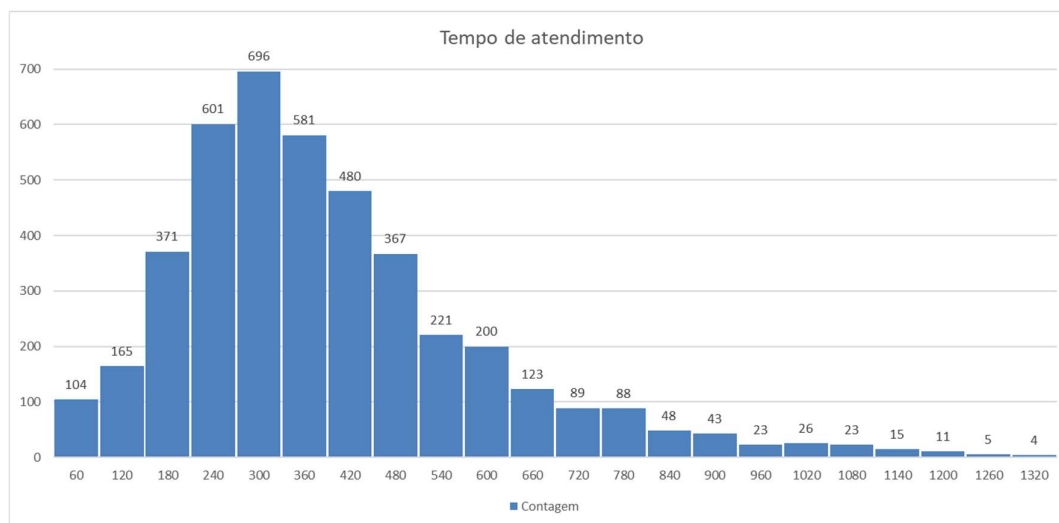
Gráfico 18 – Tempo médio entre atendimentos por Gabinete (anonimizados)

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Quanto aos tempos de atendimento para chamadas de Gabinete, são 4.284 registros — máximo 1.320 segundos (22 minutos); média de 390 segundos (6,5 minutos); e desvio padrão

de 205,96 segundos (3,14 minutos). O histograma da frequência dos tempos de atendimento nos chamados relativos aos Gabinetes (Gráfico 19) demonstra que a grande maioria dura até 8 minutos (75% dos atendimentos), ou seja, mesmo se o jurisdicionado estiver em Brasília, o tempo de atendimento é muito menor que o tempo de deslocamento até o STJ.

Gráfico 19 – Tempo de atendimento nos Gabinetes



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Entende-se que o tempo de atendimento não é longo, o que reforça o argumento de que a introdução do Balcão Virtual dos Gabinetes não deve causar perdas de produtividade. Segundo as evidências encontradas, a demanda atual é de atendimentos de 5 minutos, duas vezes por semana. No entanto, apesar de parecer pouco, algumas considerações precisam ser feitas: com a adoção do Balcão Virtual nos Gabinetes, provavelmente, haverá um aumento de demanda, pois a demanda corrente irá para outros canais de atendimento. Além disso, o tempo de preparo do atendimento é bem superior ao tempo de atendimento em si. Também, é importante evitar a inversão de prioridade (quando um processo é tratado mais rapidamente porque há um atendimento do Balcão Virtual pendente, em vez de se usar a ordem de prioridade determinada pelas políticas do Tribunal).

Essas evidências são importantes para argumentar pela adoção do Balcão Virtual nos Gabinetes, porém não são suficientes. Os Ministros são fortemente incentivados a julgar os processos no menor tempo possível, o que pode gerar resistência a qualquer modificação nos processos de trabalho que não traga redução do tempo de julgamento dos processos, e a adoção do Balcão Virtual é vista como uma distração. Além disso, não há experiência prévia de Balcão Virtual em nenhum Gabinete de Ministro, o que aumenta a incerteza sobre seus impactos. Sugere-se aos gestores do Balcão uma abordagem de caso de sucesso, trabalhando com um

Gabinete para fazer um projeto-piloto e criar experiência de funcionamento do Balcão Virtual no Gabinete, com o intuito de diminuir a resistência em outros Gabinetes e estabelecer as melhores práticas, como descrito em “Estratégia de adoção” nesta recomendação.

3.5.2.2 Recomendação 2: incluir questão de percepção de valor público do Balcão Virtual no questionário de controle de qualidade da Ouvidoria

3.5.2.2.1 Justificativa

Essa é uma das variáveis que falta na avaliação de valor do Balcão Virtual no modelo de Li e Shang (2020). Obter uma evidência direta dessa variável fortalece a conclusão de que o Balcão Virtual cria valor público.

3.5.2.2.2 Estratégia de adoção

No modelo de Li e Shang (2020), são utilizadas algumas questões que podem ser adaptadas ao contexto do Balcão Virtual: “O serviço que eu recebo do Balcão Virtual vale o tempo e o esforço que eu fiz para acessá-lo?”, “Acredito que o Balcão Virtual provê valor público?” e “O valor derivado do Balcão Virtual valeu o tempo, esforço e investimento feitos pelo Governo?”. Incorporar as três seria o ideal, mas o conceito de valor público pode não ser de conhecimento dos usuários do STJ. Além disso, o questionário pode ser eficaz por ser curto. Por esses motivos, a recomendação é a de se incluir pelo menos uma questão.

3.5.2.2.3 Evidência de suporte à recomendação

No modelo de Li e Shang (2020), a variável tem o nome PSV, do inglês *perceived service value*, ou valor percebido do serviço. Essa variável serve de intermediária entre as avaliações de valores interessantes ao público que, no modelo, são as variáveis relativas à eficiência (EFV), inclusividade (INV) e democracia (DEV), e a variável de intenção de uso continuado (CUI) que sumariza o valor total percebido pelos usuários.

3.5.2.3 Recomendação 3: incluir questão sobre percepção de segurança da informação no questionário de controle de qualidade da Ouvidoria

3.5.2.3.1 Justificativa

Essa é uma das variáveis que falta na avaliação de valor do Balcão Virtual no modelo de Li e Shang (2020), e o atual questionário da Ouvidoria não avalia a percepção de segurança

da informação. O STJ tem empenhado grandes esforços em melhorar a segurança da informação, por isso é tão importante mensurar se eles são percebidos por seus usuários.

3.5.2.3.2 Estratégia de adoção

Sugere-se incluir a questão “Acredito que meus dados são seguramente enviados ao STJ e manipulados para que *hackers* não consigam roubar minhas informações”.

3.5.2.3.3 Evidência de suporte à recomendação

Apesar de não haver relatos de problemas de segurança nem incidentes de segurança pela Coordenadoria de Segurança da Informação e Defesa Cibernética (CSID) no período analisado, o modelo de Li e Shang (2020) (disponíveis Anexo A) apresenta três perguntas que versam sobre a percepção dos usuários de que os dados estarão seguros e que podem ser adaptadas ao contexto do STJ e do Balcão Virtual. A pesquisa de satisfação do Balcão Virtual também não tem questões relativas à percepção de segurança, logo não é possível avaliar como os usuários se sentem. Por isso, sugere-se a melhoria do questionário do controle de qualidade.

3.5.2.4 Recomendação 4: aplicar questionário de controle de qualidade para outros canais

3.5.2.4.1 Justificativa

O questionário de controle de qualidade da Ouvidoria é um instrumento fundamental para gestão do serviço e identificação de fragilidades nos atendimentos. Ele é eficaz por medir vários aspectos da prestação de serviço com um número pequeno de perguntas e fornecer informações acionáveis para os gestores corrigirem eventuais problemas. Ademais, foi fundamental para a identificação de criação de valor público do Balcão Virtual neste estudo, e a mesma abordagem pode ser aplicada aos demais canais de atendimento.

3.5.2.4.2 Estratégia de adoção

Sugere-se adaptar o questionário aos demais canais de atendimento, ajustando as perguntas conforme a especificidade de cada um. Por exemplo:

- Telefone: incluir perguntas sobre a qualidade da ligação; e
- E-mail: avaliar o tempo de resposta e a compreensão da mensagem.

Na Tabela 16, estão listadas as perguntas do questionário, sendo que as destacadas de cinza são específicas do Balcão Virtual e devem ser ajustadas para outros serviços.

Tabela 16 – Perguntas do questionário da Ouvidoria

Questão
Facilidade em acessar o Balcão Virtual do STJ
Ferramenta de videochamada (Zoom)
Profissionalismo (presteza e cordialidade) dos consultores
Qualidade das informações prestadas
Duração do atendimento
Qual a probabilidade de você recomendar este serviço a um amigo ou colega de profissão?
SATISFAÇÃO GERAL - De uma forma geral, como você avaliaria o serviço "Balcão Virtual"?
SUGESTÕES E CRÍTICAS - Deixe aqui suas sugestões ou críticas sobre o serviço "Balcão Virtual".
Faixa etária
Relacionamento com o STJ
Escolaridade
Local de Residência

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Quanto ao envio do formulário, no caso do telefone, sugere-se utilizar o sistema de registro dos atendimentos e enviar o formulário para o e-mail de cadastrado (tendo o usuário a opção de não receber o e-mail). Já no caso do e-mail, basta incluir o link para o questionário na resposta. Finalmente, no caso do Espaço do Advogado, pode-se colocar um computador à disposição para que o usuário responda o questionário ao final do atendimento, disponibilizar um QR Code para respostas em dispositivos móveis ou mandar o link por e-mail.

3.5.2.4.3 Evidência de suporte à recomendação

As respostas do questionário da Ouvidoria podem destacar a excelência dos outros canais de atendimento do STJ e ser instrumentais para correções de eventuais fragilidades no atendimento.

Foi realizada uma breve comparação entre o Balcão Virtual e o atendimento telefônico, visto que este último é o serviço mais usado no STJ. No entanto, a comparação foi limitada por três fatores: o serviço telefônico não é o foco deste trabalho; os registros de atendimento disponíveis têm uma lacuna de quase um ano (321 dias); e não há pesquisa de satisfação para atendimento telefônico. Dito isso, os dados de atendimento do Balcão Virtual foram selecionados para coincidir com os dados disponíveis do atendimento telefônico (a comparação foi feita em dias que havia dados tanto do Balcão Virtual quanto do atendimento telefônico) e os resultados estão na Tabela 17.

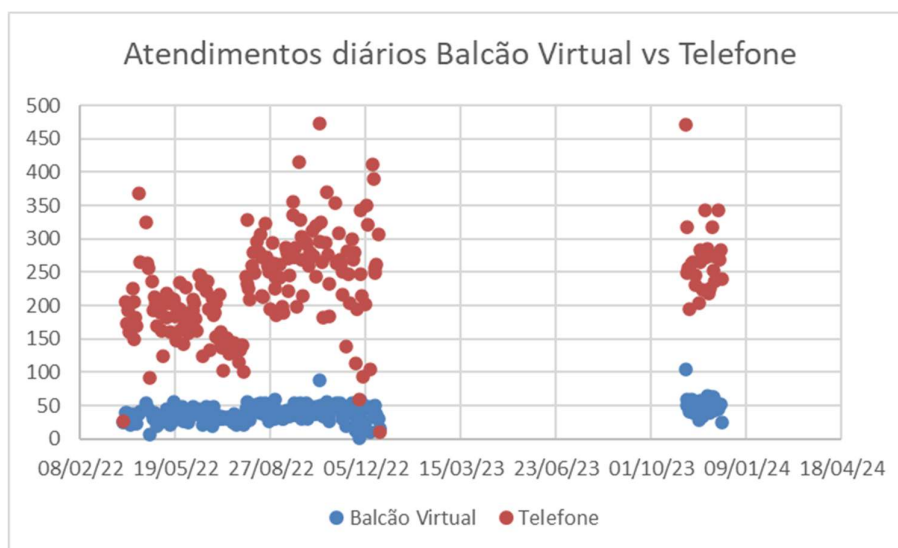
Tabela 17 – Atendimentos diários (Balcão Virtual vs Telefone)

Serviço	Total de atendimentos	Média	Desvio padrão	Mínimo	Máximo
Balcão Virtual	8.108	39.17	12.46	2	105
Telefone	47.440	229.18	72.72	10	473

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Como se pode observar, o atendimento por telefone tem um volume muito maior do que o Balcão Virtual, na ordem de 5,85 vezes mais atendimentos por dia. No Gráfico 20, estão apresentados os atendimentos tanto do Balcão Virtual como por telefone, ficando evidente a lacuna de 321 dias nos registros de atendimento por telefone.

Gráfico 20 – Atendimentos diários (Balcão Virtual vs Telefone)

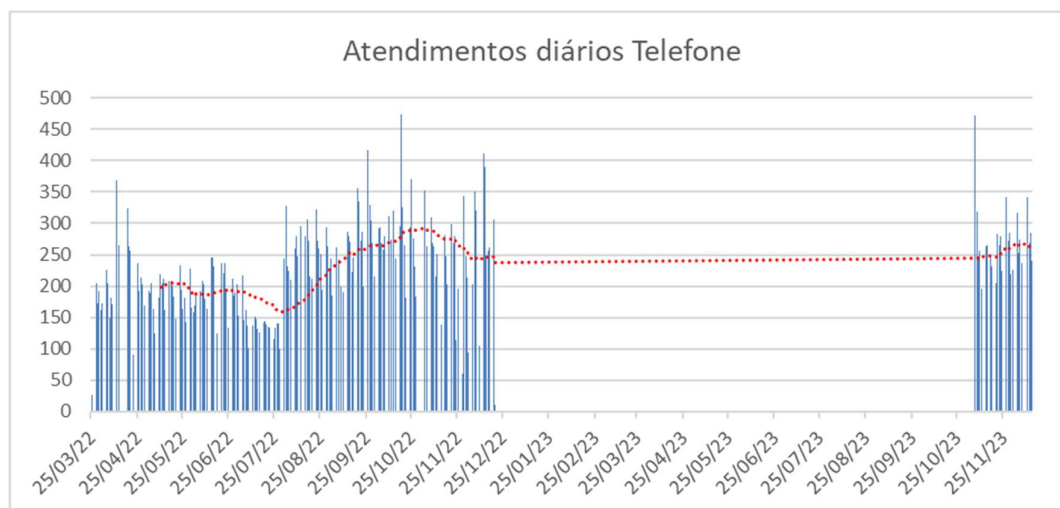


Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

No Gráfico 21, são apresentados os atendimentos por telefone com linha de tendência de média móvel. Nele, pode-se observar um aumento de chamados na segunda metade de 2022,

mas não há dados suficientes para determinar se há uma tendência de aumento constante ou se o aumento do segundo semestre de 2022 é um efeito sazonal.

Gráfico 21 – Atendimentos diários (telefone)



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

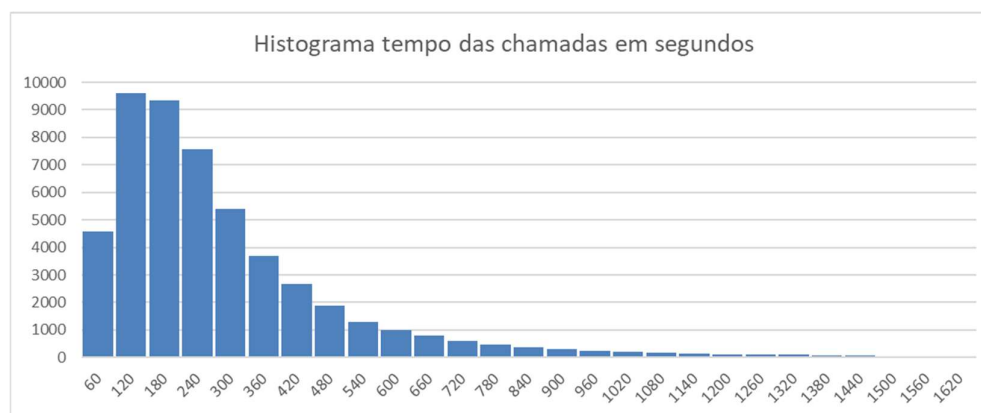
Quanto aos chamados recorrentes, são 51.979 chamados no total, dos quais 23.301 são dos mesmos usuários, o que equivale a um percentual de 44,82% de chamados recorrentes.

As estatísticas do quesito de tempo médio do atendimento telefônico estão na Tabela 18 e, no Gráfico 22, consta o histograma de duração do tempo de atendimento.

Tabela 18 – Estatísticas de tempo de atendimento telefônico

Tempo mínimo	Tempo máximo	Média	Desvio padrão	Contagem de atendimentos
4s	1.561s	262.45s	223.48s	50.823

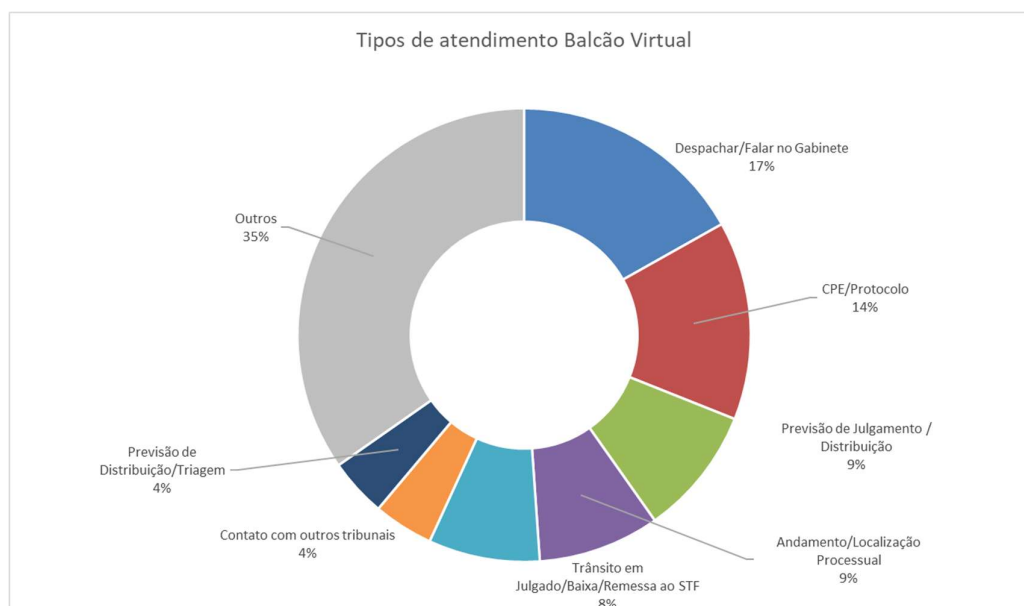
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 22 – Histograma do tempo de atendimento em segundos

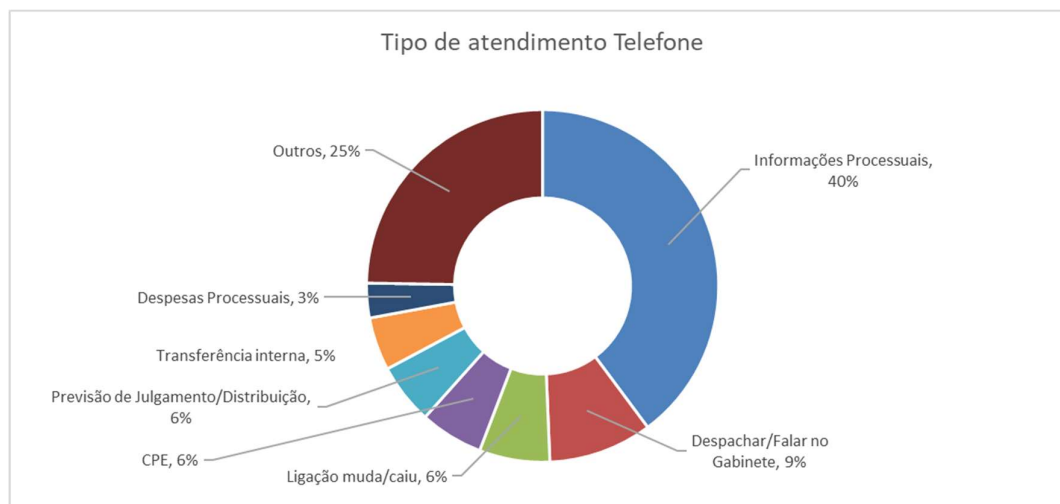
Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Comparando o Balcão Virtual ao atendimento telefônico, temos os Gráficos 23 e 24. Observa-se que o atendimento mais comum no Balcão Virtual é “Despachar/Falar no Gabinete”, com 17%. Já no atendimento telefônico, o mais comum é “Informações Processuais”, com 40% dos atendimentos. Comparando-se os atendimentos mais populares nos dois canais de atendimento, “Informações Processuais” aparece em 17º lugar, com apenas 1,52% do total de chamados do Balcão Virtual, e “Despachar/Falar no Gabinete” aparece em 2º lugar, com 9,5% dos chamados telefônicos.

Pode-se concluir que os usuários preferem obter informações processuais principalmente por telefone e que ambos os canais são usados para “Despachar/Falar no Gabinete”, mas, no Balcão Virtual, há uma preferência maior para este tipo de atendimento.

Gráfico 23 – Tipos de atendimento do Balcão Virtual

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 24 – Tipos de atendimento (telefone)

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Assim, é possível concluir que o atendimento telefônico tem mais chamadas recorrentes (o que se relaciona à variável dependente do modelo de Li e Shang (2020), intenção de uso continuado – CUI) e um tempo de atendimento mais curto (o que se associa ao atributo de valor eficiência – EFV). Isso indica uma possível geração de valor público superior ao Balcão Virtual, mas a ausência da pesquisa de satisfação da Ouvidoria para o atendimento telefônico impede uma avaliação completa, especialmente quanto à satisfação dos usuários.

3.5.2.5 Recomendação 5: identificar os usuários de maneira uniforme

3.5.2.5.1 Justificativa

A identificação dos usuários é voluntária e não validada, permitindo variações no registro (ex.: nome completo ou com abreviações, erros de grafia em nomes ou nos identificadores, como OAB ou CPF, múltiplos e-mails etc.). Isso dificulta a identificação e as análises holísticas do comportamento dos usuários nos diferentes canais de atendimento do STJ.

3.5.2.5.2 Estratégia de adoção

Sugere-se primeiro validar identificadores públicos como CPF e número da OAB para evitar erros de digitação. Para o número da OAB, é possível consultar o cadastro e preencher automaticamente o nome do advogado. Além disso, pode-se usar os dados já existentes nos registros de atendimento e nas listas de advogados de partes de processo no STJ, compilar em uma lista de usuários com todos os identificadores conhecidos (várias grafias de nome, OAB, CPF, vários endereços de e-mail) e usá-la para associar o atendimento a um identificador único do usuário.

3.5.2.5.3 Evidência de suporte à recomendação

A Tabela 19 mostra um extrato anonimizado das várias identificações usadas por um usuário nos registros de atendimento do Balcão Virtual e telefone. Se o usuário fosse identificado de maneira uniforme, a contagem teria uma única linha com 26 atendimentos.

Tabela 19 – Extrato anonimizado com diferentes identificações para um mesmo usuário

Nome	E-mail	CPF/OAB	Ligações
aaaaaa	Em branco	000.000.000/00	1
		ddd.ddd.ddd/vv	1
		Inválido	2
	não informou	Inválido	1
aaaaaa da ddddd ccccc	não informou.	ddd.ddd.ddd/vv	1
aaaaaa de eeeeeee ccccc	não informou	EENNNN (OAB)	1
aaaaaa bbbbbb	Em branco	ddd.ddd.ddd/vv	1
aaaaaa bbbbbb da ddddd ccccc	nnnnnnnnnnnnnnnn@gmail.com	ddd.ddd.ddd/vv	4
	Em branco	ddd.ddd.ddd/vv	10
	parte sem e-mail	ddd.ddd.ddd/vv	1
aaaaaa bbbbbb ccccc	Em branco	ddd.ddd.ddd/vv	1
aaaaaa bbbbbb da ddddd ccccc	Em branco	ddd.ddd.ddd/vv	1
aaaaaa bbbbbb da ddddd ccccc	Em branco	ddd.ddd.ddd/vv	1

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

3.5.2.6 Recomendação 6: melhorar o registro dos atendimentos nos canais de e-mail, telefone e Espaço do Advogado

3.5.2.6.1 Justificativa

O Balcão Virtual foi o único serviço com registros consistentes, ininterruptos e de boa qualidade. Em contraste, os registros de atendimento dos outros canais apresentaram lacunas de atendimento de vários meses. Além disso, no caso do e-mail e do Espaço do Advogado (atendimento presencial no STJ), houve registros duplicados do mesmo atendimento em diferentes serviços, comprometendo a confiabilidade dos dados.

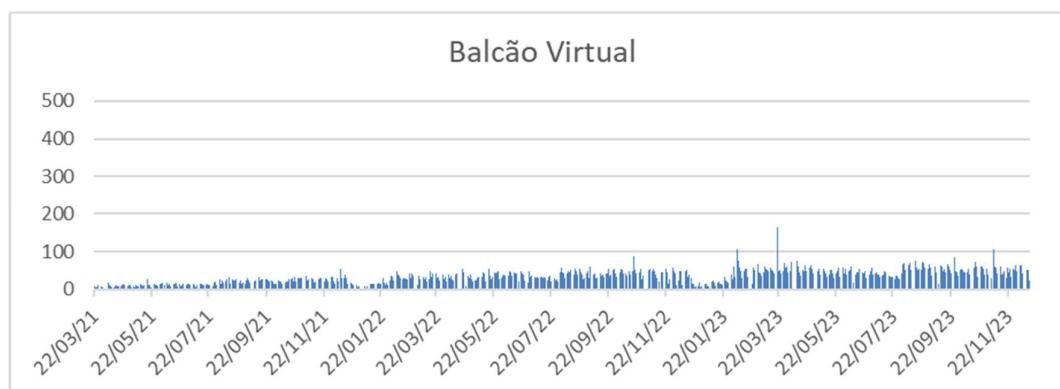
3.5.2.6.2 Estratégia de adoção

Os registros sugerem que um único sistema é utilizado para armazenar as informações do atendimento, o que indica não haver uma dificuldade tecnológica em se ter registros de qualidade. É necessário verificar se a falha ocorreu na extração de dados do banco do sistema de registros ou se os dados não foram registrados corretamente por falha humana. Em ambos os casos, a recomendação é corrigir o processo de trabalho para garantir a integridade e continuidade dos registros nos diversos canais de atendimento.

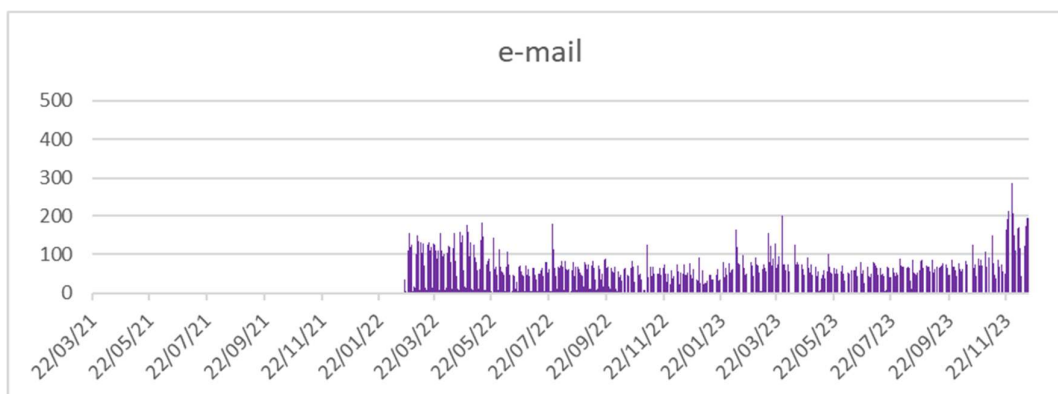
3.5.2.6.3 Evidência de suporte à recomendação

Os Gráficos 25, 26, 27 e 28 mostram as contagens diárias de atendimentos no período analisado, evidenciando as lacunas nos registros dos demais canais. Apenas o Balcão Virtual apresenta dados consistentes. Salienta-se que, com exceção do Espaço do Advogado (presencial), que ficou fechado devido à pandemia de covid-19 até março de 2022, os outros canais de atendimento estavam funcionando plenamente. Assim, as lacunas indicam a falta de registros de atendimento para estudo. Além disso, no caso do e-mail e do Espaço do Advogado, alguns registros aparecem duplicados em ambos os serviços, e há registros de atendimento no Espaço do Advogado antes mesmo da reabertura oficial, em 1º de abril de 2022.

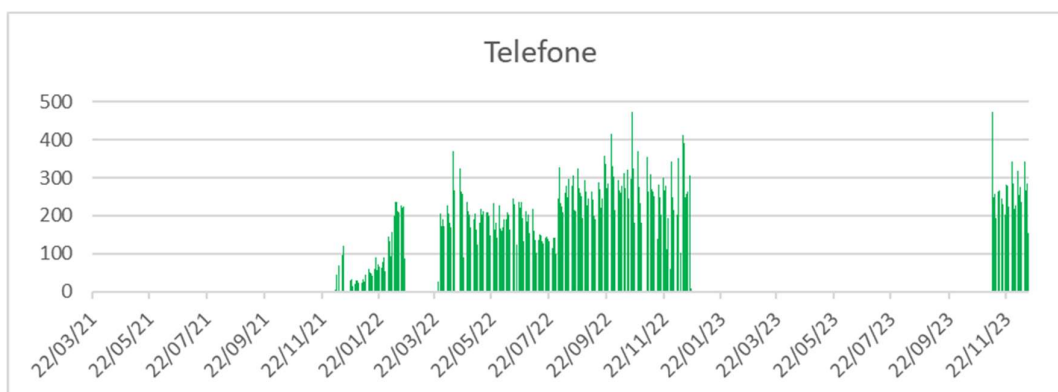
Gráfico 25 – Contagem diária de atendimentos do Balcão Virtual



Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 26 – Contagem diária de atendimentos por e-mail

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 27 – Contagem diária de atendimentos por telefone

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

Gráfico 28 – Contagem diária de atendimentos do Espaço do Advogado

Fonte: Elaborado pelo autor com base nos dados obtidos no Balcão Virtual (2024).

3.5.2.7 Recomendação 7: mensurar o tempo de espera por atendimento

3.5.2.7.1 Justificativa

O tempo de espera nos atendimentos afeta diretamente a experiência do usuário e a satisfação com o serviço. Esperas prolongadas podem levar à desistência e busca por outros canais. Por esses motivos, um indicador de tempo serviria para detectar possíveis problemas de capacidade de atendimento e apoiaria a decisão de contratar mais atendentes caso o tempo de espera aumentasse de maneira persistente.

3.5.2.7.2 Estratégia de adoção

Sugere-se contar o tempo a partir da identificação do usuário. Para isso, é preciso incluir uma coluna “Hora de identificação” no banco de dados para registro dessa informação. Atualmente, o banco de dados registra o início (coluna “Hora de início”) e o fim (coluna “Hora de fim”). Com a coluna adicional sugerida, pode-se medir o tempo de espera pela subtração de “Hora de identificação” – “Hora de início”.

3.5.2.7.3 Evidência de suporte à recomendação

O banco de dados não armazena nenhuma informação sobre o tempo de espera dos usuários. Isso impede os gestores do Balcão Virtual de identificar problemas de espera de forma antecipada e dificulta a tomada de decisões para melhorar a eficiência do atendimento.

3.5.2.8 Recomendação 8: registrar os tempos de indisponibilidade do serviço

3.5.2.8.1 Justificativa

A indisponibilidade do serviço impacta negativamente a satisfação dos usuários, por isso deve ser monitorada e gerenciada. Na Secretaria de Tecnologia da Informação (STI) do STJ, já há indicadores de disponibilidade, como o do portal do STJ, cuja meta é manter os serviços acessíveis 99% do tempo, gerando uma expectativa de que os serviços digitais do STJ estejam quase sempre disponíveis, e isso se aplica também ao Balcão Virtual.

3.5.2.8.2 Estratégia de adoção

Em um primeiro momento, os gestores podem registrar o início e o fim de períodos de indisponibilidade em um banco de dados de informações gerenciais. Uma solução mais avançada e eficaz seria utilizar o novo sistema de gerenciamento de serviços corporativos adquirido pelo Tribunal (Contrato STJ 084/2024). Esse sistema permite o mapeamento dos componentes que compõem um serviço como o Balcão Virtual e alertar os gestores sobre falhas,

antes que os usuários percebam. O sistema está em fase inicial de implantação e deve estar disponível até 2026.

3.5.2.8.3 Evidência de suporte à recomendação

Atualmente, só são registradas conexões interrompidas nos registros de atendimento quando há relatos de problemas de acesso. Como os usuários não conseguem utilizar o serviço durante períodos de indisponibilidade, não há registros automáticos desses eventos. Assim, os gestores só tomam conhecimento da falha por meio de reclamações na Ouvidoria, tornando essencial um sistema que registre e notifique a indisponibilidade de forma proativa.

3.5.2.9 Recomendação 9: melhorar a validação dos registros de atendimentos do Balcão Virtual

3.5.2.9.1 Justificativa

Os registros de atendimento contêm dados inválidos, o que compromete a qualidade das análises e a tomada de decisão. Além disso, a sanitização nem sempre é possível. Por exemplo, se o atendente digitar incorretamente um número de processo, pode ser possível detectar o erro se o formato for inválido. Entretanto, se o número errado tiver o formato válido, mas apontar para um processo diferente do que o referenciado pelo usuário, não há como detectar o erro.

3.5.2.9.2 Estratégia de adoção

Em um primeiro momento, pode-se usar o software de sanitização de dados deste estudo, selecionar os dados inválidos e verificar se no sistema de registro dos atendimentos há algum erro que poderia ocasionar o problema. Para os dados inseridos manualmente, o melhor caminho é automatizar, sempre que for possível, a entrada de dados. No caso de consulta processual, o sistema de registro poderia capturar automaticamente a referência do processo consultado e incluí-la na descrição do atendimento. Essa medida reduziria erros e, ainda, facilitaria o trabalho dos atendentes.

3.5.2.9.3 Evidência de suporte à recomendação

O programa de sanitização dos dados deste estudo rejeitou 297 registros de atendimento baseado apenas nos casos de datas inválidas (ex.: ano de atendimento com data de 2142; e até mesmo data de fim menor que o tempo de início). Quanto aos números de processo, de 17.035 registros, 819 não puderam ser consultados — a maioria devido a erros de digitação ou por estar sob segredo de justiça.

3.5.2.10 Recomendação 10: adaptar e aplicar o questionário do modelo de Li e Shang (2020) nos canais de atendimento do STJ

3.5.2.10.1 Justificativa

O foco do estudo do Capítulo 2 é identificar se o Balcão Virtual cria valor público com as informações disponíveis no STJ. Se for aplicado o questionário do modelo de Li e Shang (2020) adaptado aos canais de atendimento do Tribunal, cobrindo todas as variáveis do modelo, a identificação de criação de valor seria muito mais robusta e serviria para destacar a função de criação de valor público do atendimento do STJ.

3.5.2.10.2 Estratégia de adoção

Sugere-se traduzir e adaptar, transculturalmente, o questionário do modelo de Li e Shang (2020), ajustando as questões ao contexto dos canais de atendimento do STJ. Para aplicar o questionário aos usuários do STJ, pode-se:

1. Fazer um banco de dados dos usuários existentes baseado no registro de atendimento;
 2. Realizar uma análise demográfica e determinar a estratificação dos usuários do STJ;
- e
3. Aplicar o questionário aos diferentes estratos da comunidade de usuários para obter uma visão de como estes percebem o valor público dos serviços de atendimento do STJ.

3.5.2.10.3 Evidência de suporte à recomendação

O estudo do Capítulo 2 apresenta as evidências de criação de valor público do Balcão Virtual, mas não é completo nem definitivo — é um “melhor esforço” baseado em informações pré-existentes. O modelo completo, isto é, com todas as perguntas respondidas, de todas as variáveis de uma amostra estratificada dos usuários do STJ, apresentaria uma resposta à pergunta de pesquisa muito mais robusta. Além disso, contribuiria para o avanço da ciência pela validação do modelo Li e Shang (2020) no ambiente brasileiro.

3.5.3 CONCLUSÃO

Espera-se que as dez recomendações de melhoria apresentadas atendam às necessidades dos gestores do Balcão Virtual, manifestadas nas reuniões iniciais deste projeto, e que contribuam para aprimorar os serviços de atendimento do Tribunal. Com dezenas de novos atendimentos realizados diariamente pelo Balcão Virtual e por outros canais do STJ, as informações aqui apresentadas tendem a se tornar obsoletas com o tempo. Portanto, é essencial revisar periodicamente as análises para garantir que os serviços do STJ estejam sempre atualizados.

O software desenvolvido para gerar as evidências deste projeto está disponível no Apêndice G e pode ser utilizado pelos gestores do Balcão Virtual. Além disso, o autor se coloca à disposição para transferir o conhecimento necessário sobre sua operação e modificação à equipe de gestão.

Vale destacar que tanto o software como a abordagem apresentada no Capítulo 2 e neste projeto de assessoria são aplicáveis aos outros canais de atendimento do STJ, particularmente com relação à sanitização e análise de dados, visto que os registros dos atendimentos têm formato similar entre os diferentes canais de assistência ao jurisdicionado do STJ.

Por fim, a abordagem avaliação de serviços públicos digitais sob a ótica de valor público é importante para a gestão pública, pois mantém o foco das organizações governamentais no benefício à sociedade no sentido mais amplo, contribuindo para o atingimento do objetivo social do Brasil de construir uma sociedade livre, justa e solidária.

3.5.4 REFERÊNCIAS

CLAYTON, Sarah Jensen. *An Agile Approach to Change Management*. **Harvard Business Review**, 2021. Disponível em: <https://hbr.org/2021/01/an-agile-approach-to-change-management>. Acesso em: 8 mar. 2025.

KEARNS, Iain. *Public value and e-government*. London: Institute for Public Policy Research, 2004. Disponível em: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=1959beb80056dbb823ae035303361b6285e4a480>. Acesso em: 8 mar. 2025.

KAVANAUGH, Jeff, TARAFDAR, Rafee. *Break Down Change Management into Small Steps*. **Harvard Business Review**, 1979. Disponível em: <https://hbr.org/2021/05/break-down-change-management-into-small-steps>. Acesso em: 8 mar. 2025.

KOTTER, John, SCHLESINGER, Leonard. *Choosing Strategies for Change*. **Harvard Business Review**, 1979. Disponível em: <https://hbr.org/1979/03/choosing-strategies-for-change-2>. Acesso em: 8 mar. 2025.

LI, Yan; SHANG, Huping. *Service quality, perceived value, and citizens' continuous-use intention regarding e-government: Empirical evidence from China*. **Information & Management**, v. 57, n. 3, 2020. DOI: <https://doi.org/10.1016/j.im.2019.103197>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0378720617306912>. Acesso em: 8 mar. 2025.

MOORE, Mark (1997). *Creating Public Value: Strategic Management in Government*. **International Public Management Journal**, v. 2, n. 1, pp. 158-167, 1997. DOI: [https://doi.org/10.1016/S1096-7494\(00\)87438-3](https://doi.org/10.1016/S1096-7494(00)87438-3). Disponível em: https://www.researchgate.net/publication/257496373_Creating_public_value_Strategic_management_in_government_Cambridge_MA_Harvard_University_Press_by_Mark_Moore. Acesso em: 8 mar. 2025.

OSTROFF, Frank. *Change Management in Government*. **Harvard Business Review**, 2006. Disponível em: <https://hbr.org/2006/05/change-management-in-government>. Acesso em: 8 mar. 2025.

SUPERIOR TRIBUNAL DE JUSTIÇA. **Plano Estratégico STJ 2021-2026**: De mãos dadas: Magistratura e Cidadania. Brasília: Superior Tribunal de Justiça, 2021. Disponível em: https://www.stj.jus.br/sites/portallp/SiteAssets/Institucional/Gestao-estrategica/Planejamento-estrategico/planoEstrategico_2021_2026.pdf. Acesso em: 8 mar. 2025.

VON NEUMANN, John. Method in the physical sciences. **Collected Works**, pp. 491-498, 1955.

REFERÊNCIAS

- AGÊNCIA GOV. Internet avança mais rápido em áreas rurais do Brasil e chega a 81% dos domicílios. **Tecnologia**, 2024. Disponível em: <https://agenciagov.ebc.com.br/noticias/202408/internet-foi-acessada-em-72-5-milhoes-de-domicilios-do-pais-em-2023> Acesso em: 19 fev. 2025.
- AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES. Plano Nacional de Banda Larga. **Universalização e Ampliação do Acesso**, 2020. Disponível em: <https://www.gov.br/anatel/pt-br/regulado/universalizacao/plano-nacional-de-banda-larga>. Acesso em: 8 mar. 2025.
- AMERICA'S CYBER DEFENSE AGENCY. *Understanding Denial-of-Service Attacks*. **News**, 2021. Disponível em: <https://www.cisa.gov/news-events/news/understanding-denial-service-attacks>. Acesso em: 8 mar. 2025.
- BHATTACHERJEE, Anol. *Understanding Information Systems Continuance: An Expectation-Confirmation Model*. **MIS Quarterly**, University of Minnesota, v. 25, n. 3, pp. 375-370, 2001. DOI: <https://doi.org/10.2307/3250921>. Disponível em: <https://www.jstor.org/stable/3250921>. Acesso em: 8 mar. 2025.
- BRASIL. [Constituição (1988)]. **Constituição da República Federativa do Brasil de 1988**. Brasília, DF: Presidência da República, [2016]. Disponível em: http://www.planalto.gov.br/ccivil_03/Constituicao/Constituicao.htm. Acesso em: 8 mar. 2025.
- BRASIL. **Lei nº 11.419, de 19 de dezembro de 2006**. Dispõe sobre a informatização do processo judicial; altera a Lei nº 5.869, de 11 de janeiro de 1973 – Código de Processo Civil; e dá outras providências. Brasília. 2006. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2006/lei/111419.htm. Acesso em: 03 mar. 2025.
- CHANG, Connie; DIBB, Sally. *Reviewing and conceptualising customer-perceived value*. **The marketing review**, v. 12, n. 3, pp. 253-274, 2012. DOI: <https://doi.org/10.1362/146934712X13420906885395>. Disponível em: <https://oro.open.ac.uk/37703/>. Acesso em: 8 mar. 2025.
- CNJ. Judiciário vai receber apenas processos eletrônicos a partir de março de 2022. **Agência CNJ de Notícias**. Brasília, 2021. Disponível em: <https://www.cnj.jus.br/judiciario-vai-receber-apenas-processos-eletronicos-a-partir-de-marco-de-2022/>. Acesso em: 03 mar. 2025
- CNJ. Justiça 4.0 **Tecnologia da Informação e Comunicação**. Brasília, s.d. Disponível em: <https://www.cnj.jus.br/tecnologia-da-informacao-e-comunicacao/justica-4-0/>. Acesso em: 03 mar. 2025
- CNJ. Justiça em Números. **Agência CNJ de Notícias**. Brasília, 2024. Disponível em: <https://www.cnj.jus.br/justica-em-numeros-2024-barroso-destaca-aumento-de-95-em-novos-processos/>. Acesso em: 19 dez. 2024.
- COALI, Andrea; NOVELLI, Elena; SIRIGIRI, Anusha; SPINA, Chiara. *A scientific approach to decision-making: Key tools and design principles*. **Journal of Business Venturing Design**, v. 3, 2024. DOI: <https://doi.org/10.1016/j.jbvd.2024.100023>. Disponível em:

<https://www.sciencedirect.com/science/article/pii/S2667277424000045?via%3Dihub>. Acesso em: 03 mar. 2025.

CONJUR (Luís Roberto Barroso). Afinal, quanto vale o Poder Judiciário? **Opinião**, 2024a. Disponível em: <https://www.conjur.com.br/2024-fev-25/afinal-quanto-vale-o-poder-judiciario/>. Acesso em: 8 mar. 2025.

CONJUR (Maurício Cardoso). De cada dez processos, nove terminam na primeira instância. **Onde a justiça acontece**, 2024b. Disponível em: <https://www.conjur.com.br/2024-mai-23/de-cada-dez-processos-nove-terminam-na-primeira-instancia/>. Acesso em: 20 fev. 2025.

DATATRACKER. **RFC 4180** – Common Format and MIME Type. Disponível em: <https://datatracker.ietf.org/doc/html/rfc4180.html>. Acesso em: 8 mar. 2025.

DELONE, William; MCLEAN, Ephraim. *Information Systems Success: The Quest for the Dependent Variable*. **Information Systems Research**, v. 3, n. 1, pp. 60-95, 1992. DOI: <https://doi-org.ez54.periodicos.capes.gov.br/10.1287/isre.3.1.60>.

DELONE, William; MCLEAN, Ephraim. *The DeLone and McLean model of information systems success: A ten-year update*. **Journal of management information systems**, v. 19, n. 4, pp. 9-30, 2003. DOI: <https://doi.org/10.1080/07421222.2003.11045748>. Disponível em: <https://www.tandfonline.com/doi/abs/10.1080/07421222.2003.11045748>. Acesso em: 8 mar. 2025.

DODDS, William; MONROE, Kent. *Effects of Price, Brand, and Store Information on Buyers' Product Evaluations*. **Journal of Marketing Research**, v. 28, n. 3, pp. 307-319. DOI: <https://doi.org/10.2307/3172866>. Disponível em: <https://www.jstor.org/stable/3172866?origin=crossref>. Acesso em: 8 mar. 2025.

DUNN, Peter. **Scientific Research and Methodology: An introduction to quantitative research in science and health**, 2021. Disponível em: <https://bookdown.org/pkaldunn/Book>. Acesso em: 8 mar. 2025

EVERITT, Brian.; SKRONDAL, Anders. **The Cambridge Dictionary of Statistics 4th Edition**, Cambridge University Press, 2010. Disponível em: <https://www.stewartschultz.com/statistics/books/Cambridge%20Dictionary%20Statistics%204th.pdf>. Acesso em: 8 mar. 2025.

FIGUEIRAS, Fernando; PALOTTI, Pedro; CIRENO, Flavio. *Digital Transformation and Public Services Delivery in Brazil*. **1º Congresso do INCT.DD**, 2018. DOI: 10.13140/RG.2.2.30884.32648. Disponível em: https://www.researchgate.net/publication/330145305_Digital_Transformation_and_Public_Services_Delivery_in_Brazil. Acesso em: 8 mar. 2025.

GUTMAN, Jonathan. *A means-end chain model based on consumer categorization processes*. **Journal of Marketing**, v. 46, n. 2, 1982. DOI: <https://doi.org/10.1177/002224298204600>. Disponível em: <https://journals.sagepub.com/doi/10.1177/002224298204600207>. Acesso em: 8 mar. 2025.

HARS, Roelof. **Reserach Skills Reader**. University of Groningen, 2024. DOI: <https://doi.org/10.21827/67a4ceba2ceca>. Disponível em: <https://opentextbooks.rug.nl/rspremsc/>. Acesso em: 11 fev. 2025.

IBGE. **Censo 2022**. Disponível em: <https://censo2022.ibge.gov.br/panorama/>. Acesso em: 26 fev. 2024.

KEARNS, Iain. **Public value and e-government**. London: Institute for Public Policy Research, 2004. Disponível em: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=1959beb80056dbb823ae035303361b6285e4a480>. Acesso em: 8 mar. 2025.

LI, Yan; SHANG, Huping. *Service quality, perceived value, and citizens' continuous-use intention regarding e-government: Empirical evidence from China*. **Information & Management**, v. 57, n. 3, 2020. DOI: <https://doi.org/10.1016/j.im.2019.103197>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0378720617306912>. Acesso em: 8 mar. 2025.

MENSAH, Isaac Kofi; ZENG, Guohua; MWAKAPESA, Deborah Simon. *Understanding the drivers of the public value of e-government: Validation of a public value e-government adoption model*. **Frontiers in Psychology**, v. 13, 2022. DOI: <https://doi.org/10.3389/fpsyg.2022.962615>. Disponível em: <https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.962615/full>. Acesso em: 8 mar. 2025.

MINISTÉRIO DA FAZENDA. Arrecadação federal de 2023 somou R\$ 2,318 trilhões. **Notícias**, 2024. Disponível em: <https://www.gov.br/fazenda/pt-br/assuntos/noticias/2024/janeiro/arrecadacao-federal-de-2023-somou-r-2-318-trilhoes>. Acesso em: 8 mar. 2025.

MOORE, Mark (1997). *Creating Public Value: Strategic Management in Government*. **International Public Management Journal**, v. 2, n. 1, pp. 158-167, 1997. DOI: [https://doi.org/10.1016/S1096-7494\(00\)87438-3](https://doi.org/10.1016/S1096-7494(00)87438-3). Disponível em: https://www.researchgate.net/publication/257496373_Creating_public_value_Strategic_management_in_government_Cambridge_MA_Harvard_University_Press_by_Mark_Moore. Acesso em: 8 mar. 2025.

NDOU, Valentina Dardha. *E-government for developing countries: Opportunities and challenges*. **The Electronic Journal on Information Systems in Developing Countries**, v. 18, n. 1, pp. 1-24, 2004.

OECD. *Brazil's path towards a digital government for a digital economy and society*. In: **Digital Government Review of Brazil: Towards the Digital Transformation of the Public Sector**. Paris: OECD Publishing, 2018. DOI: <https://doi.org/10.1787/9789264307636-4-en>. Disponível em: https://www.oecd.org/content/dam/oecd/en/publications/reports/2018/11/digital-government-review-of-brazil_g1g98df2/9789264307636-en.pdf. Acesso em: 8 mar. 2025.

OLIVER, Richard. *A Cognitive Model of the Antecedents and Consequences of Satisfaction Decisions*. **Journal of Marketing Research**, v. 17, n. 4, pp. 460-469, 1980.

DOI:<https://doi.org/10.2307/3150499>. Disponível em:
<https://www.jstor.org/stable/3150499?origin=crossref>. Acesso em: 8 mar. 2025.

PYTHON. Pickle — Python object serialization. Disponível em:
<https://docs.python.org/3/library/pickle.html>. Acesso em: 8 mar. 2025.

QUE NOTEBOOK COMPRAR. **Qual notebook comprar**, s.d. Disponível em:
https://quenotebookcomprar.com.br/ofertas/?_sfm_spec_laptop_category=De+entrada&post_types=notebooks&sort_order=_sfm_sale_lowest-price+asc+num. Acesso em: 3 mar. 2025.

REICHHELD, Frederick. *The One Number You Need to Grow*. **Harvard Business Review**, 2003. Disponível em: <https://hbr.org/2003/12/the-one-number-you-need-to-grow>. Acesso em: 8 mar. 2025.

SADEK, Maria Tereza. Judiciário: mudanças e reformas. **Estudos Avançados**, v. 18, n. 51, 2004. DOI: <https://doi.org/10.1590/S0103-40142004000200005>. Disponível em:
<https://www.scielo.br/j/ea/a/rmr7WmNQZLyrPJ7VfWLFPy/>. Acesso em: 8 mar. 2025.

SECRETARIA DE COMUNICAÇÃO SOCIAL. Renda média per capita no Brasil cresce 11,5% e atinge maior valor em 12 anos. **Notícias**, 2024. Disponível em:
<https://www.gov.br/secom/pt-br/assuntos/noticias/2024/04/renda-media-per-capita-no-brasil-cresce-11-5-e-atinge-maior-valor-em-12-anos>. Acesso em: 8 mar. 2025.

SOUZA de, Natércia; NUNES, André. A evolução da transformação digital no setor público no Brasil, no período de 2000 a 2020. **Universitas**, n. 29, 2021.

STARLINK. **Starlink para residências**, s.d. Disponível em:
<https://www.starlink.com/br/residential> Acesso em: 19 fev. 2025.

SUPERIOR TRIBUNAL DE JUSTIÇA. Acordo com AGU intensifica desjudicialização e alcança mais de dois milhões de processos. **Notícias**, 2023a. Disponível em:
<https://www.stj.jus.br/sites/portalp/Paginas/Comunicacao/Noticias/2023/17082023-Acordo-com-AGU-intensifica-desjudicializacao.aspx>. Acesso em: 9 fev. 2025.

SUPERIOR TRIBUNAL DE JUSTIÇA. Atendimento Judicial. **Fale Conosco**, s.d. Disponível em: <https://www.stj.jus.br/sites/portalp/Contato-e-ajuda/Fale-conosco/Atendimento-judicial>. Acesso em: 29 dez. 2024.

SUPERIOR TRIBUNAL DE JUSTIÇA. Balcão Virtual atende advogados e público em geral por videoconferência. **Secretaria de Comunicação Social**, 2021a. Disponível em:
<https://cartadeservicos.stj.jus.br/atendimento-judicial/atendimento-judicial-presencial-espaco-do-advogado/>. Acesso em: 03 mar. 2025.

SUPERIOR TRIBUNAL FEDERAL. Corte rejeita recurso da J&F e mantém tramitação de ação que discute multa bilionária em acordo de leniência. **Notícias**, 2023b. Disponível em:
<https://www.stj.jus.br/sites/portalp/Paginas/Comunicacao/Noticias/2023/16032023-Corte-rejeita-recurso-da-J-F-e-mantem-tramitacao-de-acao-que-discute-multa-bilionaria-em-acordo-de-leniencia.aspx>. Acesso em: 26 fev. 2024.

SUPERIOR TRIBUNAL FEDERAL. Detalhamento das Ações. **LOA**, 2024a. Disponível em: https://transparencia.stj.jus.br/wp-content/uploads/2024_Lei-Orcamentaria-_LOA-.pdf. Acesso em: 17 nov. 2024.

SUPERIOR TRIBUNAL DE JUSTIÇA. **Plano Diretor de Tecnologia da Informação e Comunicação 2023-2024**. Secretaria de Tecnologia da Informação e Comunicação: Brasília, 2024b. Disponível em: https://transparencia.stj.jus.br/wp-content/uploads/PDTIC_2023_2024_Revisao_Janeiro2024.pdf. Acesso em: 8 mar. 2025.

SUPERIOR TRIBUNAL DE JUSTIÇA. **Plano Estratégico STJ 2021-2026**: De mãos dadas: Magistratura e Cidadania. Brasília: Superior Tribunal de Justiça, 2021b. Disponível em: https://www.stj.jus.br/sites/portalp/SiteAssets/Institucional/Gestao-estrategica/Planejamento-estrategico/planoEstrategico_2021_2026.pdf. Acesso em: 8 mar. 2025.

SUPERIOR TRIBUNAL DE JUSTIÇA. Resolução STJ/GP n. 9, de 25 de março de 2022. **Atos Normativos do STJ**, 2022. Disponível em: <https://bdjur.stj.jus.br/items/d82fc044-347e-459e-9de0-d7f18a853f88>. Acesso em: 8 mar. 2025.

SUPERIOR TRIBUNAL DE JUSTIÇA. Missão, Visão e Valores. **Transparência e prestação de contas**, s.d. Disponível em: <https://transparencia.stj.jus.br/gestao/objetivos-estrategicos/missao-visao-valores-do-stj/>. Acesso em: 9 fev. 2025.

SUPERIOR TRIBUNAL DE JUSTIÇA. Tribunal lança Sala Acessível no Balcão Virtual para atender usuários com deficiência. **Notícias**, 2023c. Disponível em: <https://www.stj.jus.br/sites/portalp/Paginas/Comunicacao/Noticias/2023/20042023-Tribunal-lanca-Sala-Acessivel-no-Balcao-Virtual-para-atender-usuarios-com-deficiencia.aspx>. Acesso em: 19 fev. 2025.

TELETIME. Estudo da Anatel mostra queda de preços na telefonia e banda larga. **Conectividade** (Marcos Urupá), 2024. Disponível em: <https://teletime.com.br/22/03/2024/estudo-da-anatel-mostra-queda-de-precos-na-telefonia-e-banda-larga/>. Acesso em: 19 fev. 2025.

TWIZEYIMANA, Jean Damascene; ANDERSSON, Annika. *The public value of E-Government – A literature review*. **Government Information Quarterly**, v. 36, n. 2, pp. 167-178. DOI:<https://doi.org/10.1016/j.giq.2019.01.001>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0740624X1730196X?via%3Dihub>. Acesso em: 8 mar. 2025.

UN Department of Economic and Social Affairs. **UN E-Government Survey**, 2022. Disponível em: <https://publicadministration.un.org/egovkb/en-us/Data/Country-Information/id/24-Brazil>. Acesso em: 22 jul. 2023.

VON NEUMANN, J. (1955). Method in the physical sciences. **Collected Works**, pp. 491-498.

ZAUNER, Alexander; KOLLER, Monika; Hatak, Isabella; Walla, Peter. *Customer perceived value — Conceptualization and avenues for future research*. **Cogent Psychology**, v. 2. DOI: <https://doi.org/10.1080/23311908.2015.1061782>. Disponível em: <https://www.tandfonline.com/doi/full/10.1080/23311908.2015.1061782>. Acesso em: 8 mar. 2025.

ZEITHAML, Valarie. *Consumer Perceptions of Price, Quality, and Value: A Means-End Model and Synthesis of Evidence*. ***Journal of Marketing***, v. 52, n. 3, pp. 2-22, 1988. DOI: <https://doi.org/10.2307/1251446>. Disponível em: <https://www.jstor.org/stable/1251446>. Acesso em: 8 mar. 2025.

ANEXO A – QUESTÕES DO MODELO DE LI E SHANG (2020)

Quadro 5 – Questões do modelo Li e Shang (2020)

Questões do modelo Li e Shang	
SYQ	SYQ1. I can successfully log on to the government website every time.
	SYQ2. I can successfully visit the related links provided at the home page.
	SYQ3. Using the website lets me surf effortlessly through relevant webpages while performing my e-government transactions.
	SYQ4. The function modules on government websites are arranged in a user-friendly way.
REL	REL1. The website system operated stably for the e-government transactions.
	REL2. Faults I encountered when using websites can be corrected as soon as possible to ensure the reliable operation of the system.
	REL3. All the functions and services on the website operate normally.
SCT	SCT1. I know that my personal information submitted to the government websites is used in a secure manner.
	SCT2. The website has adequate security measures to protect my personal information from being stolen or leaked.
	SCT3. I trust that hackers will not be able to access the information I provide when performing my e-government transactions.
ACB	ACB1. I'm informed and introduced to how to use the website by the government through various initiatives, which makes me have equal access to online public service.
	ACB2. I do not encounter any problem in accessing the website using my computer to perform my e-government transactions.
	ACB3. I do not need to perform complicated technical configurations in order to access services on government websites.
SEC	SEC1. I am able to complete different e-government transactions using the same website.
	SEC2. All functions needed to complete my required service are available from the websites.
	SEC3. The website lets me complete transaction provided by different governmental organizations.
IQ	IQ1. Information provided at the website is up-to-date.
	IQ2. I can quickly find the required information on the website.
	IQ3. The information provided by the website is well organized.
	IQ4. The website provides accurate information about relevant services.
RPS	RPS1. The website can inform me of the progress of my transaction by e-mail or message.
	RRS2. The website takes prompt action when I encounter problems performing my task.
	RPS3. The website system responded in a timely fashion during my e-government transaction.
INT	INT1. Using the websites makes me communicate with government more easily.
	INT2. Public opinion surveys and interactive interviews are often held on government websites.
	INT3. Using the websites makes me feel more involved in public decisions.
EFV	VE1. Using government website is an efficient way to access public service.
	VE2. I value the convenience brought by government website in service delivery.
	VE3. Using e-government system to access public service increases the government efficiency.
DEV	VD1. Using government website to access service makes me know more about what the government is doing.
	VD2. Using government website to access service brings me closer to public affairs.
	VD3. Using government website to access service increases the government transparency.

	VD4. Using government website to access service increases the government accountability.
INV	VI1. Using government website makes the public service more available to more people.
	VI2. Using government website is valuable to providing equal public service to all citizens.
	VI3. Using government website makes the disadvantaged groups benefit more from public service.
PSV	OSV1. Overall, I believe that using government website to access public service provides public value.
	OSV2. The value I receive from government website is worth of the time, effort and money I have invested.
	OSV3. The value derives from services on government website is worth of the time, effort and money the government have invested.
CUI	CUI1. I intend to increase the use of government websites in the future.
	CUI2. I will recommend others to use government websites.
	CUI3. In the future, I will consider e-Government websites to be my first choice to do business with the government.
SAT	SAT1. I am satisfied with the service I received from the government websites.
	SAT2. I am very pleased with my past experience of using government website service.
	SAT3. The government website provides satisfied public services that meet my needs.

Fonte: Li e Shang (2020, p. 11).

APÊNDICE A – QUESTIONÁRIO DA OUVIDORIA

Questão	Min	Máximo	Média	Desvio Padrã
ID	1	3955		
Hora de início	8/12/21 16:05	9/16/21 13:28		
Hora de conclusão	8/12/21 16:06	9/16/21 13:28		
Email	Informação Pessoal			
Nome	Informação Pessoal			
Facilidade em acessar o Balcão Virtual do STJ	-1	10	9.60	1.18
Ferramenta de videochamada (Zoom)	-1	10	9.61	1.15
Profissionalismo (presteza e cordialidade) dos consultores	-1	10	9.88	0.78
Qualidade das informações prestadas	-1	10	9.76	1.06
Duração do atendimento	-1	10	9.73	1.08
Qual a probabilidade de você recomendar este serviço a um amigo ou colega de profissão?	-1	10	9.81	1.00
SATISFAÇÃO GERAL - De uma forma geral, como você avaliaria o serviço "Balcão Virtual"?	-1	10	9.74	1.00
SUGESTÕES E CRÍTICAS - Deixe aqui suas sugestões ou críticas sobre o serviço "Balcão Virtual". Caso deseje receber uma resposta, por favor, deixe também seu nome completo e e-mail para que seja r...	N/A			
Faixa etária	0	0		
Relacionamento com o STJ				
Escolaridade				
Local de Residência				

Fonte: Questionário da Ouvidoria do Superior Tribunal de Justiça.

APÊNDICE B – EXEMPLO DE DADOS DE CONSULTA PROCESSUAL

Dados de consulta processual/*open data* (<https://processo.stj.jus.br/processo/pesquisa/>) — estas informações são importantes porque permitem detalhar o ramo do direito, Tribunal de origem, qual ministro é o relator e as partes no processo, destacados em **negrito** na lista abaixo, que podem auxiliar no recorte de públicos específicos para mensuração de indicadores. A consulta abaixo refere-se a um processo citado em um artigo público⁴³:

PROCESSO: **EMBARGOS DE DIVERGÊNCIA EM AGRAVO EM RECURSO ESPECIAL**

EMBARGANTE: **ARCOS DOURADOS COMÉRCIO DE ALIMENTOS SA**

OUTRO NOME: **ARCOS DOURADOS COMÉRCIO DE ALIMENTOS SA**

ADVOGADO: ALAN ADUALDO PERETTI DE ARAÚJO - RJ127615

ADVOGADO: ANDRE TORRES DOS SANTOS - DF035161

ADVOGADO: HELOISA LOHANE GONÇALVES DA SILVA - DF064461

ADVOGADO: JÚLIA MELLO LEITÃO MOREIRA DE CARVALHO - RJ228062

ADVOGADO: LUIZ ROBERTO PEROBA BARBOSA - RJ129738

ADVOGADO: WILLIAM ROBERTO CRESTANI - SP258602

EMBARGADO: **MUNICÍPIO DE RIO DE JANEIRO**

PROCURADOR: MARIA IZABEL VIEIRA DE BRITO - RJ079272

INTERES.: **ORDEN DOS ADVOGADOS DO BRASIL CONSELHO FEDERAL - "AMICUS CURIAE"**

ADVOGADO: FRANCIMEIRE HERMOSINA DE BRITO - DF037576

ADVOGADO: PRISCILLA LISBOA PEREIRA - DF039915

ADVOGADO: ULISSES RABANEDA DOS SANTOS - MT008948

LOCALIZAÇÃO: Entrada em GABINETE DO MINISTRO JOÃO OTÁVIO DE NORONHA em 18/04/2023

TIPO: Processo eletrônico.

AUTUAÇÃO: 06/04/2023

NÚMERO ÚNICO: 0132488-96.2009.8.19.0001

RELATOR(A): **Min. JOÃO OTÁVIO DE NORONHA - CORTE ESPECIAL**

RAMO DO DIREITO: **DIREITO TRIBUTÁRIO**

ASSUNTO(S): DIREITO TRIBUTÁRIO, Dívida Ativa (Execução Fiscal). Impostos, ISS/ Imposto sobre Serviços.

TRIBUNAL DE ORIGEM: **TRIBUNAL DE JUSTIÇA DO ESTADO DO RIO DE JANEIRO**

NÚMEROS DE ORIGEM: 01324889620098190001, 04199229520168190001, 1324889620098190001, 202024510896, 4199229520168190001.

1 volume, nenhum apenso.

ÚLTIMA FASE: 26/05/2023 (19:11) JUNTADA DE PETIÇÃO DE IMPUGNAÇÃO Nº 503468/2023

⁴³ Caso “OAB requer ingresso em causa do STJ que fixou honorários por equidade” disponível para consulta em: <https://www.migalhas.com.br/quentes/372158/oab-requer-ingresso-em-causa-do-stj-que-fixou-honorarios-por-equidade>. Acesso em: 30 dez. 2024.

APÊNDICE C – TIPOS DE ATENDIMENTO

Tabela 20 – Relação dos tipos de atendimento e assuntos

Atendimento	Contagem Balcão Virtual	Percentual
Despachar/Falar no Gabinete	4389	16,86%
CPE/Protocolo	3689	14,17%
Previsão de julgamento/Distribuição	2393	9,19%
Andamento/Localização processual	2258	8,67%
Trânsito em Julgado/Baixa/Remessa ao STF	2071	7,96%
Contato com outros tribunais	1113	4,28%
Previsão de distribuição/Triagem	1081	4,15%
Sustentação oral	839	3,22%
DJE/Publicação/Intimação	714	2,74%
Correção de autuação/Habilitação	673	2,59%
Sessões julgamento/Preferência	635	2,44%
HDE/Carta rogatória	609	2,34%
Julgamento virtual	514	1,97%
Despesas processuais	511	1,96%
Vista para o MPF	500	1,92%
GRU	405	1,56%
Informações processuais	395	1,52%
Plantão/Recesso/Feriado	363	1,39%
Informações Administrativas	345	1,33%
CPE – Indisponibilidade	320	1,23%
Memoriais	287	1,10%
Conexão interrompida	268	1,03%
Certidão	237	0,91%
Consultoria jurídica	225	0,86%
Execução/RPV/Precatório	223	0,86%
Recurso repetitivo/Jurisprudência	157	0,60%
GRU – Indisponibilidade	136	0,52%
Malote digital	127	0,49%
Ciência antecipada	108	0,41%
Segredo de Justiça	106	0,41%
Corte especial (Sigilo)	92	0,35%
Chave de acesso	73	0,28%
Sistema <i>Push</i>	67	0,26%
Publicação/Intimação	35	0,13%
Depósito judicial	24	0,09%
Tentativa de fraude	18	0,07%
BacenJUD (SISBAJUD)	11	0,04%
Informações administrativas	9	0,03%
Audiência presidência	6	0,02%
Preferência julgamento	2	0,01%
Protocolo judicial	2	0,01%
Andamento/Localização processual/ Execução/RPV/Precatório	1	0,00%
Previsão de despachar/Falar no Gabinete	1	0,00%

Habilitação nos autos	1	0,00%
------------------------------	----------	--------------

Assunto	Contagem Telefone	Percentual
Informações Processuais	20333	39.8%
Despachar/Falar no Gabinete	4865	9.5%
Ligação muda/caiu	3312	6.5%
CPE	2971	5.8%
Previsão de Julgamento/Distribuição	2842	5.6%
Transferência interna	2493	4.9%
Despesas Processuais	1624	3.2%
GRU	1263	2.5%
Sustentação Oral	1169	2.3%
Trânsito em Julgado/Baixa/Remessa ao STF	1100	2.2%
Certidão	983	1.9%
Falar na Coordenadoria	748	1.5%
Sessões Julgamento/Preferência	722	1.4%
CPE – Indisponibilidade	670	1.3%
Correção de Autuação/Habilitação	658	1.3%
Protocolo Judicial	589	1.2%
Plantão/Recesso/Feriado	565	1.1%
HDE/Carta Rogatória	499	1.0%
Consultoria Jurídica	415	0.8%
GRU – Indisponibilidade	407	0.8%
Julgamento Virtual	395	0.8%
Informações Administrativas	392	0.8%
Contato com outros tribunais	328	0.6%
Malote Digital	288	0.6%
DJE/Publicação/Intimação	237	0.5%
Recurso Repetitivo/Jurisprudência	225	0.4%
Memoriais	188	0.4%
Processos sigilosos (corte especial)	158	0.3%
Indisponibilidade de sistema	125	0.2%
Preferência Julgamento	88	0.2%
Fraude	85	0.2%
Habilitação nos autos	78	0.2%
Sistema <i>Push</i>	70	0.1%
Chave de Acesso	65	0.1%
Depósito Judicial	45	0.1%
BacenJUD (SISBAJUD)	43	0.1%
Carta Rogatória	28	0.1%
Corte Especial (Sigilo)	8	0.0%
Despachar com Ministro/Plantão/Recesso/Feriado	1	0.0%

Fonte: Banco de dados do Balcão Virtual (2024).

APÊNDICE D – EXEMPLO DE PROBLEMA DE IDENTIFICAÇÃO DE USUÁRIO

Na Tabela 21, o mesmo usuário se identifica de várias maneiras diferentes — várias grafias do nome, CPFs variados, e-mail às vezes informado, às vezes não informado. Se o usuário fosse devidamente identificado, haveria apenas uma entrada com 26 atendimentos.

Tabela 21 – Exemplo de problema de identificação única de um usuário

Nome	E-mail	CPF/OAB	Ligações
<i>aaaaaa</i>	Em branco	000.000.000/00	1
		ddd.ddd.ddd/vv	1
		Inválido	2
	Não informou	Inválido	1
<i>aaaaaa da ddddd ccccc</i>	Não informou	ddd.ddd.ddd/vv	1
<i>aaaaaa de eeeeeee ccccc</i>	Não informou	EENNNN (OAB)	1
<i>aaaaaa bbbbbb</i>	Em branco	ddd.ddd.ddd/vv	1
<i>aaaaaa bbbbbb da ddddd ccccc</i>	nnnnnnnnnnnnnnnnnnnn@gmail.com	ddd.ddd.ddd/vv	4
	Em branco	ddd.ddd.ddd/vv	10
	Parte sem e-mail	ddd.ddd.ddd/vv	1
<i>aaaaaa bbbbbb ccccc</i>	Em branco	ddd.ddd.ddd/vv	1
<i>aaaaaa bbbbbb da ddddd ccccc</i>	Em branco	ddd.ddd.ddd/vv	1
<i>aaaaaa bbbbbb da ddddd ccccc</i>	Em branco	ddd.ddd.ddd/vv	1

Fonte: Elaborado pelo autor com base no banco de dados do Balcão Virtual (2024).

APÊNDICE E – TRIBUNAIS DE ORIGEM

Tabela 22 – Lista detalhada dos 40 tribunais que mais figuram nos processos referenciados no atendimento do Balcão Virtual

Tribunal de Origem	Contagem	Percentual
TRIBUNAL DE JUSTIÇA DO ESTADO DO RIO DE JANEIRO	2611	15,60%
TRIBUNAL DE JUSTIÇA DO ESTADO DE SÃO PAULO – CONSELHEIRO FURTADO – PÁTIO DO COLÉGIO	1720	10,28%
SUPERIOR TRIBUNAL DE JUSTIÇA	1272	7,60%
TRIBUNAL DE JUSTIÇA DO DISTRITO FEDERAL E DOS TERRITÓRIOS	937	5,60%
TRIBUNAL DE JUSTIÇA DO ESTADO DE SÃO PAULO – RUA DA GLÓRIA	882	5,27%
TRIBUNAL DE JUSTIÇA DO ESTADO DE MINAS GERAIS	841	5,02%
TRIBUNAL DE JUSTIÇA DO ESTADO DA BAHIA	595	3,55%
TRIBUNAL REGIONAL FEDERAL DA 2ª REGIÃO	503	3,01%
TRIBUNAL DE JUSTIÇA DO ESTADO DO RIO GRANDE DO SUL	499	2,98%
TRIBUNAL REGIONAL FEDERAL DA 5ª REGIÃO	480	2,87%
TRIBUNAL DE JUSTIÇA DO ESTADO DE GOIÁS	472	2,82%
TRIBUNAL DE JUSTIÇA DO ESTADO DO PARANÁ	459	2,74%
TRIBUNAL DE JUSTIÇA DO ESTADO DE SÃO PAULO – AV. BRIGADEIRO	433	2,59%
TRIBUNAL REGIONAL FEDERAL DA 4ª REGIÃO	419	2,50%
TRIBUNAL DE JUSTIÇA DO ESTADO DE SANTA CATARINA	359	2,14%
TRIBUNAL DE JUSTIÇA DO ESTADO DO CEARÁ	357	2,13%
TRIBUNAL DE JUSTIÇA DO ESTADO DE PERNAMBUCO	352	2,10%
TRIBUNAL DE JUSTIÇA DO ESTADO DO MARANHÃO	332	1,98%
TRIBUNAL REGIONAL FEDERAL DA 3ª REGIÃO	330	1,97%
TRIBUNAL REGIONAL FEDERAL DA 1ª REGIÃO	306	1,83%
TRIBUNAL DE JUSTIÇA DO ESTADO DE MATO GROSSO	289	1,73%

TRIBUNAL DE JUSTIÇA DO ESTADO DO PARÁ	225	1,34%
TRIBUNAL DE JUSTIÇA DO ESTADO DE MATO GROSSO DO SUL	207	1,24%
TRIBUNAL DE JUSTIÇA DO ESTADO DO AMAZONAS	197	1,18%
TRIBUNAL DE JUSTIÇA DO ESTADO DO PIAUÍ	191	1,14%
TRIBUNAL DE JUSTIÇA DO ESTADO DO RIO GRANDE DO NORTE	143	0,85%
TRIBUNAL DE JUSTIÇA DO ESTADO DO TOCANTINS	142	0,85%
TRIBUNAL DE JUSTIÇA DO ESTADO DO ESPÍRITO SANTO	130	0,78%
TRIBUNAL DE JUSTIÇA DO ESTADO DA PARAÍBA	127	0,76%
TRIBUNAL DE JUSTIÇA DO ESTADO DE RONDÔNIA	123	0,73%
TRIBUNAL DE JUSTIÇA DO ESTADO DE SERGIPE	93	0,56%
TRIBUNAL DE JUSTIÇA DO ESTADO DO AMAPÁ	84	0,50%
JUSTIÇA ESTADUAL 1ª INSTÂNCIA SÃO PAULO	62	0,37%
TRIBUNAL DE JUSTIÇA DO ESTADO DE ALAGOAS	59	0,35%
TRIBUNAL DE JUSTIÇA DO ESTADO DO ACRE	44	0,26%
TRIBUNAL DE JUSTIÇA DO ESTADO DE RORAIMA	40	0,24%
JUSTIÇA ESTADUAL 1ª INSTÂNCIA DO RIO DE JANEIRO	37	0,22%
JUSTIÇA FEDERAL 1ª INSTÂNCIA DISTRITO FEDERAL	23	0,14%
Outros	362	2,16%

Fonte: Elaborado pelo autor com base no banco de dados do Balcão Virtual (2024).

APÊNDICE F – EXEMPLO DE REGISTRO DE ATENDIMENTO ANONIMIZADO

Tabela 23 – Exemplo de registro de atendimento anonimizado

ID	22530	22529	22528
Title	XXXXXXXXXX YYYYYY ZZZZZZZ	Aaaaaa Bbbb Ccccccc	Mmmmm Nnnnnn Oo Ppppppp Qqqq
Hora de início	DD/MM/AA 15:22	DD/MM/AA 15:22	DD/MM/AA 15:16
Hora de conclusão	DD/MM/AA 15:29	DD/MM/AA 15:28	DD/MM/AA 15:26
Perfil	Advogado(a)	Cidadã(o)	Advogado(a)
CPF ou OAB	NNNNNNNNN	NNNNNNNNN	EENNNNNNN
E-mail	XXXYYYZZZ@YAHOO.C OM.BR	aaabbccc@gmail.c om	mmmmmmmmmm@mm mm.br
Assunto 1	Andamento/Localização processual	Contato com outros tribunais	Correção de autuação/Habilitação
Assunto 2	Contato com outros tribunais	X	X
Assunto 3	X	X	X
Descrição do atendimento	HC NNNNNN – Advogado que que o STJ reitere o ofício solicitando informações para o TJXX. Expliquei que o setor costuma aguardar pelo menos 20 dias para que o Tribunal/vara se manifeste. Como ele insistiu, eu entrei em contato com a CPPE para verificar se podem reiterar o ofício antes dos 20 dias e ela explicou que é só após os 20 mesmo. Pedi pro advogado aguardar o prazo mencionado e tbm tentar diligenciar essas informações junto ao TJXX.	Queria emitir certidão de objeto e pé do processo de nº NNNNNNNNNN NNN. Informei que esse processo não é do STJ. Ele não soube informar em qual Tribunal o processo transitou. Direcionei ele o Tribunal de <<cidade>>, tendo em vista a numeração e a localidade dele.	REsp NNNNNNN/<Estado> (AAAA/NNNNNNN-D) – Solicitou a correção do advogado. O número da OAB <estado>NNNNNN está errado, bem como retorna o advogado Aaaaaa Bbbbbb. O número correto é OAB <estado>MMMMMM M, advogado Ccccccc Ddddddd, conforme a procuração nos autos de fl. NN. Abertura de demanda DEMANDA Nº NNNNN
Dispositivo	Computador	Computador	Computador
Pessoa surda?	Não	Não	Não
Chat?	Não	Não	Não
Consultor	xxxxxxx@stj.jus.br	yyyyyyyyy@stj.ju s.br	zzzzzzzzz@stj.jus.br
Modificado por	Aaaaa bbb Cccc	Dddddd Eeeee	Fffffff Ggggg Hhhhhh
Acessibilidade	FALSO	FALSO	FALSO
Sugestão	FALSO	FALSO	FALSO
Enviar SAC	FALSO	FALSO	FALSO
Atendimento SAC	FALSO	FALSO	FALSO

Criado	DD/MM/AA 15:29	DD/MM/AA 15:28	DD/MM/AA 15:26
SJD	FALSO	FALSO	FALSO
Atendimento finalizado	FALSO	FALSO	FALSO

Fonte: Elaborado pelo autor com base no banco de dados do Balcão Virtual (2024).

APÊNDICE G – SOFTWARE DE TRATAMENTO E ANÁLISE DE DADOS

```

===== Config.py =====
#
# Arquivo com constantes
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
ATENDIMENTOS_PARQUET = "atendimentos.parquet"
LISTA_CONSULTA_PROCESSUAL_PARQUET = "lista_consulta_processual.parquet"
DIRETORIO_CONSULTA_PROCESSUAL_BATCH = "consultas_html"
FILE_DADOS_CONSULTADOS = "dados_consultados.parquet"
===== Stage1_dataToParquet.py
=====
#
# Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
# Copyright (c) STJ 2025 Licença CC BY-NC
#
# Programa para transformar os dados de entrada em xls para o formato Parquet que é mais
# eficiente para Python/Pandas
# PRIMEIRO PROGRAMA PARA RODAR!!
# Sanitiza dados e salva em formato eficiente
#
import numpy
import pandas
from pandas import DataFrame, Series
import re
import unicodedata
from datetime import time, timedelta, datetime
from sanitizaCPFOAB import sanitizaCPFOAB
from normalizeAtendimento import sanitizaAtendimento
import Ministros
import Processos
from accents import limpaString
# Program to convert all available data to Parquet format

# xls file name and corresponding sheet where the data is
XLS_FILE_LIST= {
    "Dados atualizados Balcão Virtual.xlsx" : "query 1"
    #, "Dados e-mail.xlsx" : "query (6) 1"
    #, "Dados Espaço do Advogado.xlsx" : "query (6) 1"
    #, "Dados Telefone.xlsx" : "query (5) 1"
    #"Gabinete Ministros.xlsx"
    #"Gabinete Ministros_paulo.xlsx"
}
PARQUET_FILE_EXTENSION = ".parquet"
XLS_DIRECTORY = r"dez2023 work"
PARQUET_DIRECTORY = r"parquet"

#ATENDIMENTOS_XLS = "dez2023 work\\Dados e-mail.xlsx"
#ATENDIMENTOS_SHEET = "query (6) 1"
#ATENDIMENTOS_PARQUET = "atendimentosEmail.parquet"

#### Note -> have to validate OAB/CPF
#
# Rotina principal de limpeza de dados
#
def clean_data(dirty_df: DataFrame) -> DataFrame:
    cont_longo_demais=0
    cont_duração_negativa=0
    cont_data_ruim=0
    cont_inicial=dirty_df.shape[0] # Numero inicial de linhas
    cont_nattype=0
    cont_null_name=0
    ## Dados vão até dezembro 2023, qualquer coisa depois de 1/1/24 está errada
    data_limite=numpy.datetime64('2024-01-01')
    SeteHoras = 25200 # 7h em segundos
    print(">>> IN Cleaning data")
    # Nos dados do BV nome do usuário é 'Title' nos outros é 'Nome'
    # Altera para Nome para ficar consistente
    if 'Title' in dirty_df.columns:
        print("Renaming Title")
        dirty_df.rename(columns={'Title': 'Nome'}, inplace=True)
    #print(dirty_df.info())

```

```

## Dados do BV tem lixo com Title == x < ==> drop row
df=dirty_df[dirty_df['Nome']!="x"]
# Drop linhas duplicadas
dirty_df.drop_duplicates(keep='first', inplace=True)
#dirty_df.reset_index(drop=True, allow_duplicates=False, inplace=True)
cont_ruim = cont_inicial - df.shape[0]
## Drop linhas com tempos invalidos: começo depois do fim, data depois da coleta de dados
Dez/23 ou duração maior do que 7hs (expediente STJ)
df.insert(len(df.columns),"Valid",True)
#print(df.info())
# SLOW
# check bad time stamps and drop from de sample
for index, row in df.iterrows():
    if pandas.isnull(row['Nome']):
        #print("Null name",row)
        df.at[index, 'Nome'] = ""
        cont_null_name += 1
    if pandas.isnull(row["Hora de conclusão"]) or pandas.isnull(row["Hora de início"]) :
        #print("Bad row NaTtype", row)
        df.at[index, 'Valid'] = False
        cont_nattype += 1
    else: # Valid times
        dt = timedelta.total_seconds(row["Hora de conclusão"] - row["Hora de início"])
        if( dt < 0): # Negative time, bad!
            #print("Negative: ",dt, index)
            #print(df.iloc[index, 0:4])
            cont_duração_negativa += 1
            df.at[index, 'Valid']=False
        if( dt > SeteHoras): # More than 5hs, bad
            #print("Too long", dt, index)
            #print(df.iloc[index, 0:4])
            cont_longo_demais +=1
            df.at[index, 'Valid'] = False
        if row["Hora de conclusão"] > data_limite or row["Hora de início"] > data_limite:
            cont_data_ruim += 1
            df.at[index, 'Valid'] = False

# Drop bad data
df=df[df['Valid']==True]
# Remove Valid colum
df.drop(columns=['Valid'],inplace=True)
##
## Normalizando e-mail
##
# TOLower e remove brancos aspás, etc.
##
print("CLEAN e-mail")
df['E-mail limpo'] = df['E-mail'].values.astype(str)
df['E-mail limpo'] = df['E-mail limpo'].str.strip().str.lower()
df['E-mail limpo'] = df['E-mail limpo'].str.replace(r"\s", "")
##
## Normalizando OAB/CPF
##
# Remove espaço em branco
# MELHORIA: Separar e formatar OAB/CPF
df['CPF_OAB limpo'] = df['CPF ou OAB'].apply(sanitizaCPFOAB)

print("CLEAN: usuário")
#
# Sanitiza nomes dos usuário - remove espaços extras, coloca tudo em minúsculas e remove
acentos
#
df.insert(len(df.columns), "Nome limpo", df['Nome'].apply(limpaString))
## Elimina linhas cuja descrição é vazia - sem possibilidade de extrair mais informações
## Busca informações sobre qual gabinete foi demandado
##
## Verifica se gabinete foi mencionado na descrição
print("CLEAN verifica ministro")
df['Ministro'] = df['Descrição do
Atendimento'].apply(Ministros.MinistroClass.id_minister_name)
## Verifica se algum processo/instrumento foi mencionado na descrição
print("CLEAN verifica atendimento")
df['Instrumento'] = df['Descrição do Atendimento'].apply(Processos.match_processo)
# Corrige erros nas descrições de atendimento
print("CLEAN assuntos")
df['Assunto 1 limpo'] = df['Assunto 1'].apply(sanitizaAtendimento)

```

```

df['Assunto 2 limpo'] = df['Assunto 2'].apply(sanitizaAtendimento)
df['Assunto 3 limpo'] = df['Assunto 3'].apply(sanitizaAtendimento)
print("Marked bad data rows: ", cont_ruim)
print("Tempo negativo: ", cont_duração_negativa)
print("Longo demais: ", cont_longo_demais)
print("Data fora dos limites", cont_data_ruim)
print("NaTType: ", cont_natttype)
print("Null names: ", cont_null_name)
print(df.info())
print(df.describe())
print(">>> OUT of Cleaning data")
return df

#
# Rotina principal de conversão de XLS (lento) -> parquet (rápido)
# Nota: chama a rotina de limpeza de dados
# Entrada: Arquivo com todos os dados em formato XLS
# Sairas
# dados limpos, entradas com problemas removidas
# parquet/<<arquivo de entrada>>.parquet
# <<arquivo de entrada>>_clean.xls
# Dados com instrumento jurídico identificado para consulta processual
# <<arquivo de entrada_consulta>>.parquet
# <<arquivo de entrada_consulta>>.xls
#
def convert2parquet(xlsfile, sheet, parquetfile):
    print(">>>> Converting XLS file "+ xlsfile + " to parquet file " + parquetfile)
    print("\nReading excel file "+ xlsfile)
    atendimentos_data_frame: DataFrame = pandas.read_excel(xlsfile, sheet_name=sheet,
dtype={'CPF ou OAB': str, 'E-mail':str, 'Caminho': str})
    print("Head: ", atendimentos_data_frame.head())
    print("Tail", atendimentos_data_frame.tail())
    print("shape: ", atendimentos_data_frame.shape)
    print("Types: ", atendimentos_data_frame.dtypes)
    #
    # Limpa os dados
    #
    print("Cleaning data")
    clean_data_frame=clean_data(atendimentos_data_frame)
    ## Salva dados limpos como parquet e xlsx
    print("Saving as Parquet file " + parquetfile)
    clean_data_frame.to_parquet(parquetfile, engine='pyarrow')
    out_name = re.sub(r"^parquet\\", r"", parquetfile)
    xls_output_name=re.sub(r".parquet$", r"_clean.xlsx",out_name)
    print("XLS output = ",xls_output_name)
    clean_data_frame.to_excel(xls_output_name)
    ## Separa registros com algum processo identificado para fazer consulta processual
    consulta_processual: DataFrame=clean_data_frame[ clean_data_frame['Instrumento'].values !=
None ]
    consulta_output_parquetname = re.sub(r".parquet$", r"_consulta.parquet", out_name)
    consulta_output_xlsname = re.sub(r".parquet$", r"_consulta.xlsx", out_name)
    print("Consultas parquet output = ", consulta_output_parquetname)
    print("Consultas XLS output = ", consulta_output_xlsname)
    consulta_processual.to_parquet(consulta_output_parquetname, engine='pyarrow')
    consulta_processual.to_excel(consulta_output_xlsname)
    return

if __name__ == "__main__":
    for xlsfile in XLS_FILE_LIST:
        parquetfile = re.sub(r"\.xlsx$", PARQUET_FILE_EXTENSION, xlsfile)
        print(XLS_DIRECTORY+"\\ "+xlsfile, " -> ",parquetfile)
        convert2parquet(XLS_DIRECTORY+"\\ "+xlsfile, XLS_FILE_LIST[xlsfile],
PARQUET_DIRECTORY+"\\ "+parquetfile)

    exit()
===== ConsultaProcessualBatch.py
=====
###
### Query em lote a consulta pública do processo e faz normalização do número para o padrão CNJ
e também
### busca o relator para associar o processo ao Gabinete.
###
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

```

```

import os.path
import random
import time
import pandas
from pandas.core.interchange.dataframe_protocol import DataFrame

import Processo
from Config import
DIRETORIO_CONSULTA_PROCESSUAL_BATCH, LISTA_CONSULTA_PROCESSUAL_PARQUET, FILE_DADOS_CONSULTADOS
from QueryPublicRecord import consulta_processo_web
from ParseConsultaProcessual import parse_consulta_html
from Processo import ProcessoClass

limite_de_consultas_por_batch=50000

## BAD Nomes das colunas
## Must match ret_value in parse_consulta_html
col_names = ['Relator',
              'Tribunal de origem',
              'Número único',
              'ParteA',
              'ParteB' ]

# Remove instrumentos/processos sem número
def filter_dict(df: pandas.Series):
    resp=[]
    for lst in df:
        temp = {}
        for k, v in lst.items():
            if v:
                temp.update({k:v})
        resp.append(temp)
    print (len(df),resp)
    return resp
    #return {k: v for k, v in df[0].items() if v!=None}

## Consulta série de processos no dataframe consultas_feitas
## Só processos que ainda não foram consultados
def consulta_batch(consultas_feitas: DataFrame, outfile: str):
    consultas_executadas=0
    #print("Reading data from file " + parquet_file_name)
    #consultas_feitas: pandas.DataFrame = pandas.read_parquet(parquet_file_name,
engine='pyarrow')
    ## Para o caso de multiplos processos em um registro, salvamos o
    ## os dados dos processos adicionais e depois incluimos no dataframe
    #registros_adicionais=[]
    #indice_registros_adicionais=consultas_feitas.last_valid_index()+1
    #Cria colunas para inserir os resultados
    for col in col_names:
        consultas_feitas[col]=""
    # cria pandas.Series filtrando valores nulos da coluna dos processos
    lista_processos= consultas_feitas.loc[:, 'Instrumento'].apply(filter_dict)
    # Usa o indice da coluna "ID", para depois poder reconciliar os resultados
    lista_processos = lista_processos.set_axis(consultas_feitas.loc[:, 'ID'], axis=0)
    #Cria colunas para inserir os resultados
    for col in col_names:
        consultas_feitas[col]=""
    # A serie é indice, lista de processos
    for idx, lst in lista_processos.items():
        lst_idx=0
        # Para cada entrada podem haver vários processos para consultar
        for value in lst:
            # Verifica se há consulta salva
            print("Entrada: ",idx," - a consultar ", value)
            #if value is None or value is "":
            #    print("Bad value, skipping: ", value)
            #    continue
            html_file=DIRETORIO_CONSULTA_PROCESSUAL_BATCH+r"/"+str(idx)+"_"+str(lst_idx)+".html"
            if not os.path.isfile(html_file):
                ## Se consulta não foi salva, faz a consulta
                print("Has to query ",value, " ",html_file)
                ## Evita muita carga no servidor
                time.sleep(0.3 + 1.0*random.random())
                resp_list=consulta_processo_web(value)

```

```

        if(not len(resp_list)):
            print("ERRO >>>>> Query failed")
            continue
        #print("resp_list, check [0]", resp_list)
        f=open(html_file,"w", encoding="utf-8")
        f.write(resp_list[0] )
        f.close()
        consulta_html=resp_list[0]
        consultas_executadas += 1
    else:
        # Lê consulta salva
        print(html_file, " exists")
        ## Skip
        continue
        # indice_registros_adicionais=indice_registros_adicionais+1
        lst_idx=lst_idx+1
    if consultas_executadas > limite_de_consultas_por_batch:
        print(">>>>\n>>>>\nLIMITE DE CONSULTAS EXCEDIDO ", consultas_executadas,
"\n<<<<\n<<<<")
        break
    # inclui registros adicionais no dataframe
    #print("Reg ad:",registros_adicionais)
    #print("indices",consultas_feitas.index.max(),indice_registros_adicionais)
    #temppd=pandas.DataFrame( registros_adicionais,
    #
index=list(range(consultas_feitas.last_valid_index()+1,indice_registros_adicionais))
    #
    #
    #columns=consultas_feitas.columns
    #
    #
    #temppd.dropna(how='all', inplace=True)
    #temppd.dropna(axis='columns',inplace=True)
    #consultas_feitas.dropna(axis='columns',inplace=True)
    #print(temppd)
    #print(temppd.info)
    #print("consultas_feitas ANTES concat")
    #print(consultas_feitas.info)
    #print(consultas_feitas)
    #consultas_feitas.to_excel("temp_consultas_feitas.xlsx")
    #temppd.to_excel("temppd.xlsx")
    #consultas_feitas=pandas.concat([consultas_feitas,temppd])
    print("consultas_feitas:")
    print(consultas_feitas.info)
    print(consultas_feitas)
    # Salva resultado
    consultas_feitas.to_parquet(outfile+".parquet", engine='pyarrow')
    consultas_feitas.to_excel(outfile+".xlsx")

def main():
    # consulta_batch(LISTA_CONSULTA_PROCESSUAL_PARQUET)

if __name__ == "__main__":
    #consulta_batch(LISTA_CONSULTA_PROCESSUAL_PARQUET, "dados_consultados_dez")
    #consulta_batch("falas_min_inst.parquet", "ministros_consultados")
    infile="Dados atualizados Balcão Virtual_consulta.parquet"
    consultas_executadas=0
    print("Reading data from file " + infile)
    in_df: pandas.DataFrame = pandas.read_parquet(infile, engine='pyarrow')
    consulta_batch(in_df, "full_consultados_jan25")
    ===== ParseConsultaProcessual.py
    =====
    #
    # Extrai dados de um arquivo html contendo uma consulta processual
    #
    ### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
    ### Copyright (c) STJ 2025 Licença CC BY-NC
    import re
    import os

    from bs4 import BeautifulSoup
    from Ministros import MinistroClass
    from Config import DIRETORIO_CONSULTA_PROCESSUAL_BATCH
    from Processo import ProcessoClass

    relator_re = re.compile(r"RELATOR\((A\)")

```

```

re_non_tag = re.compile(r">.*?<")
re_sistema_indisponivel=re.compile(r"\s*Sistema indisponível.\s*")

## BAD tem que match Processos.py
token_processo = "Processo"

#def find_relator(req):
#    for lin in req.text.splitlines():
#        if relator_re.search(lin):
#            print(lin)
#            print(match_minister(lin))
#            return match_minister(lin)
#    return None
#
# Faz o parse da consulta html e extrai dados relevantes (partes, numero único, etc.)
# Formato de saída da consulta processual de 23/10/2023
# returns a dictionary with the parsed data
# Return a ProcessoClass object
def parse_consulta_html(qhtml) -> ProcessoClass:
    soup = BeautifulSoup(qhtml, 'html.parser')
    #ret_value = { 'Relator': None,
    #              'Tribunal de origem': None,
    #              'Número único': None,
    #              'Partes': None
    #            }
    ## Verifica se houve erro -> clsMensagemBloco
    ret_value : ProcessoClass= None
    bloco_erro= soup.select(r"div[id=idDivBlocoMensagem] > .clsMensagemLinha") ## Bloco de erro
    if(len(bloco_erro)): # Algum problema, bail com ret_value vazio
        print("ERRO>>>>> ", bloco_erro[0].get_text(), ": ",ret_value)
        return ret_value
    bloco_sistema_indisponivel=soup.select("div[id=idInterfaceVisualAreaBlocoInterno]")
    if(re_sistema_indisponivel.match(bloco_sistema_indisponivel[0].get_text())):
        print("ERRO>>>>> Sistema indisponível", ret_value)
        return ret_value
    #
    # Bloco 1 Tipo de processo, localização, número único, partes & advogados
    #
    bloco1 = soup.select(
        r"span[id=idProcessoDetalhesBloco1] > div[id=idDetalhesPartesAdvogadosProcuradores]
    > .classDivLinhaDetalhes") ## Partes/advogados/numero unico
    #print(bloco1[0].get_text().strip())
    bloco1 = soup.select(r"#idProcessoDetalhesBloco1 >div >span") ## Partes/advogados/numero
    unico
    if(not len(bloco1)):
        print("ERRO>>>>> falha no parse", ret_value)
        return ret_value
    #
    # Partes
    #
    partes = soup.select(
        r"#idProcessoDetalhesBloco1 >div[id=idDetalhesPartesAdvogadosProcuradores] >div") ##
    Partes/advogados/numero unico
    bloco1_partes = []
    #print("Tipo de processo: :", bloco1[1].get_text(),":")
    ret_value = ProcessoClass.nova_instancia(str.strip(bloco1[1].get_text()))
    #print(p)
    #print(">> BLOC01")
    c=0
    for i in partes:
        bloco1_partes.append(i.get_text())
        #print("BLOC01 Partes [",c,"]: ",str.strip(i.get_text()))
        c+=1
    ret_value.incorpora_partes(ProcessoClass.parse_parte(bloco1_partes))
    #print("Parsed: ", parsed)
    #c = 0
    #print(bloco1)
    #print("len(bloco1)-1", len(bloco1) - 1)
    #print("Numero único:", bloco1[len(bloco1) - 1].get_text())
    #
    # Número único
    #
    ret_value.numero_unico= (bloco1[len(bloco1) - 1].get_text())
    # print("<< BLOC01")

```

```

# print(soup.select(r".clsProcessoDetalhesBlocos"))
#
# Bloco2 Relator, ramo do direito, assuntos
# Bloco 2 pode não existir em caso de processos registrados indevidamente, vide
NNNNNNNNNNNNNNNNNNNNNN
#
bloco2 = soup.select(r"span[id=idProcessoDetalhesBloco2] > div > span") ## Relator
if len(bloco2)!= 0:
    #print("bloco2:", bloco2, " len=",len(bloco2))
    #c=0
    #for i in bloco2:
    #    print("Bloco2 ",c,":",i.get_text())
    #    c=c+1
    ## Relator
    #print("Relator: ",bloco2[1].get_text())
    temp=MinistroClass.match_minister(bloco2[1].get_text())
    if(temp != 'none'):
        ret_value.relator= temp
    #c=0
#
# Bloco 3 tribunal de origem, números de origem, volumes
# Bloco 3 pode não existir em caso de processos registrados indevidamente, vide NNNN NNN
NNNN
bloco3 = soup.select(r"#idProcessoDetalhesBloco3 > div > span") ## Partes/advogados/numero
unico
if len(bloco3) !=0:
    #c=0
    #for i in bloco3:
    #    print("Bloco3 ",c,":",i.get_text())
    #    c+=1
    ## Origem e números
    #print("Origem:",bloco3[1].get_text(), bloco3[3].get_text())
    ret_value.tribunal_de_origem= bloco3[1].get_text()
    #p.set_origem(ret_value['Tribunal de origem'])
    # for i in bloco2:
    #    print("II",i.get_text(),"II")
    #    print("***",re.findall(r">.*?\<",i.get_text()),"***")
    # text_bloco2 = [re.findall(r"\>.*?\<",line,flags=0) for line in bloco2]
    # text_bloco2 = list(filter(re_non_tag.findall(line) for line in bloco2])
    # print (text_bloco2)
    # print(soup.select(r"#idProcessoDetalhesBloco3")) ## Tribunal, número de origem
    # print(r.url)
    # print ([line for line in r.text if 'RELATOR' in line])
    #print (ret_value)
## Bloco 4 Fases -> não utilizado aqui
#print(p.como_lista())
return ret_value
#return find_relator(r)

if __name__ == "__main__":
    # Run a test with queries 0, 1 and 4
    for idx in [ 10912, 0,1,4, 102, 751,
                 3480, 5193, 8541, 13870, 17577, 1960 # Sem tribunal de origem
                ]:
        # Verifica se há consulta salva
        # print("Entrada: ",idx," - a consultar ", value)
        html_file = DIRETORIO_CONSULTA_PROCESSUAL_BATCH + r"/" + str(idx) + "_0.html"
        if os.path.isfile(html_file):
            print("Reading file " + html_file)
            f = open(html_file, "r", encoding='utf-8')
            consulta_html = f.read()
            print("Done, now parsing:")
            temp: ProcessoClass = parse_consulta_html(consulta_html)
            if temp is not None:
                print(temp.como_lista())
            f.close()
            print("Done, closed file")
        else:
            ## Se consulta não foi salva, faz a consulta
            print("Didn't find ", html_file, ", skipping")
            ===== ParseConsultasBatch.py
            =====
    ###
    ### Query em lote a consulta pública do processo e faz normalização do número para o padrão CNJ
    e também

```

```

### busca o relator para associar o processo ao Gabinete.
###
### Dados de entrada vem do main.py, que prepara os dados a consultar
### Paulo Tiberio Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import os.path
import pickle
import random
import time
import token
import pandas
from pandas.core.interchange.dataframe_protocol import DataFrame

import Processo
import sanitizaCPFOAB
from Processo import ProcessoClass
from Config import
DIRETORIO_CONSULTA_PROCESSUAL_BATCH, LISTA_CONSULTA_PROCESSUAL_PARQUET, FILE_DADOS_CONSULTADOS
from ParseConsultaProcessual import parse_consulta_html
from smallSampleConsultas import SMALL_SAMPLE_FILE
import Advogado

## BAD Nomes das colunas
## Must match ret_value in parse_consulta_html
col_names = ['Relator',
              'Tribunal de origem',
              'Número único',
              'Partes',
              'Caller Canonico'
              ]

#
# Try to match caller to one of the parties in the process
#
def fixCaller(caller: str, callers_oab: str, consulta: list) -> str:
    if caller is None:
        return None
    for p in consulta:
        #print("fixCaller (" , caller, " oab=", callers_oab, " p=", p)
        oab_clean=sanitizaCPFOAB.sanitizaCPFOAB(callers_oab)
        m: Advogado =Advogado.Advogado.match_advogado(caller, "", oab_clean, p.lista_adv_objs)
        if m is not None:
            return m.nome
    # Didn't find, return None
    return None

# Remove instrumentos/processos sem número
def elimina_entradas_sem_processo(df: pandas.Series):
    resp=[]
    for lst in df:
        temp = {}
        for k, v in lst.items():
            if v:
                temp.update({k:v})
        resp.append(temp)
    #print (len(df),resp)
    return resp
    #return {k: v for k, v in df[0].items() if v!=None}

## Faz parse das consultas LISTA_CONSULTA_PROCESSUAL_PARQUET arquivadas em consultas_html
## e compila dataframe com dados extraídos dos html
def parse_batch(consultas_feitas: DataFrame, token_saida: str):
    #print("Reading data from file " + parquet_file_name)
    #consultas_feitas= pandas.DataFrame = pandas.read_parquet(parquet_file_name,
engine='pyarrow')
    ## Para o caso de multiplos processos em um registro, salvamos o
    ## os dados dos processos adicionais e depois incluimos no dataframe como um array de lista
    de objetos Processo

    #indice_registros_adicionais=consultas_feitas.last_valid_index()+1
    # cria pandas.Series filtrando valores nulos da coluna dos processos
    #lista_processos= consultas_feitas.iloc[:,31].apply(filter_dict)
    lista_processos= consultas_feitas.loc[:, 'Instrumento'].apply(elimina_entradas_sem_processo)
    lista_processos = lista_processos.set_axis(consultas_feitas.loc[:, 'ID'], axis=0)

```

```

print("lista_processos: ", lista_processos.head())
print("consultas_feitas: ", consultas_feitas.head())
#Cria colunas para inserir os resultados
for col in col_names:
    consultas_feitas[col]=""
consultas_feitas['Consultas']=" "
consultas_obarray=dict()
# A serie é indice, lista de processos
for idx, lst in lista_processos.items():
    lst_idx=0
    ### translates idx do the same row on consultas_feitas
    cfi=consultas_feitas.index[consultas_feitas.ID == idx][0]
    # print(idx, cfi, consultas_feitas.at[cfi,'ID'])
    # Para cada atendimento podem haver vários processos para consultar
    resultado_consultas=[]
    resultado_consultas_procobj_list=[]
    tribunais_origem=[]
    relatores=[]
    nups=[]
    instrumentos=[]
    for value in lst:
        # Verifica se há consulta salva
        print("Entrada: ",idx," - a consultar ", value)
        html_file=DIRETORIO_CONSULTA_PROCESSUAL_BATCH+r"/"+str(idx)+"_"+str(lst_idx)+".html"
        if os.path.isfile(html_file):
            # Lê consulta salva
            print(html_file, " exists")
            f=open(html_file,"r", encoding="utf-8")
            consulta_html=f.read()
            f.close()
        else:
            print(html_file, " não existe, skipping")
            continue
        ## Nesse ponto temos a consulta em HTML, então faz-se o parse da
        ## instância com dados extraídos
        proc_obj : ProcessoClass =parse_consulta_html(consulta_html)
        if proc_obj is not None:
            print(proc_obj.começo_lista())
            resultado_consultas.append(proc_obj.começo_lista())
            resultado_consultas_procobj_list.append(proc_obj)
            tribunais_origem.append(proc_obj.tribunal_de_origem)
            if proc_obj.relator is not None:
                relatores.append(proc_obj.relator.NomePreferido)
            else:
                relatores.append(">>> Relator não identificado")
            instrumentos.append(proc_obj.tipo_processo)
            nups.append(proc_obj.numero_unico)
        else:
            resultado_consultas.append(None)
        lst_idx=lst_idx+1
    ## Verifica caller
    caller_fixed=fixCaller(consultas_feitas.at[cfi,'Nome'], consultas_feitas.at[cfi,'CPF ou
OAB'], resultado_consultas_procobj_list)
    if caller_fixed is not None:
        consultas_feitas.at[cfi, 'Caller Canonico'] = caller_fixed
        consultas_feitas.at[cfi, 'Consultas'] = resultado_consultas
        consultas_feitas.at[cfi, 'Relator'] = relatores
        consultas_feitas.at[cfi, 'Instrumento'] = instrumentos
        consultas_feitas.at[cfi, 'Tribunal de origem'] = tribunais_origem
        consultas_feitas.at[cfi, 'Número único'] = nups
        consultas_obarray[cfi]=resultado_consultas_procobj_list
    # Processou todas os processos relativos a um atendimento -> salva nos resultados
    #
    #if cfi > 100:
    #    break
#
# Terminou de parse todas as consultas, listas em ProcessoClass.lista_instancias e
# Advogado.advogados_set estão completas
#
# consultas_feitas.to_excel("temp_consultas_feitas.xlsx")
print("consultas_feitas depois concat:")
print(consultas_feitas.info)
print(consultas_feitas)
# Salva resultado

```

```

# to_parket não funciona com objetos mais complexos. Assim, as partes são salvas como
listas flat
# Para salvar o array de objetos, usamos pickle.
consultas_feitas.to_excel("dados_consultados_"+token_saida+".xlsx")
consultas_feitas.to_parquet("dados_consultados_"+token_saida+".parquet", engine='pyarrow')
with open("consultas_objs"+token_saida+".pkl", "wb") as f:
    pickle.dump(consultas_obarray, f)
    f.close()
    print ("Saved consultas_objs"+token_saida+".pkl pickle file")
with open("advogados_set"+token_saida+".pkl", "wb") as f:
    pickle.dump(Advogado.Advogado.advogados_set, f)
    f.close()
    print ("Saved advogados_set"+token_saida+".pkl pickle file")

if __name__ == "__main__":
    # parse_batch(SMALL_SAMPLE_FILE)
    # parse_batch(LISTA_CONSULTA_PROCESSUAL_PARQUET)
    parquet_file_name="Dados atualizados Balcão Virtual consulta.parquet"
    print("Reading data from file " + parquet_file_name)
    in_df: pandas.DataFrame = pandas.read_parquet(parquet_file_name, engine='pyarrow')
    parse_batch(in_df, "_full_mar25")
    ===== ProcessaPartes.py
    =====
    #
    # Recebe uma instância de processo e examina as partes, separa advogaos e partes
    #
    ### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
    ### Copyright (c) STJ 2025 Licença CC BY-NC
    from Processo import ProcessoClass
    from venv import Advogado

    #
    # Lista é algo tipo:
    # [ {'IMPETRANTE': 'xxxx'}, {'ADVOGADO': 'xxx - NNNNOAB'},
    #   {'IMPETRADO': 'TRIBUNAL DE JUSTIÇA DO ESTADO XXX'}, {'PACIENTE': 'AAAA BBBB (PRESO)'},
    #   {'INTERES.': 'MINISTÉRIO PÚBLICO DO ESTADO XXX'}]
    #
    # Ideia é fazer lista de advogados x partes x chamados BV
    def processa_partes(proc: ProcessoClass):
        if proc is None:
            return None
        if proc.lista_adv_partes is not None:
            for p in proc.lista_adv_partes:
                if "ADVOGADO" in p:
                    Advogado(p["ADVOGADO"])
                else:
                    print(p)
        else:
            return NoneProcesso.py

    #
    # Class com a lista classes processuais conhecidas
    #
    ### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
    ### Copyright (c) STJ 2025 Licença CC BY-NC
    import re
    from mailbox import NotEmptyError
    import string
    import copy
    from Ministros import MinistroClass
    from Advogado import Advogado

    LISTA_TIPOS_PARTES = ['AGRAVANTE', 'AGRAVADO', 'AUTOR',
                          'RÉU', 'IMPETRANTE', 'IMPETRADO', 'RECORRENTE', 'RECORRIDO', 'REQUERENTE', 'REQUERIDO',
                          'SUSCITANTE', 'SUSCITADO', 'PACIENTE', 'INTERES.'].

    # Parse as partes de uma consulata processual em https://processo.stj.jus.br/processo/pesquisa/
    # Resp:
    # Formato:
    # 1 ou mais Partes:
    # -> RECORRENTE
    # -> <<NOME>> | ADVOGADO: <<NOME>> - <<OAB>>
    # -> RECORRIDO
    # -> <<NOME>> | ADVOGADO: <<NOME>> - <<OAB>>

```

```

# Retorna dict

class ProcessoClass():
    lista_instancias = []
    def __init__(self, nome_por_extenso: str, tipo_processo: str, lista_nome_partes: list,
codigo:str, alias_list: list):
        ## Shared init, não devem ser mudadas (compartilhadas entre instâncias)
        self.por_extenso = nome_por_extenso
        self.tipo_processo = tipo_processo # Como aparece na consulta processual
        self.nome_partes = lista_nome_partes # Agravante/Agravado, Impetrante/Impetrado,
Interessado, etc.
        self.codigoCanonicos = codigo
        self.alias_list = alias_list
        # Lista partes é lista com dicionários com nome da parte, advogados, pacientes.
interessados, etc.
        # As variáveis abaixo são inicializadas depois de nova_instância, que clona o objeto
        # inicializado acima com shallow copy (se inicializar aqui, vai dar merda [ instancias
vão compartilhar dados])
        self.lista_partes = None
        # Lista completa de partes e advogados, como aparece na consulta processual
        self.lista_adv_partes = None
        # Lista dos Advogados associados ao processo
        self.lista_adv_objs = None
        self.numero_unico = None
        self.turma = None
        self.relator : MinistroClass = None # depois insere o relator
        self.tribunal_de_origem= None
        self.ramo_do_direito= None
        ## May be not good - there can be race conditions on the class variable lista_classes
        ProcessoClass.lista_instancias.append(self)

    def __str__(self):
        return (f"Classe processual {self.por_extenso}, sigla {self.codigoCanonicos}, alias:
[ {self.alias_list} ]" )

    def como_lista(self):
        #return [ self.tipo_processo, self.numero_unico, str(self.relator),
self.tribunal_de_origem, self.lista_partes, self.lista_adv_partes ]
        return [self.tipo_processo, self.numero_unico, str(self.relator),
self.tribunal_de_origem ]

    # Cria uma nova instância a partir dos processos pré-inicializados. Se não acha algo
inicializado retorna None.
    @classmethod
    def nova_instancia(cls, tipo_de_processo: string) :
        for i in cls.lista_instancias:
            if i.tipo_processo == tipo_de_processo:
                return copy.copy(i)
        return None ## Talvez alocar um novo??

    #def set_nup(self, NUP: string):
    #    self.numero_unico=NUP

    #def set_relator(self, m: MinistroClass):
    #    self.relator = m

    #def set_origem(self, t: string):
    #    self.tribunal_de_origem=t

    #def get_partes(self):
    #    return self.lista_partes

    # Incorpora lista de partes no objeto
    # entrada lista de dictionaries com parte/advogados : <<dados>>
    # Insere nomes das partes em self.lista_partes (conjunto, sem repetições)
    # Salva cópia da lista original em self.lista_adv_partes (com repetições, inclui advogados)
    # Sai da [ parte , parte, parte ]
    #
    def incorpora_partes(self, in_lst: list) -> list:
        #print("incorpora_partes in_lst: ",in_lst)
        self.lista_partes=[]
        self.lista_adv_partes = in_lst.copy() # Lista advogados -> Texto
        self.lista_adv_objs = set() # Conjunto de objetos Advogado

```

```

# Usamos set para remover partes duplicadas
resultado=set()
for p in in_lst:
    #print("inc: ", p, p.keys(), list(p))
    # Se não é advogado ou promotor, é parte
    if list(p)[0] in ["ADVOGADO", "PROMOTOR"]:
        if(list(p)[0] == "PROMOTOR"):
            #print("PROMOTOR", in_lst)
            exit(0)
        #print("Advogado: ", list(p.values())[0])
        adv=Advogado.registra(p["ADVOGADO"])
        self.lista_adv_objs.add(adv)
    else: # Parte
        #print("adicionando parte: ", list(p.values())[0])
        resultado.add(list(p.values())[0])
self.lista_partes=list(resultado)
print(self.lista_partes)
return self.lista_partes

#def get_adv_partes(self):
#    return self.lista_adv_partes

@classmethod
def nome_por_extenso(cls, alias: str) -> str:
    for i in cls.lista_instancias:
        for a in i.aliases:
            if alias == a:
                return i.por_extenso
    return ""

# parte [ { 'nome': agravante | agravado | recorrente | recorrido | réu|autor|"outro
nome" ,
#         'partes': [ lista partes ]
#         'advogados': [lista advogados da parte ]
# [
# [ partea [paciente/etc. ] [lista advogados] ], [parte b ...], interes..
# [ ] ...
# ]
@classmethod
def parse_parte(cls, str_in: list) -> list :
    result =[]
    for txt in str_in:
        #print("Matching", txt)
        #m = re.match(r"^(\\S+) :..([A-Z ]+)..$", txt, re.DOTALL)
        m = re.match(r"^(\\S+) *:(.*)", txt, flags=re.DOTALL)
        #m = re.match(r"^(\\s(\\S+)\\s*:.*", txt)
        if m is None:
            # Tenta match em duas palavras com espaço entre elas << MELHORAR>>
            # print("No match ", txt)
            m = re.match(r"^(\\S+ \\S+) *:(.*)", txt, flags=re.DOTALL)
        if m:
            #print("Tipo",m.group(1))
            #print("Nome",str.strip(m.group(2)))
            result.append({ str.strip(m.group(1)): str.strip(m.group(2)) })
        else:
            print("No match ", txt)
    #print(result)
    return result

# Outras Partes: INTERES. PROCURADOR REQSTE
ProcessoClass( "AÇÃO PENAL", "AÇÃO PENAL", ["AUTOR", "RÉU"], 'APn', ['APn', 'APN'] )
ProcessoClass( 'Ação Rescisória', "AÇÃO RESCISÓRIA",["AUTOR", "RÉU"], "AR", ["AR"])
ProcessoClass( 'Agravado em Recurso Especial', 'AGRAVO EM RECURSO ESPECIAL', ['AGRAVANTE',
'AGRAVADO'], 'AREsp', 'AREsp.', 'Aresp', 'AREsp', 'aresp', 'AREsp', 'ARESP', 'AGRAVO
EM RECURSO ESPECIAL', 'Agravado em Recurso Especial', 'AREap'])
ProcessoClass( "Conflito de Competência", 'CONFLITO DE COMPETÊNCIA', ["SUSCITANTE",
"SUSCITADO"], 'CC', ['CC'])
ProcessoClass( "EMBARGOS DE DIVERGÊNCIA EM RESP", "EMBARGOS DE DIVERGÊNCIA EM RESP",
['EMBARGANTE', 'EMBARGADO'], 'EResp', ['EResp', 'Eresp'])
ProcessoClass( "EXECUÇÃO EM MANDADO DE SEGURANÇA", "EXECUÇÃO EM MANDADO DE SEGURANÇA",
['EXEQUENTE', "EXECUTADO"], 'ExeMS', ['ExeMS', 'EXeMS'])
ProcessoClass( 'Habeas Corpus', 'HABEAS CORPUS', ['IMPETRANTE', 'IMPETRADO', 'PACIENTE' ], 'HC',
['HC', 'Hc', 'hc'])
ProcessoClass( "PETIÇÃO", "PETIÇÃO", ["REQUERENTE", "REQUERIDO"], 'Pet', ['Pet', 'PET', 'PET'] )

```

```

ProcessoClass( "PRECATÓRIO","PRECATÓRIO",["REQUERENTE", "REQUERIDO", "REQSTE"],'Prc', ['Prc',
'PRC'])
ProcessoClass( 'Recurso em Habeas Corpus', 'RECURSO EM HABEAS CORPUS', ['RECORRENTE',
'RECORRIDO'], 'RHC', ['RHC'])
ProcessoClass( 'Recurso Especial', 'RECURSO ESPECIAL', ['RECORRENTE', 'RECORRIDO'], 'Resp',
['Resp', 'Resp', 'RESP', 'RECURSO ESPECIAL', 'resp', 'REspS', 'Recurso Especial'])
ProcessoClass( "Recurso em mandato de segurança", 'RECURSO EM MANDADO DE SEGURANÇA',
['RECORRENTE', 'RECORRIDO'], 'RMS', ['RMS'])
ProcessoClass( "REQUISIÇÃO DE PEQUENO VALOR", "REQUISIÇÃO DE PEQUENO VALOR", ["REQUERENTE",
"REQUERIDO"], 'RPV', ['RPV'])
ProcessoClass( "Restituição de Coisas Apreendidas", 'RESTITUIÇÃO DE COISAS
APREENDIDAS', ["REQUERENTE", "REQUERIDO"], 'ReCoAp', ['ReCoAp'])
ProcessoClass( "Homologação de Decisão Estrangeira", "HOMOLOGAÇÃO DE DECISÃO ESTRANGEIRA",
["REQUERENTE", "REQUERIDO"], 'HDE', ['HDE', 'HDe'])
ProcessoClass( "TUTELA ANTECIPADA ANTECEDENTE", "TUTELA ANTECIPADA ANTECEDENTE", ["REQUERENTE",
"REQUERIDO"], 'TutAntAnt', ['TutAntAnt'])
ProcessoClass( "TUTELA CAUTELAR ANTECEDENTE", "TUTELA CAUTELAR ANTECEDENTE", ["REQUERENTE",
"REQUERIDO"], 'TutCautAnt', ['TutCautAnt'])
ProcessoClass( "EMBARGOS DE DIVERGÊNCIA EM AGRAVO EM RECURSO ESPECIAL", "EMBARGOS DE DIVERGÊNCIA
EM AGRAVO EM RECURSO ESPECIAL", ['EMBARGANTE', 'EMBARGADO'], 'EAREsp', ['EAREsp', 'EAREsp'])
ProcessoClass( "EXECUÇÃO EM AÇÃO RESCISÓRIA", "EXECUÇÃO EM AÇÃO
RESCISÓRIA", ["EXEQUENTE", "EXECUTADO"], 'ExeAR', ['ExeAR', 'AR'])
ProcessoClass( "RECLAMAÇÃO", "RECLAMAÇÃO", ["RECLAMANTE", "RECLAMADO"], 'Rcl', ['Rcl', 'RCL'])
ProcessoClass( "Mandato de Segurança", "MANDADO DE SEGURANÇA", ["IMPETRANTE", "IMPETRADO"],
'MS', ['MS'])
ProcessoClass( "PEDIDO DE UNIFORMIZAÇÃO DE INTERPRETAÇÃO DE LEI", "PEDIDO DE UNIFORMIZAÇÃO DE
INTERPRETAÇÃO DE LEI", ["REQUERENTE", "REQUERIDO"], 'PUIL', ['PUIL'])
ProcessoClass( "AGRAVO DE INSTRUMENTO", "AGRAVO DE INSTRUMENTO", ['AGRAVANTE', 'AGRAVADO'], 'Ag',
['AG', 'Ag'])
ProcessoClass( "Carta Rogatória", "CARTA ROGATÓRIA", ["JUSROGANTE", "A.CENTRAL", "PARTE"], 'CR',
['CR'])
ProcessoClass( "SENTENÇA ESTRANGEIRA", "SENTENÇA ESTRANGEIRA", ["REQUERENTE", "REQUERIDO"],
'ExeSE', ['ExeSE'])
ProcessoClass( "SUSPENSÃO DE LIMINAR E DE SENTENÇA", "SUSPENSÃO DE LIMINAR E DE
SENTENÇA", ["REQUERENTE", "REQUERIDO"], 'SLS', ['SLS'])
ProcessoClass( "SENTENÇA ESTRANGEIRA CONTESTADA", "SENTENÇA ESTRANGEIRA
CONTESTADA", ["REQUERENTE", "REQUERIDO"], 'SEC', ['SEC'])
ProcessoClass( "HABEAS DATA", "HABEAS DATA", ["IMPETRANTE", "IMPETRADO"], 'HD', 'HD')
ProcessoClass( "SUSPENSÃO DE SEGURANÇA", "SUSPENSÃO DE SEGURANÇA", ["REQUERENTE", "REQUERIDO"],
'SS', ['SS'])
ProcessoClass( "PEDIDO DE TUTELA PROVISÓRIA", "PEDIDO DE TUTELA PROVISÓRIA", ["REQUERENTE",
"REQUERIDO"], 'TP', ['TP'])
ProcessoClass( "EMBARGOS À EXECUÇÃO EM MANDADO DE SEGURANÇA", "EMBARGOS À EXECUÇÃO EM MANDADO DE
SEGURANÇA", ['EMBARGANTE', 'EMBARGADO'], 'EmbExeMS', ['EmbExeMS'])
ProcessoClass( "REVISÃO CRIMINAL", "REVISÃO CRIMINAL", ["REQUERENTE", "REQUERIDO"], 'RvCr',
['RvCr', 'RVCR'])
ProcessoClass( "EMBARGOS DE TERCEIRO", "EMBARGOS DE TERCEIRO", ['EMBARGANTE', 'EMBARGADO'], 'ET',
['ET'])
ProcessoClass( "EMBARGOS À EXECUÇÃO EM AÇÃO RESCISÓRIA", "EMBARGOS À EXECUÇÃO EM AÇÃO
RESCISÓRIA", ['EMBARGANTE', 'EMBARGADO'], 'EmbExeAr', ['EmbExeAr'])
ProcessoClass( "SISBAJUD", "SISBAJUD", ["AUTOR", "SISBAJUD", ["SISBAJUD"]])
ProcessoClass( "MEDIDA CAUTELAR", "MEDIDA CAUTELAR", ["REQUERENTE", "REQUERIDO"], 'MC', ["MC"])
ProcessoClass( "EMBARGOS DO ACUSADO", "EMBARGOS DO ACUSADO", ['EMBARGANTE', 'EMBARGADO'], 'EmbAc',
['EmbAc'])
ProcessoClass( "INTERPELAÇÃO JUDICIAL", "INTERPELAÇÃO JUDICIAL", ["INTPTE", "INTPDO",
"INTERPELANTE", "INTERPELADO"], 'IJ', ['IJ'])
### Verificar partes de quebra de sigilo
ProcessoClass( "Pedido de Quebra de Sigilo de Dados e/ou Telefônico", "QUERA DE SIGILO",
["REQUERENTE", "REQUERIDO"], 'QuebSig', ['QuebSig'])
if __name__ == "__main__":
    test_list=[ '\nAGRAVANTE :\n\nMINISTÉRIO PÚBLICO FEDERAL\n\n', '\nAGRAVANTE :\n\nAAA B
CCC\n\n',
                '\nADVOGADO:\n\nFFFF KKKKK - EENNNNNN - EENNNNNN\n\n',
                '\nAGRAVADO :\n\nMINISTÉRIO PÚBLICO FEDERAL\n\n',
                '\nAGRAVADO :\n\nAAA B CCC\n\n', '\nADVOGADO:\n\nFFFF KKKKK - EENNNNNN\n\n',
                '\nAGRAVADO :\n\n0000 PPPP\n\n',
                '\nADVOGADO:\n\nAAAAA CCCCCC MMMM - EENNNNNN\n\n',
                '\nAGRAVADO :\n\nINSTITUTO JJJJJJJJ \n\n'
            ]
    p = ProcessoClass.nova_instancia('HABEAS CORPUS')
    p.incorpora_partes(ProcessoClass.parse_parte(test_list))
    print(str(ProcessoClass.nova_instancia('HABEAS CORPUS')))
    exit(0)

```

===== Processos.py =====


```

        return None
    if not isinstance(strlst, str):
        return None
    last_remainder = ''
    remainder = strlst
    match_array = []
    while (remainder != last_remainder):
        ## Primeiro procuramos pelo instrumento jurídico - AResp, HC, etc.
        last_remainder = remainder
        #print("remainder 170:", remainder)
        match_instrumento_juridico = tokens_re.match(remainder)
        if match_instrumento_juridico is not None:
            #print("IJ 173:", match_instrumento_juridico.groups(), "Remainder:", remainder)
            # Achou instrumento juridico, agora tentar achar o número do instrumento ou processo
            remainder = match_instrumento_juridico.group('after')
            #print("IJ 176:", match_instrumento_juridico.groups(), "Remainder:", remainder)
            match_ij_number = numero_ij_re.match(remainder)
            if match_ij_number is not None:
                ## Achou numero - all is good, acrescenta na lista de resposta, inclui registro
                do STJ se existir!
                #print("IJ 180: ", match_instrumento_juridico.groups(), "-> ", match_ij_number.groups())
                reg_stj = re_numero_registro_stj.match(match_ij_number.group(2))
                if reg_stj is not None:
                    match_array.append({token_registro_stj: normaliza_registro_stj(reg_stj),
                                         normaliza_token(match_instrumento_juridico.group(1))})
                match_ij_number.group(2))
            else:
                #print(">>> IJ Groups", match_ij_number.groups())
                if len(match_ij_number.group(2))!=25 and len(match_ij_number.group(2))!=20:
                    match_array.append({normaliza_token(match_instrumento_juridico.group(1)):
                                         match_ij_number.group(2)})
            else:
                ## numero é NUP, usa nome_coluna_contagem processo
                print(">>> NUP")
                match_array.append({token_processo: match_ij_number.group(2)})
                ## na próxima iteração, vamos ver se tem mais algum instrumento jurídico no resto
                da string
                remainder = match_ij_number.group('after')
                # No match instrumento juridico -> tenta ver se tem número de processo em toda a string
                recebida
                # print("match_processo array:", match_array)
                if len(match_array) == 0:
                    ls_number = numero_processo_re.search(strlst)
                    ## Achou numero "solto", assume-se que é numero de processo
                    if (ls_number):
                        # Verifica se é um número de registro do STJ
                        reg_stj = re_numero_registro_stj.match(ls_number.group(1))
                        if (reg_stj):
                            match_array.append({token_registro_stj: normaliza_registro_stj(reg_stj)})
                        else: ## Não é registro do STJ, assume que numero unico
                            #print("Assumindo número de processo: ")
                            match_array.append({token_processo: ls_number.group(1)})
                        return match_array
                    else:
                        return None
                else:
                    return match_array

    ## Test
    PTMB_test_strings = [

        "AREsp nº NNNNNNN/KK - Queria despachar com a Ministra XXXXX YYYYYY. Passei as orientações
        revelantes."
        , "respNNNNNNNN - Queria despachar com o Ministro ZZZZZ WWWWWW; Passei as informações."

    ]

    ### Rotinas de teste, comentar
    if __name__ == "__main__":
        # Run a test
        tst_re = re.compile('.*(' + todos_tokens + ')?(P<after>.*)', re.DOTALL)
        for ts in PTMB_test_strings:

```

```

print("TEST STR:", "\"" + ts + "\"")
m = match_processo(ts)
if m:
    # print(tst_re.match(ts).groups(), ts)
    print(m)
else:
    print("NO MATCH")
===== QueryPublicRecord.py
=====
###
### Query a consulta pública do processo e faz normalização do número para o padrão CNJ e também
### busca o relator para associar o processo ao Gabinete.
###
### PTMB 8/2023
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import re
import requests
from bs4 import BeautifulSoup
# import Ministros
from Config import DIRETORIO_CONSULTA_PROCESSUAL_BATCH

import os
## BAD should import
from Processos import token_processo, token_registro_stj
#token_processo = 'Processo'
#token_registro_stj = 'RegistroSTJ'

URL_PUBLIC_QUERY = r"https://processo.stj.jus.br/processo/pesquisa/"
URL_TIPO_PESQUISA = "tipoPesquisa"
URL_TIPO_PESQUISA_GENERIC = "tipoPesquisaGenerica"
URL_TIPO_PESQUISA_UNICO = "tipoPesquisaNumeroUnico"
URL_TIPO_PESQUISA_REGISTRO = "tipoPesquisaNumeroRegistro"
URL_SEARCH_TOKEN = "termo"
URL_APP_KEY = "aplicacao"
URL_APP_VALUE = "processos.ea"
QUERY_TIMEOUT = 120 # In seconds

re_processo_com_origem=re.compile( r'(?P<numB>[0-9]+)(?:\s*)/(?:\s*)[A-Z][A-Z]\s*[0-9]+(?:\s*)/(?:\s*)[0-9]+--[0-9]+\s*'+"|"+
                                     r'(?P<numA>[0-9]+)\s*/\s*[A-Z][A-Z]\s*',
                                     re.IGNORECASE | re.DOTALL)

re_dot=re.compile(r"\.")
### check https://requests.readthedocs.io/en/latest/user/quickstart/#passing-parameters-in-urls

## Arruma informação sobre o processo para fazer ficar aceitável a consulta processual
def fix_instrumento(inst):
    # Remove ponto separando dígitos
    inst=re_dot.sub("",inst)
    # verifica o caso de 202104093468 -> 2021/0409346-8
    m=re_processo_com_origem.match(inst)
    ## Return anything that matched or the original number
    if(m):
        return next((s for s in m.groups() if s is not None), inst)
    else:
        return inst

### Faz consulta processual e retorna uma lista de textos html
### idx é o índice no dataframe dos atendimento - é único
### dict é o dicionário com números de processo ou instrumento jurídico + numero
### pode haver mais de um processo associado a um atendimento
def consulta_processo_web(dict):
    response=[] # Stars with the empty list
    # Se tem núme STJ, usa este
    if token_registro_stj in dict:
        numero=dict[token_registro_stj]
        payload = [(URL_TIPO_PESQUISA, URL_TIPO_PESQUISA_REGISTRO),
                    (URL_SEARCH_TOKEN, numero),
                    (URL_APP_KEY, URL_APP_VALUE) ]
    else:

```

```

for instrumento, numero in dict.items():
    if numero is None:
        continue # Skip empty instruments
    ## Se não tem instrumento (AREsp, etc.) consulta numero do processo
    print(instrumento, numero)
    match instrumento:
        case 'Processo':
            if len(numero)==20 or len(numero)==25:
                # Se tem 20 digitos, deve ser número unico
                # 25 digitos é no formato com .-: 0374937-45.2023.3.00.0000
                #print("::: Processo", numero, len(numero))
                payload = [(URL_TIPO_PESQUISA, URL_TIPO_PESQUISA_UNICO),
                           (URL_SEARCH_TOKEN, numero),
                           (URL_APP_KEY, URL_APP_VALUE)]
            else:
                payload = [(URL_TIPO_PESQUISA, URL_TIPO_PESQUISA_GENERIC),
                           (URL_SEARCH_TOKEN, numero),
                           (URL_APP_KEY, URL_APP_VALUE)]
        case _:
            # https: // processo.stj.jus.br / processo / pesquisa /?tipoPesquisa =
            tipoPesquisaGenerica & termo = RMS + 66283 & aplicacao = processos
            # Se tem instrumento e número, usa os dois dados para pesquisa
            fnumero=fix_instrumento(numero)
            print(instrumento, "Nro orig:", numero, "fixed: ", fnumero)
            payload = [(URL_TIPO_PESQUISA, URL_TIPO_PESQUISA_GENERIC),
                       (URL_SEARCH_TOKEN, instrumento + " " + fnumero),
                       (URL_APP_KEY, URL_APP_VALUE)]
            print(URL_PUBLIC_QUERY, payload)
            # r = requests.get(URL_PUBLIC_QUERY, params=payload)
            req_headers = {'User-Agent': r'Mozilla/5.0 (Windows NT 10.0; rv:109.0) Gecko/20100101
Firefox/118.0',
                          'Accept':
r"text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q = 0.8"
            }
            req_cookies = {
                'JSESSIONID': r'T0F0uBnboBd5Pwp3gfngAx5YTqt0_ceef7qFvV6.svlp-jboss-06',
                'cf_clearance' : r'FGe3Qit3I1nrmFRa1_95pa_3DLW3CV6XoH.sd_PR17w-1697403274-0-1-
4f5c0d29.3ea68550.185d52c9-0.2.1697403274',
                'STJPROCESSOSACESSADOS':
r'202203470413|AREsp|2240277|20231015175438|MA~202101212634|RMS|66283|20231008230401|AP~20200178
0916|RMS|64026|20231008225659|MA~'
            }
            session = requests.Session()
            r = session.get("https://processo.stj.jus.br/processo/pesquisa/", params=payload,
                           headers=req_headers, cookies=req_cookies, timeout= QUERY_TIMEOUT)
            if(r.status_code!=requests.codes.ok):
                print("Query failed")
                return response
            # print(r.text)
            response.append(r.text)
            return response

===== Ministros.py =====
#
# Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
#
# Dictionary com os ministros do STJ hoje (Junho 2023) e tokens - substrings para indentificar
os
# ministros nos registros inclui grafia alternativa, em geral sem acentos, para melhorar a
chance de
# um match positivo
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import math
import re
from enum import Enum

class TIPO_MINISTRO(Enum):

    PRESIDENCIA = "Presidência"
    VICE PRESIDENCIA = "Vice-presidência"
    MINISTRO = "Ministro"

```

```

MINISTRA = "Ministra"
DESEMBARGADOR_CONVOCADADO = "Desembargador Convocado"
DESEMBARGADORA_CONVOCADADA = "Desembargadora Convocada"

class MinistroClass:
    """Classe para armazenar informações sobre ministros, mantém lista de instâncias"""

    lista_ministros=[]
    def __init__(self, nome, nome_preferido, codigo, alias_array, tipo, nome_anon):
        self.NomeCompleto=nome
        self.NomePreferido=nome_preferido
        self.alias=alias_array
        self.codigo_CATJD=codigo
        self.tipo=tipo
        self.anon=nome_anon # Código de anonimização do Ministro
        ## May be not good - there can be race conditions on the class variable
lista_ministros
        MinistroClass.lista_ministros.append(self)
    def __str__(self):
        #return(f"Ministro {self.NomePreferido}, Código={self.codigo_CATJD}, {self.alias}")
        return(f"Ministro {self.NomePreferido}")

    @classmethod
    def id_minister_name(cls, strlst):
        m=MinistroClass.match_minister(strlst)
        if(m is None):
            return "none"
        else:
            return m.NomePreferido

    @classmethod
    def match_minister(cls, strlst):
        found_m="none"
        found_count: int=0
        last_count=0
        last_found=None

        # Loop over ministers
        #print(strlst)
        if strlst is None:
            return None
        if not isinstance(strlst, str):
            return None

        #print(len(cls.lista_ministros))
        for m in cls.lista_ministros:
            #print(m)
            found_count = 0
            # loop over the minister name's components
            for token in m.alias:
                # print("match_minister: ", m, strlst, token)
                if re.search(token, strlst, re.IGNORECASE) is not None:
                    found_count += 1
                    # print("match ",strlst," ",token)
            if(found_count > last_count):
                last_count=found_count
                last_found=m

            #print(m, found_count, last_count)
            #if(last_count>0):
            #print("\nmatch_minister: " + last_found)
            # print(strlst, "\nmatch_minister: " + last_found)
        #print("Found:",last_found)
        return last_found

#
# Inicializa a lista de ministras, ministros e desembargadores convocados.
#
MinistroClass("Presidência","Presidência","ARP", ['Presidente', 'Presidência', 'Presidencia',
'Presid', 'PRESIDENTE DO STJ'], TIPO_MINISTRO.PRESIDENCIA, "Presidência")
MinistroClass("Vice-Presidência","Vice-Presidência","GVP",['vice', 'vice-presidencia', 'vice-
presidência'], TIPO_MINISTRO.VICE_PRESIDENCIA, "Vice-Presidência")
MinistroClass("Antonio Carlos Ferreira", "Antonio Carlos Ferreira","GMACF",['Antonio',
'Antônio','Carlos', 'Ferreira'], TIPO_MINISTRO.MINISTRO, "Ministro01")

```

```

MinistroClass("Antonio Herman de Vasconcelos e Benjamin","Herman Benjamin","GMHB", ['Antonio',
'Antônio','Herman', 'Vasconcelos', 'Benjamin'], TIPO_MINISTRO.MINISTRO, "Ministro02") # Sem 'de'
e 'e'
MinistroClass("Antonio Saldanha Palheiro","Antonio Saldanha Palheiro","GMASP", ['Antonio',
'Antônio','Saldanha', 'Palheiro'], TIPO_MINISTRO.MINISTRO, "Ministro03" )
MinistroClass("Assuete Dumont Reis Magalhães", "Assuete Magalhães", "GMAM", ['Assuete',
'Dumont', 'Reis', 'Magalhães', 'Magalhaes'], TIPO_MINISTRO.MINISTRA, "Ministro04" ) # Grafia sem
acentos de Magalhães
MinistroClass("Benedito Gonçalves", "Benedito Gonçalves", "GMBG", ['Benedito', 'Gonçalves',
'Goncalves'], TIPO_MINISTRO.MINISTRO, "Ministro05")
MinistroClass("Daniela Rodrigues Teixeira", "Daniela Teixeira", "GMDT", ['Daniela', 'Teixeira'],
TIPO_MINISTRO.MINISTRA, "Ministro06")
MinistroClass("Fátima Nancy Andrighi", "Nancy Andrighi", "GMFN", ['Fátima', 'Nancy', 'Andrighi',
'Fatima'], TIPO_MINISTRO.MINISTRA, "Ministro07") # Grafia sem acentos de Fatima
MinistroClass("Felix Fischer", "Felix Fischer", "GMFF", ['Felix', 'Félix', 'Fischer'],
TIPO_MINISTRO.MINISTRO, "Ministro08")
MinistroClass("Francisco Cândido de Melo Falcão Neto", "Francisco Falcão", "GMFC",
['Francisco', 'Cândido', 'Candido', 'Melo', 'Falcão', 'Neto', 'Falcao'], TIPO_MINISTRO.MINISTRO,
"Ministro09") # Grafia sem acentos de Falcão e sem 'de'
MinistroClass("Geraldo Og Nicéas Marques Fernandes", "Og Fernandes", "GMOF", [r'(^\\W)Og',
'Nicéas', 'Og Fernandes', 'Niceas'], TIPO_MINISTRO.MINISTRO, "Ministro10") # Grafia sem acentos
de Nicéas
#MinistroClass("Geraldo Og Nicéas Marques Fernandes", "Og Fernandes", "GMOF", ['Geraldo',
r'(^\\W)Og', 'Nicéas', 'Marques', 'Fernandes', 'Niceas'], TIPO_MINISTRO.MINISTRO, "Ministro10")
# Grafia sem acentos de Nicéas
MinistroClass("Humberto Eustáquio Soares Martins", "Humberto Martins", "GMHE", ['Humberto',
'Eustáquio', 'Soares', 'Martins', 'Eustaquio'], TIPO_MINISTRO.MINISTRO, "Ministro11") # Grafia
sem acentos de Eustáquio
MinistroClass("João Otávio de Noronha", "João Otávio de Noronha", "GMJON", ['João', 'Otávio',
'Noronha', 'Joao', 'Otavio'], TIPO_MINISTRO.MINISTRO, "Ministro12") # Grafia sem acentos de
João e Otávio e sem 'de'
MinistroClass("Joel Ilan Paciornik", "Joel Ilan Paciornik", "GMJP", ['Joel', 'Ilan',
'Paciornik'], TIPO_MINISTRO.MINISTRO, "Ministro13")
MinistroClass("José Afrânio Vilela", "Afrânio Vilela", "GMAV",
['José', 'Jose', 'Afranio', 'Afrânio', 'Vilela'], TIPO_MINISTRO.MINISTRO, "Ministro14")
MinistroClass("Luis Felipe Salomão", "Luis Felipe Salomão", "GMLFS", ['Luis', 'Felipe',
'Salomão', 'Salomao'], TIPO_MINISTRO.MINISTRO, "Ministro15") # Grafia sem acentos de Salomão
MinistroClass("Luiz Alberto Gurgel de Faria", "Gurgel de Faria", "GMLAG", ['Luiz', 'Alberto',
'Gurgel', 'Faria'], TIPO_MINISTRO.MINISTRO, "Ministro16") # Sem o 'de'
MinistroClass("Marcelo Navarro Ribeiro Dantas", "Ribeiro Dantas", "GMRD", ['Marcelo',
'Navarro', 'Ribeiro', 'Dantas'], TIPO_MINISTRO.MINISTRO, "Ministro17")
MinistroClass("Marco Aurélio Bellizze Oliveira", "Marco Aurélio Bellizze", "GMMAB", ['Marco',
'Aurélío', 'Bellizze', 'Oliveira', 'Aurelio'], TIPO_MINISTRO.MINISTRO, "Ministro18") # Grafia
sem acentos de Aurélio
MinistroClass("Marco Aurélio Gastaldi Buzzi", "Marco Buzzi", "GMMB", ['Marco', 'Aurélío',
'Gastaldi', 'Buzzi', 'Aurelio'], TIPO_MINISTRO.MINISTRO, "Ministro19") # Grafia sem acentos de
Aurélio
MinistroClass("Maria Isabel Diniz Gallotti Rodrigues", "Isabel Gallotti", "GMIG", ['Maria',
'Isabel', 'Diniz', 'Gallotti', 'Rodrigues'], TIPO_MINISTRO.MINISTRA, "Ministro20")
MinistroClass("Maria Thereza Rocha de Assis Moura", "Maria Thereza de Assis Moura", "GMMT",
['Maria', 'Thereza', 'Rocha', 'Assis', 'Moura'], TIPO_MINISTRO.MINISTRA, "Ministro21") # Sem o
'de'
MinistroClass("Mauro Luiz Campbell Marques", "Mauro Campbell Marques", "GMMCM", ['Mauro',
'Luiz', 'Campbell', 'Marques'], TIPO_MINISTRO.MINISTRO, "Ministro22")
MinistroClass("Messod Azulay Neto", "Messod Azulay Neto", "GMMAN", ['Messod', 'Azulay', 'Neto'],
TIPO_MINISTRO.MINISTRO, "Ministro23")
MinistroClass("Paulo de Tarso Vieira Sanseverino", "Paulo de Tarso Sanseverino", "GMPTS",
['Paulo', 'Tarso', 'Vieira', 'Sanseverino'], TIPO_MINISTRO.MINISTRO, "Ministro24")
MinistroClass("Paulo Dias de Moura Ribeiro", "Moura Ribeiro", "GMMMR", ['Paulo', 'Dias', 'Moura',
'Ribeiro'], TIPO_MINISTRO.MINISTRO, "Ministro25") # Sem o 'de'
MinistroClass("Paulo Sérgio Domingues", "Paulo Sérgio Domingues", "GMPD", ['Paulo', 'Sérgio',
'Domingues', 'Sergio'], TIPO_MINISTRO.MINISTRO, "Ministro26")
MinistroClass("Raul Araújo Filho", "Raul Araújo", "GMAF", ['Raul', 'Araújo', 'Filho', 'Araujo'],
TIPO_MINISTRO.MINISTRO, "Ministro27")
MinistroClass("Regina Helena Costa", "Regina Helena Costa", "GMRHC", ['Regina', 'Helena',
'Costa'], TIPO_MINISTRO.MINISTRA, "Ministro28")
MinistroClass("Reynaldo Soares da Fonseca", "Reynaldo Soares da Fonseca", "GMRSF", ['Reynaldo',
'Soares', 'Fonseca', 'Rynaldo', 'Reinaldo'], TIPO_MINISTRO.MINISTRO, "Ministro29") # Sem o 'da'
e com erro do ortografia encontrado nos registros
MinistroClass("Ricardo Villas Bôas Cueva", "Villas Bôas Cueva", "GMVBC", ['Ricardo', 'Villas',
'Bôas', 'Cueva', 'Boas'], TIPO_MINISTRO.MINISTRO, "Ministro30") #Grafia sem acento de Bôas
MinistroClass("Rogerio Schietti Machado Cruz", "Rogerio Schietti Cruz", "GMRSC",
['Rogerio', 'Rogério', 'Schietti', 'Machado', 'Cruz'], TIPO_MINISTRO.MINISTRO, "Ministro31")

```

```

MinistroClass("Sebastião Alves dos Reis Júnior", "Sebastião Reis Júnior", "GMSRJ", ['Sebastião',
'Alves', 'Reis', 'Júnior', 'Sebastiao', 'Junior'], TIPO_MINISTRO.MINISTRO, "Ministro32")#
Grafia sem acento de 'Sebastião' e 'Júnior', sem 'dos'
MinistroClass("Sérgio Luiz Kukina", "Sérgio Kukina", "GMSK", ['Sérgio', 'Luiz', 'Kukina',
'Sergio', 'Luiz'], TIPO_MINISTRO.MINISTRO, "Ministro33") # Grafia sem acento de 'Sérgio',
'Luiz'
MinistroClass("Teodoro Silva Santos", "Teodoro Silva Santos", "GMTSS", ['Teodoro', 'Silva',
'Santos'], TIPO_MINISTRO.MINISTRO, "Ministro34")
## Aposentados
MinistroClass("Jorge Mussi", "Jorge Mussi", "GMJM", ['Jorge', 'Mussi'], TIPO_MINISTRO.MINISTRO,
"Ministro35")
MinistroClass("Napoleão Nunes Maia Filho", "Napoleão Nunes Maia
Filho", "GMNMF", ['Napoleão', 'Napoleao', 'Nunes', 'Maia', 'Filho'], TIPO_MINISTRO.MINISTRO,
"Ministro36")
MinistroClass("Nefi Cordeiro", "Nefi Cordeiro", "GMNC", ['Nefi', 'Cordeiro'],
TIPO_MINISTRO.MINISTRO, "Ministro37")
MinistroClass("Laurita Hilário Vaz", "Laurita Vaz", "Aposentada", ['Laurita', 'Hilário', 'Vaz',
'Hilario'], TIPO_MINISTRO.MINISTRA, "Ministro38") # Grafia sem acentos de Hilário
## Desembargadores convidados
MinistroClass("Jesuino Aparecido Rissato", "Jesuino Aparecido Rissato", "GDCJAR", ['Jesuino',
'Aparecido', 'Rissato', 'Jesuino'], TIPO_MINISTRO.DESEMBARGADOR_CONVOCADO, "Ministro39") #
Grafia sem acento de 'Jesuino'
MinistroClass("João Batista Moreira", "João Batista Moreira", "GDCJBM", ['João', 'Batista',
'Moreira', 'Joao'], TIPO_MINISTRO.DESEMBARGADOR_CONVOCADO, "Ministro40") # Grafia sem acento de
'João'
MinistroClass("Manoel de Oliveira Erhardt", "Manoel Erhardt", "SEM CODIGO ERHARDT", ['Manoel',
'Oliveira', 'Erhardt'], TIPO_MINISTRO.DESEMBARGADOR_CONVOCADO, "Ministro40") # sem o 'de'
MinistroClass("Olindo Herculano Menezes", "Olindo Menezes", "SEM CODIGO OLINDO HERCULANO MENEZES",
['Olindo', 'Herculano', 'Menezes'], TIPO_MINISTRO.DESEMBARGADOR_CONVOCADO, "Ministro42")

if __name__ == "__main__":
    # Run a test
    for m in MinistroClass.lista_ministros:
        print(m)
        print(MinistroClass.match_minister("Kukina"))
    )

===== Advogado.py =====
#
# Classe para armazenar dados sobre os advogados
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import re
import pickle
from typing import BinaryIO

import sanitizaCPFOAB
from Stage1_dataToParquet import limpaString
from thefuzz import fuzz

class Advogado:
    # Texto da entrada na consulta processual, tem o formato:
    # 'LEONARDO BRAGA DUARTE - T0008161'
    # 'NOME - EEDDDDD' EE = estado da seccional da OAB e DDDDD é o número da OAB
    advogados_set = set()
    @classmethod
    def advogado_cadastrado(cls, oab: str):
        for a in list(Advogado.advogados_set):
            if a is not None:
                if(a.OAB == oab):
                    return a
        return None

    # Registra um advogado no conjunto de advogados, se já existe, retorna a
    # instância existente, se não cria uma nova e inclui no
    @classmethod
    def registra(cls, texto):
        m=re.match(r"^(.+)-([A-Z][A-Z])(\d\d\d\d\d\d)", texto)
        if m is None:
            return None
        if texto == 'SEM REPRESENTAÇÃO NOS AUTOS - SE000000M': # Caso especial - sem advogado
            return None
        #print(m.groups())
        oab=m.group(2)+m.group(3)

```

```

a=Advogado.advogado_cadastrado(oab)
if a is not None:
    # Já cadastrado, não adiciona outra instância
    print("Cadastrado", a.nome)
    return a
else: # Cadastra
    ni = Advogado()
    ni.nome = m.group(1)
    ni.OAB = m.group(2)+m.group(3)
    ni.OAB_num_only = m.group(3)
    ni.Estado = m.group(2)
    Advogado.advogados_set.add(ni)
    print(ni, " cadastrado")
    return ni
# Compara OAB com Advogado.advogados_set
# Retorna objeto Advogado ou None
@classmethod
def match_oab(cls, oab_adv, against_set=None):
    # check if OAB matches
    # Match OAB
    #print("match_oab( ", oab_adv, " )")
    if against_set is not None:
        test_against : set = against_set
    else:
        test_against : set = cls.advogados_set
    o = sanitizaCPFOAB.sanitizaCPFOAB(oab_adv)
    m = re.match(r"(\d+\.\d+/\d\d)|(\d{9,11})|([A-Z][A-Z])?(\d{0,4}\d\d)(A-Z)?", o)
    if m is not None:
        #print(m.groups(), m.group(4))
        if m.group(4) is not None:
            # Tem OAB
            oab_str :str = m.group(3)+format(int(m.group(4)), "06d")
            # print(oab_str, m.group(3).format("06d"))
            for a in test_against:
                if a is None:
                    return None
                #print(a, a.OAB, a.OAB_num_only, oab_str)
                #if int(a.OAB_num_only) == int(m.group(4)):
                #    print(" >>> found int match", a.OAB_num_only, m.group(4))
                if a.OAB == oab_str:
                    #print("Found match ", a.nome, " ", a.OAB, " - ", m.groups() )
                    return a
        else:
            #print ("OAB No Match")
            return None

@classmethod
def match_advogado(cls, nome_adv, email_adv, oab_adv, against_set=None):
    #print("match_advogado( ",nome_adv, email_adv, oab_adv," )")
    n=nome_adv.upper()
    o=oab_adv.upper()
    adv_as_words=nome_adv.split()
    # If set given, use that, otherwise check against all known lawyers
    if against_set is not None:
        test_against : set = against_set
    else:
        test_against : set = cls.advogados_set
    # check if OAB matches
    # Match OAB, if yes, done.
    oab_matched=cls.match_oab(o, against_set)
    if oab_matched is not None:
        return oab_matched
    # No OAB, tries names, which is tricky...
    if len(adv_as_words) == 1:
        # Can't really match if it is a single word (usually the first name), so bail...
        return None
    ## Two or more words try matching
    for a in test_against:
        #print("a:", a, against_set)
        if a is None:
            return None
        a_limpo = limpaString(a.nome)
        a_lista = a_limpo.split()
        ## Full match, great! Done!
        if a_limpo == n:

```

```

        #print("***Found NAME match: " , a.nome, " , " , a.OAB )
        return a
    setr = fuzz.token_set_ratio(a_limpo, n)
    #setr = fuzz.token_sort_ratio(a_limpo, n)
    #if setr > 60:
    #    print("<><><> setr =", setr, a_limpo, n)
    if setr > 90:
        #print("***Found FUZZ NAME match: " , setr," , n = ",n," , " , a.nome, " , " , a.OAB )
        return a
    ## Conta a quantos match
    #print(nome_adv, " ==>>NO MATCH")
    return None

def __init__(self):
    self.nome = ""
    self.OAB = ""
    self.OAB_num_only = ""
    self.Estado = ""
    return

def __str__(self):
    return self.nome

Test_list = [
    [ "xxxx yyyy zzzz" , "xxxyyyzzz@yahoo.com.br" , "NNN.000.PPP/YY" ],
]
if __name__ == "__main__":
    if False: # Test 1
        test_list=[
            'xxxx yyyy zzzz - EE999999',
            'xxxx yyyy zzzz - EE999999',
            'AAAA BBBB CCC - EE001122'
        ]
        res=[]
        for a in test_list:
            adv=Advogado.registra(a)
            res.append(adv)
            print(str(adv))
            print(adv.OAB)
        print(res)
        print(Advogado.advogados_set)
    if True: # Test 2
        with open("advogados_set.pkl", "rb") as f:
            Advogado.advogados_set=pickle.load(f)
            print("Loaded advogados_set.pkl")
        for l in Test_list:
            Advogado.match_advogado(l[0],l[1],l[2])

    exit(0)

===== atendimentosPorDia.py
=====
#
# Recebe um data frame com os atendimentos e retorna um dataframe com a contagem por dia.
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import time
import pandas
from pandas import DataFrame

#ATENDIMENTOS_PARQUET = r'parquet\Dados atualizados Balcão Virtual.parquet'
#ATENDIMENTOS_PARQUET = r'parquet\Dados Espaço do Advogado.parquet'

def atendimentosPorDia(df_in: DataFrame, nome_coluna_contagem):
    for header in df_in.columns.values:
        print(":"+header+":")
        contPorDia = df_in[["Hora de início", "Hora de conclusão"]].groupby(pandas.Grouper(key="Hora de início", freq="d")).count()
        contPorDia.rename(columns={"Hora de conclusão": "Contagem " + nome_coluna_contagem}, inplace=True)
        contPorDia.index.rename("Dia", inplace=True)

```

```

contPorDia.info()
#print(contPorDia.head())
#print(contPorDia.tail())
#print("Teste2 END ===== ")
#print("agrupado START ===== ")
return contPorDia

if __name__ == "__main__":
    ## Test - read the data file and ouputs excel filas with daily counts
    ## Also saves excel with all the counts combined in one time series

    ## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
    FileTokenDict = { # r'Dados Espaço do Advogado.parquet' : 'ea',
                      r'Dados atualizados Balcão Virtual.parquet' : 'bv',
                      #r'Dados e-mail.parquet': 'email',
                      r'Dados Telefone.parquet': 'tel'
                    }
    fileCount=0
    for pfile, token in FileTokenDict.items():
        print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ",token , "file
count=", fileCount)
        in_df : DataFrame = pandas.read_parquet('parquet\\' + pfile)
        porDia: DataFrame = atendimentosPorDia(in_df, token)
        porDia.to_excel("ts_count"+token+".xlsx")
        if fileCount==0 :
            todos=porDia.copy(deep=True)
        else:
            todos=todos.merge(porDia,how='outer',left_index=True, right_index=True)
        fileCount=fileCount+1

    todos.to_excel(r"ts_count_all.xlsx")

    exit()
===== atendimentosPorUsuario.py
=====
#
# "Best effort" para contar o número de atendimentos por usuário
#
### Paulo Tiberio Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import pandas
from pandas import DataFrame, Series

# Somente o nome é garantido de existir, CPF/ OAB e e-mail não
def listaDeUsuarios(df_in):
    for header in df_in.columns.values:
        print(":" + header + ":")
    usuarios= Series(sorted(df_in["Nome limpo"].unique()))
    emails: Series= Series(df_in["E-mail"].unique())
    CPFouOAB: Series = Series(df_in["CPF ou OAB"].unique())
    print("Nomes: ", usuarios.shape[0])
    print(usuarios.head())
    print(usuarios.info())
    print("e-mails: ", emails.shape[0])
    print(emails.head())
    print("CPF ou OAB",CPFouOAB.shape[0])
    print(CPFouOAB.head())
    combined: DataFrame = df_in[['Nome limpo', 'E-mail', 'CPF ou OAB']].copy()
    combined.sort_values(by=['Nome limpo', 'E-mail', 'CPF ou OAB'], na_position='first',
inplace=True)
    combined.to_excel("usu.xlsx")
    print("Combined: ", combined.shape[0])
    print(combined.head())
    print(combined.info())
    repeat_counts = df_in.pivot_table(index = ['Nome limpo', 'E-mail limpo', 'CPF ou OAB'],
aggfunc='size')
    repeat_counts.to_excel("usu_repeat.xlsx")
    return combined

if __name__ == "__main__":
    ## Test - read the data file and ouputs excel filas with daily counts

```

```

## Also saves excel with all the counts combined in one time series

## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
FileTokenDict = { #r'Dados Espaço do Advogado.parquet': 'ea',
                  #r'Dados atualizados Balcão Virtual.parquet': 'bv',
                  #r'Dados e-mail.parquet': 'email',
                  #r'Dados Telefone.parquet': 'tel'
                }

fileCount = 0
for pfile, token in FileTokenDict.items():
    print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ", token, "file
count=", fileCount)
    in_df: DataFrame = pandas.read_parquet('parquet\\' + pfile)
    print(in_df.info())
    #temp = listaDeUsuarios(in_df)
    repeat_counts = in_df.pivot_table(index=['Nome limpo', 'E-mail limpo', 'CPF_OAB limpo'],
aggfunc='size')
    #repeat_counts = in_df.pivot_table(index=['Nome limpo'], aggfunc='size')
    repeat_counts.to_excel("usu_"+token+"_repeat.xlsx")
    # temp is a Series with time in seconds per atendimento
    #temp.plot(title="Contagem x segundos")
    #temp.hist(bins=60)
    #matplotlib.pyplot.show()
    #plt.hist(temp.values)
    #porDia: DataFrame = tiposDeAtendimento(in_df, token)
    #porDia.to_excel("atend_unique_" + token + ".xlsx")
    #if fileCount == 0:
    #    todos: Series = temp.copy(deep=True)
    #else:
    #    todos = pandas.concat([todos, temp], ignore_index=True, sort=True)
    fileCount = fileCount + 1
    #input("Press Enter to continue...")
    #todos=Series(todos.unique())
    #todos.to_excel(r"atend_all.xlsx")

    exit()

===== contagemAdv.py =====
#
# Contagem de advogados
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import pandas
from pandas.core.interchange.dataframe_protocol import DataFrame
from pandas.core.series import Series

import normalizeAtendimento
from normalizeAtendimento import sanitizaAtendimento
from sanitizaDispositivo import sanitizaDispositivo

if __name__ == "__main__":
    ## Test - read the data file and ouputs excel filas with daily counts
    ## Also saves excel with all the counts combined in one time series

    ## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
    FileTokenDict = { #r'Dados Espaço do Advogado.parquet': 'ea',
                      #r'Dados atualizados Balcão Virtual.parquet': 'bv'
                      # r'Dados e-mail.parquet': 'email',
                      #, r'Dados Telefone.parquet': 'tel'
                    }

    fileCount = 0
    for pfile, token in FileTokenDict.items():
        print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ", token, "file
count=", fileCount)
        in_df: DataFrame = pandas.read_parquet('parquet\\' + pfile)
        print(in_df.info())
        #temp = listaDeUsuarios(in_df)
        perfis: Series = in_df['Perfil'].value_counts()
        print(perfis)
        perfis.to_excel("cont_perfis_"+token+"_count.xlsx")
        fileCount = fileCount + 1

    exit()

```

```

===== contagemDeficienteAuditivo.py
=====
#
# Conta registros de atendimentos com pessoas que sofre de deficiencia auditiva
#
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import pandas
from pandas.core.interchange.dataframe_protocol import DataFrame

import normalizeAtendimento
from normalizeAtendimento import sanitizaAtendimento
from sanitizaDispositivo import sanitizaDispositivo

if __name__ == "__main__":
    ## Test - read the data file and ouputs excel filas with daily counts
    ## Also saves excel with all the counts combined in one time series

    ## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
    FileTokenDict = { #r'Dados Espaço do Advogado.parquet': 'ea',
                      r'Dados atualizados Balcão Virtual.parquet': 'bv'
                      # r'Dados e-mail.parquet': 'email',
                      #, r'Dados Telefone.parquet': 'tel'
                    }

    fileCount = 0
    for pfile, token in FileTokenDict.items():
        print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ", token, "file
count=", fileCount)
        in_df: DataFrame = pandas.read_parquet('parquet\\' + pfile)
        print(in_df.info())
        # temp = listaDeUsuarios(in_df)
        defAuditiva: pandas.Series = in_df['Pessoa surda?'].value_counts()
        print(defAuditiva)
        defAuditiva.to_excel("cont_defAudio_" + token + "_count.xlsx")
        fileCount = fileCount + 1
        #input("Press Enter to continue...")
        #todos=Series(todos.unique())
        #todos.to_excel(r"atend_all.xlsx")

    exit()
===== contagemDeInstrumentos.py
=====
#
# Faz contagem dos instrumentos que mais aparecem nas consultas ao balcão virtual
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import pandas
from pandas import DataFrame, Series
from Processo import ProcessoClass

from Config import FILE_DADOS_CONSULTADOS

#
# Conta quantas vezes um instrumento aparece numa lista de atendimentos
#
# A coluna ['Instrumento'] tem uma lista de dicionários com instrumentos
#
def contaInstrumentos(df: DataFrame) -> DataFrame:
    instrumentos=Series(df['Instrumento'])
    todos_os_instrumentos=set()
    contagem=dict()
    consultas_validas=0
    print("Contando Instrumentos")
    for ilst in instrumentos:
        # print("Lista de intrumentos = ",ilst)
        if ilst is not None:
            consultas_validas+=1
            for instdict in ilst:
                for key, v in instdict.items():
                    if v:
                        if key != "RegistroSTJ" and key != "Processo": # Processo e Registro no
STJ não são tipos de processo
                            k :str = ProcessoClass.nome_por_extenso(key)

```

```

        if k == "":
            ##print("key, k, v = ", key, k, v)
            k = key
        if k in contagem:
            contagem[k] += 1
        else:
            contagem[k]=1
        todos_os_instrumentos.add(k)
    print(todos_os_instrumentos)
    print("consultas válidas:", consultas_validas)
    contagem_df=DataFrame.from_dict(contagem, orient='index',columns= ['Contagem'])
    contagem_df.sort_values(by='Contagem',ascending=False, inplace=True)
    contagem_df.index.rename("Instrumento", inplace=True)
    return contagem_df

if __name__ == "__main__":
    ## Entradas para contagem de instrumentos em todos os dados
    #infile=FILE_DADOS_CONSULTADOS
    #outfile="freq_instrumentos.xlsx"
    ## Entradas para contagem de instrumentos nas chamadas de ministros
    infile="falas_min_consultados_new.parquet"
    outfile="freq_instrumentos_mins.xlsx"
    print("Reading data from file " + infile)
    df: pandas.DataFrame = pandas.read_parquet(infile, engine='pyarrow')
    contagem_inst: DataFrame = contaInstrumentos(df)
    contagem_inst.to_excel(outfile)
    exit(0)

===== contagemDePartes.py
=====
#
# Faz contagem das partes que mais aparecem nas consultas ao balcão virtual
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import pandas
from pandas import DataFrame, Series

from main import FILE_DADOS_CONSULTADOS

#
# Conta quantas vezes uma parte aparece nas consultas
#
# A coluna ['Consultas'] tem o dicionário com os dados do processo,
# partes está em df[Consultas][Partes] e é uma lista de nomes de partes
def contaPartes(df: DataFrame):
    consultas=Series(df['Consultas'])
    todas_as_partes=set()
    contagem=dict()
    consultas_validas=0
    print("Contando partes")
    for c in consultas:
        if c[0]['Partes'] is not None:
            consultas_validas+=1
            for p in c[0]['Partes']:
                if p in contagem:
                    contagem[p] += 1
                else:
                    contagem[p]=1
            todas_as_partes.add(p)
    print(todas_as_partes)
    #serie_sp=Series(list(todas_as_partes))
    #serie_sp.to_excel("lista_partes.xlsx")
    print("consultas válidas:", consultas_validas)
    contagem_df=DataFrame.from_dict(contagem, orient='index',columns= ['Contagem'])
    contagem_df.sort_values(by='Contagem',ascending=False, inplace=True)
    contagem_df.to_excel("freq_partes.xlsx")

if __name__ == "__main__":
    print("Reading data from file " + FILE_DADOS_CONSULTADOS)
    df: pandas.DataFrame = pandas.read_parquet(FILE_DADOS_CONSULTADOS, engine='pyarrow')
    contaPartes(df)
    exit(0)

===== contagemDispositivoAtendimento.py
=====
#

```

```

# Contagem de dispositivos de atendimento - computador/celular
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import pandas
from pandas.core.interchange.dataframe_protocol import DataFrame

import normalizeAtendimento
from normalizeAtendimento import sanitizaAtendimento
from sanitizaDispositivo import sanitizaDispositivo

if __name__ == "__main__":
    ## Test - read the data file and outputs excel filas with daily counts
    ## Also saves excel with all the counts combined in one time series

    ## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
    FileTokenDict = { #r'Dados Espaço do Advogado.parquet': 'ea',
                      # r'Dados atualizados Balcão Virtual.parquet': 'bv'
                      # r'Dados e-mail.parquet': 'email',
                      #, r'Dados Telefone.parquet': 'tel'
                    }

    fileCount = 0
    for pfile, token in FileTokenDict.items():
        print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ", token, "file
count=", fileCount)
        in_df: DataFrame = pandas.read_parquet('parquet\\' + pfile)
        print(in_df.info())
        # temp = listaDeUsuarios(in_df)
        disps: pandas.Series = in_df['Dispositivo'].value_counts()
        print(disps)
        disps.to_excel("cont_disp_" + token + "_count.xlsx")
        fileCount = fileCount + 1
        #input("Press Enter to continue...")
        #todos=Series(todos.unique())
        #todos.to_excel(r"atend_all.xlsx")

    exit()
===== contagemTiposDeAtendimento.py
=====
#
# Contagem de tipos de atendimento
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

import pandas
from pandas.core.interchange.dataframe_protocol import DataFrame

import normalizeAtendimento
from normalizeAtendimento import sanitizaAtendimento

if __name__ == "__main__":
    ## Test - read the data file and outputs excel filas with daily counts
    ## Also saves excel with all the counts combined in one time series

    ## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
    FileTokenDict = { #r'Dados Espaço do Advogado.parquet': 'ea',
                      # r'Dados atualizados Balcão Virtual.parquet': 'bv',
                      # r'Dados e-mail.parquet': 'email',
                      # r'Dados Telefone.parquet': 'tel'
                    }

    fileCount = 0
    for pfile, token in FileTokenDict.items():
        print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ", token, "file
count=", fileCount)
        in_df: DataFrame = pandas.read_parquet('parquet\\' + pfile)
        #print(in_df.info())
        #temp = listaDeUsuarios(in_df)
        #repeat_counts = in_df.pivot_table(index=['Nome limpo', 'E-mail limpo', 'CPF_OAB
limpo'], aggfunc='size')

        ## Moving cleaning to Stage1
        in_df['Assunto 1 limpo'] = in_df['Assunto 1'].apply(sanitizaAtendimento)
        in_df['Assunto 2 limpo'] = in_df['Assunto 2'].apply(sanitizaAtendimento)

```

```

in_df['Assunto 3 limpo'] = in_df['Assunto 3'].apply(sanitizaAtendimento)
assunto1 = in_df.pivot_table(index=['Assunto 1 limpo'], aggfunc='size')
assunto1.name = 'Assunto 1 count'
assunto2 = in_df.pivot_table(index=['Assunto 2 limpo'], aggfunc='size')
assunto2.name = 'Assunto 2 count'
assunto3 = in_df.pivot_table(index=['Assunto 3 limpo'], aggfunc='size')
assunto3.name = 'Assunto 3 count'
print(assunto1.describe())
print(assunto1.info())
print(assunto1.head())
print(assunto2.describe())
print(assunto2.info())
print(assunto2.head())

counts_assuntos = pandas.concat([assunto1,assunto2,assunto3], axis=1)
# counts_assuntos['Totais'] = counts_assuntos['Assunto 1
count'] + counts_assuntos['Assunto 1 count'] + counts_assuntos['Assunto 3 count']
counts_assuntos['Assuntos total'] = counts_assuntos[['Assunto 1 count', 'Assunto 2
count', 'Assunto 3 count']].agg('sum', axis=1)
counts_assuntos.to_excel("assunto_" + token + "_count.xlsx")
#count_tot.to_excel("assunto_tot_" + token + "_count.xlsx")
#assunto1.to_excel("assunto1_" + token + "_count.xlsx")
#assunto2.to_excel("assunto2_" + token + "_count.xlsx")
#assunto3.to_excel("assunto3_" + token + "_count.xlsx")
# temp is a Series with time in seconds per atendimento
#temp.plot(title="Contagem x segundos")
#temp.hist(bins=60)
#matplotlib.pyplot.show()
#plt.hist(temp.values)
#porDia: DataFrame = tiposDeAtendimento(in_df, token)
#porDia.to_excel("atend_unique_" + token + ".xlsx")
#if fileCount == 0:
#    todos: Series = temp.copy(deep=True)
#else:
#    todos = pandas.concat([todos, temp], ignore_index=True, sort=True)
fileCount = fileCount + 1
#input("Press Enter to continue...")
#todos=Series(todos.unique())
#todos.to_excel(r"atend_all.xlsx")

exit()
===== freqTribunaisOrigem.py
=====
#
# Faz contagem dos tribunais de origem nos dados consultados
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import pandas
from pandas import DataFrame, Series
from pyarrow.types import is_list
import pickle
from main import FILE_DADOS_CONSULTADOS
import Advogado
import Processo
#
# Conta quantas vezes uma parte aparece nas consultas
#
# A coluna ['Consultas'] temp o dicionário com os dados do processo,
# partes está em df[Consultas][Partes] e é um lista de nomes de partes
def contaTribunais( consultas_bin_array: list):
    contagem=len(consultas_bin_array)
    print(contagem)
    cont_valid=0
    tribunal_counts=dict()
    print("Contando tribunais")
    for idx in consultas_bin_array: # Array of lists of ProcessoClass.Processo objects
        cl=consultas_bin_array[idx]
        print(cl)
        for p in cl:
            print(p)
            if p is not None:
                proc: Processo.ProcessoClass = p
                cont_valid += 1
                if proc.tribunal_de_origem in tribunal_counts:

```

```

        tribunal_counts[proc.tribunal_de_origem] += 1
    else:
        tribunal_counts[proc.tribunal_de_origem] = 1
    print("Tribunais de origem: ", tribunal_counts.keys())
    print("Entradas válidas =", cont_valid)
    contagem_df=DataFrame.from_dict(tribunal_counts, orient='index', columns= ['Contagem'])
    contagem_df.index.name='Tribunal de Origem'
    contagem_df.sort_values(by='Contagem',ascending=False, inplace=True)
    contagem_df.to_excel("freq_tribunais.xlsx")

if __name__ == "__main__":
    print("Reading data from file " + FILE_DADOS_CONSULTADOS)
    df: pandas.DataFrame = pandas.read_parquet(FILE_DADOS_CONSULTADOS, engine='pyarrow')
    with open("consultas_objs.pkl", "rb") as f:
        consultas_obarray=pickle.load(f)
        f.close()
        print("Loaded consultas_objs.pkl pickle file")
    with open("advogados_set.pkl", "rb") as f:
        Advogado.Advogado.advogados_set= pickle.load(f)
        f.close()
        print("Loaded advogados_set.pkl pickle file")

    contaTribunais(consultas_obarray)
    exit(0)

===== normalizeAtendimento.py
=====
#
# Corrige erros diversas grafias dos atendimentos
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import math

# Lista de atendimentos
Atendimentos_canonicos={
# Alias          CanonicoAlias
'Andamento processual' : 'Andamento/Localização Processual',
'Andamento/Localização Processual' : 'Andamento/Localização Processual',
'Andamento/Localização Processual ' : 'Andamento/Localização Processual',
'Andamento/Localização ProcessualExecução/RPV/Precatório' : 'Execução/RPV/Precatório',
'Audiência presidência' : 'Audiência presidência',
'BacenJUD (SISBAJUD)' : 'BacenJUD (SISBAJUD)',
'Carta Rogatória' : 'Carta Rogatória',
'Certidão' : 'Certidão',
'Certidão ' : 'Certidão',
'Chave de Acesso' : 'Chave de Acesso',
'Ciência Antecipada' : 'Ciência Antecipada',
'Ciência Antecipada ' : 'Ciência Antecipada',
'Conexão Interrompida' : 'Conexão Interrompida',
'Conexão Interrompida ' : 'Conexão Interrompida',
'Consultoria jurídica' : 'Consultoria Jurídica',
'Consultoria Jurídica' : 'Consultoria Jurídica',
'Consultoria Jurídica ' : 'Consultoria Jurídica',
'Contato com outros tribunais' : 'Contato com outros tribunais',
'Contato com outros tribunais ' : 'Contato com outros tribunais',
'Correção autuação' : 'Correção de Autuação/Habilitação',
'Correção Autuação' : 'Correção de Autuação/Habilitação',
'Correção de Autuação/Habilitação' : 'Correção de Autuação/Habilitação',
'Correção de Autuação/Habilitação ' : 'Correção de Autuação/Habilitação',
'Corte Especial (Sigilo)' : 'Corte Especial (Sigilo)',
'Corte Especial (Sigilo) ' : 'Corte Especial (Sigilo)',
'CPE' : 'CPE',
'CPE - Indisponibilidade' : 'CPE - Indisponibilidade',
'CPE/Protocolo' : 'CPE/Protocolo',
'CPE/Protocolo ' : 'CPE/Protocolo',
'Depósito Judicial' : 'Depósito Judicial',
'Despachar com Ministro' : 'Despachar/Falar no Gabinete',
'Despachar com MinistroPlantão / Recesso / Feriado' : 'Despachar com
Ministro/Plantão/Recesso/Feriado',
'Despachar/Falar no Gabinete' : 'Despachar/Falar no Gabinete',
'Despachar/Falar no Gabinete ' : 'Despachar/Falar no Gabinete',
'Despesas Processuais' : 'Despesas Processuais',
'Despesas processuais' : 'Despesas Processuais',
'DJE/intimação' : 'DJE/Publicação/Intimação',
'DJE/Portal de intimação' : 'DJE/Publicação/Intimação',

```

```

'DJE/Publicação/Intimação'      : 'DJE/Publicação/Intimação',
'DJE/Publicação/Intimação '    : 'DJE/Publicação/Intimação',
'Execução/RPV/Precatório'     : 'Execução/RPV/Precatório',
'Execução/RPV/Precatório '    : 'Execução/RPV/Precatório',
'Falar na Coordenadoria'       : 'Falar na Coordenadoria',
'Fraude'                        : 'Fraude',
'GRU'                           : 'GRU',
'GRU '                          : 'GRU',
'GRU - Indisponibilidade'      : 'GRU - Indisponibilidade',
'Habilitação nos autos'       : 'Habilitação nos autos',
'HDE'                          : 'HDE/Carta Rogatória',
'HDE/Carta Rogatória'         : 'HDE/Carta Rogatória',
'HDE/Carta Rogatória '        : 'HDE/Carta Rogatória',
'Indisponibilidade de sistema' : 'Indisponibilidade de sistema',
'Informações Administrativas' : 'Informações Administrativas',
'Informações Administrativas ' : 'Informações Administrativas ',
'Informações Processuais'     : 'Informações Processuais',
'informações processuais'      : 'Informações Processuais',
'Informações Processuais '    : 'Informações Processuais',
'Julgamento Virtual'         : 'Julgamento Virtual',
'Julgamento virtual'         : 'Julgamento Virtual',
'Julgamento Virtual '        : 'Julgamento Virtual',
'Ligação muda / caiu'         : 'Ligação muda/caiu',
'Malote Digital'              : 'Malote Digital',
'Malote digital'              : 'Malote Digital',
'Malote Digital '             : 'Malote Digital',
'Memoriais'                   : 'Memoriais',
'MEMORIAIS'                   : 'Memoriais',
'Memoriais '                  : 'Memoriais',
'Plantão / Recesso / Feriado' : 'Plantão/Recesso/Feriado',
'Plantão/Recesso/Feriado'     : 'Plantão/Recesso/Feriado',
'Plantão/Recesso/Feriado '    : 'Plantão/Recesso/Feriado',
'Preferência Julgamento'     : 'Preferência Julgamento',
'Previsão de Distribuição/Triagem' : 'Previsão de Distribuição/Triagem',
'Previsão de Distribuição/Triagem ' : 'Previsão de Distribuição/Triagem ',
'Previsão de Julgamento'      : 'Previsão de Julgamento/Distribuição',
'Previsão de Julgamento '    : 'Previsão de Julgamento/Distribuição',
'Previsão deDespachar/Falar no Gabinete' : 'Previsão de Despachar/Falar no Gabinete',
'Previsão julgamento / distribuição' : 'Previsão de Julgamento/Distribuição',
'Processos sigilosos (corte especial)' : 'Processos sigilosos (corte especial)',
'Protocolo Judicial'          : 'Protocolo Judicial',
'Publicação/Intimação'        : 'Publicação/Intimação',
'Recesso / Plantão'           : 'Plantão/Recesso/Feriado',
'Recurso Repetitivo'          : 'Recurso Repetitivo/Jurisprudência',
'Recurso Repetitivo '        : 'Recurso Repetitivo/Jurisprudência',
'Recurso Repetitivo/Jurisprudência' : 'Recurso Repetitivo/Jurisprudência',
'Segredo de Justiça'          : 'Segredo de Justiça',
'Segredo de Justiça '         : 'Segredo de Justiça',
'Sessões Julgamento'         : 'Sessões Julgamento/Preferência',
'Sessões Julgamento/Preferência' : 'Sessões Julgamento/Preferência',
'Sessões Julgamento/Preferência ' : 'Sessões Julgamento/Preferência',
'Sistema PUSH'                : 'Sistema Push',
'Sistema Push'                : 'Sistema Push',
'Sistema PUSH '               : 'Sistema Push',
'Sustentação Oral'            : 'Sustentação Oral',
'Sustentação oral'            : 'Sustentação Oral',
'sustentação Oral'            : 'Sustentação Oral',
'Sustentação Oral '           : 'Sustentação Oral',
'Tentativa de Fraude'         : 'Tentativa de Fraude',
'Transferência interna'       : 'Transferência interna',
'Transito em Julgado/Baixa de processo' : 'Trânsito em Julgado/Baixa/Remessa ao STF',
'Trânsito/Baixa/Remessa ao STF' : 'Trânsito em Julgado/Baixa/Remessa ao STF',
'Trânsito/Baixa/Remessa ao STF ' : 'Trânsito em Julgado/Baixa/Remessa ao STF',
'Vista para o MPF'            : 'Vista para o MPF',
'Vista para o MPF '           : 'Vista para o MPF',
'X'                            : 'X',
'x'                            : 'X',
'XDJE/Publicação/Intimação'   : 'DJE/Publicação/Intimação'
}

```

```

## Retorna o nome canonico do atendimento - dá KeyError se str não for conhecida (i.e., não está na tabela acima)

```

```

## Se KeyError for um problema, é possible substituir por Atendimento.canonicos.get(str)

```

```

def sanitizaAtendimento(atendimento: str) -> str:

```

```

    if atendimento is None:

```

```

        return "X"
    if isinstance(atendimento, str):
        return Atendimentos_canonicos[atendimento]
    else:
        return "X"

===== sanitizaCPFOAB.py
=====
# Sanitiza campo que tem CPF ou OAB
#
# OAB Em geral tem o formato <<2 letras estados>><<numero 4-6 digitos>>
# CPF São 11 digitos, o ideal é passar validação de digito verificador
# Remove espaço em branco
# MELHORIA: Separarar e formatar OAB/CPF
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import math
import re

#from openpyxl.pivot.fields import Number
from pyparsing import NotAny

UF = ['ac', 'al', 'ap', 'am', 'ba', 'ce', 'df', 'es', 'go', 'ma', 'mt', 'ms', 'mg', 'pa', 'pb', 'pr', 'pe',
      'pi', 'rj', 'rn', 'rs', 'ro', 'rr', 'sc', 'sp', 'se', 'to']
]
def valida_cpf(cpf):
    # If not 11 digits, fail
    if not re.match(r'\d{11}', cpf):
        return False
    # 11 digits, good!
    mult=10
    dv1=0
    for d in cpf[:9]:
        dv1+=int(d)*mult
        mult-=1
        #print("d, dv1, mult",d,dv1,mult)
    dv1 = dv1 % 11
    #dv1= (cpf[0]*10+cpf[1]*9+cpf[2]*8+cpf[3]*7+cpf[4]*6+cpf[5]*5+cpf[6]*4+cpf[7]*3+cpf[8]*2) %
11
    if dv1 < 2:
        dv1=0
    else:
        dv1 = 11 - dv1
    #dv2 =
    (cpf[0]*11+cpf[1]*10+cpf[2]*9+cpf[3]*8+cpf[4]*7+cpf[5]*6+cpf[6]*5+cpf[7]*4+cpf[8]*3+dv1*2) % 11
    #print(dv1)

    mult=11
    dv2=0
    for d in cpf[:9]:
        dv2 += int(d) * mult
        mult -= 1
        #print("d, dv2, mult", d, dv2, mult)
    dv2 += dv1*2
    dv2 = dv2 % 11
    if dv2 < 2:
        dv2=0
    else:
        dv2 = 11 - dv2
    #print(dv2)

    dvs= str(dv1)+str(dv2)
    #print(dvs, " - ", cpf[9:])
    return dvs == cpf[9:]

def formatCPF(cpf):
    # Should verify digit and known invalid (000.000.000/00, etc.)
    return cpf[:3]+"."+cpf[3:6]+"."+cpf[6:9]+"/"+cpf[9:]

def formatOAB( uf, digitos):
    # Remove 0s na frente dos outros digitos
    #print("format UF: ", uf, " digitis:", digitos)
    digitos=re.sub(r"^0+", "",digitos)
    return uf+digitos

```

```

def sanitizaCPFOAB( in_str: str):
    # Remove espaços no fim/inicio e coloca em minusculas
    if in_str is None:
        return "RUIM_None"
    if isinstance(in_str,float):
        if math.isnan(in_str):
            return "RUIM_NaN"
        else:
            print("CPFOAB: Number", in_str)
            in_str=str(in_str)
    work=in_str.strip()
    work=work.lower()
    # Remove espaços internos, / - .
    work=re.sub(r"[\s\-\.\./]", "", work)
    # Remove OAB
    work = re.sub(r"oab", "", work)
    # Se tem somente 11 digitos deve ser CPF
    if valida_cpf(work): # CPF
        return formatCPF(work)
    # Não é CPF Tenta OAB
    if(len(work) <4):
        # Muito curta, marca como ruim
        return "RUIM_"+work
    Front2c=work[:2]
    Back2c=work[-2:]
    #print(work, " Front: ", Front2c," Back:", Back2c)
    if Front2c in UF:
        return formatOAB(Front2c,work[2:])
    if Back2c in UF:
        return formatOAB(Front2c,work[:-2])
    else:
        return "RUIM_" + work

if __name__ == "__main__":
    exit(0)

===== sanitizaDispositivo.py
=====
#
# Dictionarty de entradas brutas para entradas sanitizadas da coluna Dispositivos dos dados do
# Balcão Virtual
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

cleaning_table_dispositivo = {
    "Computador": "Computador",
    "Computador ": "Computador",
    "Computador/Notebook": "Computador",
    "Compuador": "Computador",
    "Cmputador": "Computador",
    "Celular": "Celular/Tablet",
    "Celular/Tablet": "Celular/Tablet",
    "Celular/Tablet ": "Celular/Tablet",
    "Celular ": "Celular/Tablet",
    "x": "X"
}

## Retorna o nome canonico do atendimento - dá KeyError se str não for conhecida (i.e., não está
na tabela acima)
## Se KeyError for um porblema, é possible substituir pot Atendimnetos_canonicos.get(str)
def sanitizaDispositivo(dis):
    if disp == None:
        return "X"
    else:
        return cleaning_table_dispositivo[disp.strip()]
===== separaConsultaGabinete.py
=====
#
# Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
#
# Programa separar dados de relacionados a despachos com gabinete de ministro
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br

```

```

### Copyright (c) STJ 2025 Licença CC BY-NC

import re
from typing import Pattern
import Processos
from pandas import DataFrame
from Ministros import MinistroClass
import pandas
import numpy

## Dados processados por Stage1_dataToParquet.py
ATENDIMENTOS_PARQUET = r"parquet\Dados atualizados Balcão Virtual.parquet"
COLUNAS_STAGE1 = [ 7, 8, 9 ]
## Dados consultados
CONSULTADOS_PARQUET= r"Dados atualizados Balcão Virtual_consulta.parquet"
CONSULTADOS_COLUNAS = [ 44, 45, 46]
## SE o formato dos dados do Balcão virtual mudar, tem que mudar o índice das colunas abaixo

#LISTA_CONSULTA_PROCESSUAL_PARQUET = "lista_consulta_processual.parquet"
#DIRETORIO_CONSULTA_PROCESSUAL_BATCH = "consultas_html"
#FILE_DADOS_CONSULTADOS="dados_consultados.parquet"

#MINI = {'Despachar/Falar no Gabinete', 'Despachar com Ministro', 'Despachar - Falar no
gabinete',
#       'Despachar e Falar no gabinete', 'Despachar- Falar no gabinete', 'Despachar-Falar no
gabinete',
#       'Despachar e falar no gabinete', 'Falar no gabinete', 'Despachar e Falar no Gabinete',
#       'Solicitação de Audiência em gabinete', 'solicitação de AUDIÊNCIA em gabinete'
#       }
RE_Falar_com_Gabinete: Pattern[str] = re.compile('.*(ministro|gabinete).*', re.IGNORECASE)

#
# True se algum dos três assuntos da consulta é relacionado a Gabinete.
#
def e_fala_com_gabinete(a1: numpy.ndarray, col1, col2, col3) -> bool:
    #print(a1[COLUNA_ASSUNTO1], a1[COLUNA_ASSUNTO2], a1[COLUNA_ASSUNTO3])
    bol1=False
    bol2=False
    bol3=False
    print("[", a1[43], " , ", a1[col1], " , ", a1[col2], " , ", a1[col3], "]")
    if a1[col1] is not None:
        bol1 =RE_Falar_com_Gabinete.match(a1[col1]) is not None
    if a1[col2] is not None:
        bol2 =RE_Falar_com_Gabinete.match(a1[col2]) is not None
    if a1[col3] is not None:
        bol3 =RE_Falar_com_Gabinete.match(a1[col3]) is not None
    return bol1 or bol2 or bol3

# Entrada - NOME do arquivo parquet
# Colunas = [ índice da coluna do atendimento 1, 2 , 3 ]
# Saída = Nome do arquivo de saída sem extensão

def separaConsultaGabinete(atendimentos: DataFrame, colunas: list) -> DataFrame:
    #
    # Lê dados sanitizados por Stage1_dataToParquet
    #

    print(colunas)
    falas_com_gabinete = atendimentos.loc[ atendimentos.apply(e_fala_com_gabinete,axis=1,
raw=True, result_type=None, col1=colunas[0], col2=colunas[1], col3=colunas[2]) ]
    print(falas_com_gabinete.info)
    return falas_com_gabinete

if __name__ == "__main__":
    # Dados para filtrar dados do stage1
    #infile=ATENDIMENTOS_PARQUET
    #outfile="falas_min_inst_new" # arquivo de saída sem extensão
    #cols=COLUNAS_STAGE1
    # Dados para filtrar dados consultados
    infile = CONSULTADOS_PARQUET
    outfile = "falas_min_consultados_new" # arquivo de saída sem extensão
    cols =CONSULTADOS_COLUNAS
    print("Reading data from file " + infile)
    atendimentos: DataFrame = pandas.read_parquet(infile, engine='pyarrow')

```

```

print("Dataframe: ")
print(atendimentos.info, atendimentos.describe())
df_saida: DataFrame = separaConsultaGabinete(atendimentos, cols)
print("Dataframe separado: ", df_saida.info, df_saida.describe())
df_saida.to_excel(outfile+".xlsx")
df_saida.to_parquet(outfile+".parquet", engine='pyarrow')
===== tempoMedioDeAtendimento.py
=====
#
# Calcula o tempo médio de atendimento de um dataframe
# Baseia-se em "Hora de início", "Hora de conclusão"
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC
import numpy
import pandas
from scipy import stats
from pandas.core.interchange.dataframe_protocol import DataFrame
from pandas.io.formats.format import SeriesFormatter
from datetime import time, timedelta
import matplotlib
#
# Calcula o tempo médio de atendimento de uma lista de tipos de atendimento
# Se a lista for None, calcula a média geral do DataFram
def tempoMedioDeAtendimento(df: DataFrame, lista_atendimentos):
    print(df.info())
    temp: pandas.Series = pandas.Series( df["Hora de conclusão"] - df["Hora de início"])
    #temp: pandas.Series = pandas.Series(df["Hora de conclusão"] - df["Hora de início"],
index=df["Hora de início"])
    print(temp)
    temp2 = temp.map(timedelta.total_seconds)
    # Drop 0 length calls
    temp2 = temp2[temp2.values!=0]
    tempos = pandas.Series(temp2.values)
    print(tempos.describe(), tempos.info())
    tempos=tempos[numpy.abs(stats.zscore(tempos)<3)]

    print("Minium time:",tempos.min(), "Max time:",tempos.max())
    return tempos

if __name__ == "__main__":
    ## Test - read the data file and ouputs excel filas with daily counts
    ## Also saves excel with all the counts combined in one time series

    ## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
    FileTokenDict = { #r'Dados Espaço do Advogado.parquet': 'ea',
                      #r'Dados atualizados Balcão Virtual.parquet': 'bv',
                      r'\..\falas_min_inst.parquet': 'bv',
                      #r'Dados e-mail.parquet': 'email',
                      #r'Dados Telefone.parquet': 'tel'
                    }

    fileCount = 0
    for pfile, token in FileTokenDict.items():
        print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ", token, "file
count=", fileCount)
        in_df: DataFrame = pandas.read_parquet('parquet\\' + pfile)
        temp = tempoMedioDeAtendimento(in_df, None)
        print("START TOKEN: ", token)
        print(temp.describe())
        temp.plot(kind="hist", bins=100, title=pfile)
        matplotlib.pyplot.show()
        temp.to_excel("tempo_medio_" + token + ".xlsx")
        fileCount = fileCount + 1
        print("END TOKEN:", token)

    exit()
===== tiposDeAtendimento.py
=====
#
# Lê os arquivos de atendimentos e compila lista global de tipos de atendimentos
# que serve de insumo para a tabela em normalizeAtendimento
# A lista de atendimentos é compilada das colunas "Assunto 1", "Assunto 2" e "Assunto 3"
#
### Paulo Tibério Muradas Bulhões - tiberiob@stj.jus.br
### Copyright (c) STJ 2025 Licença CC BY-NC

```

```

import time
import pandas
from pandas import DataFrame, Series

#ATENDIMENTOS_PARQUET = r'parquet\Dados atualizados Balcão Virtual.parquet'
#ATENDIMENTOS_PARQUET = r'parquet\Dados Espaço do Advogado.parquet'

# Faz lista geral de tipos de atendimentos
#
def tiposDeAtendimento(df_in: DataFrame, nome_coluna_contagem):
    for header in df_in.columns.values:
        print(":" + header + ":")
    print(df_in["Assunto 1"].unique())
    atend1 = pandas.Series(df_in["Assunto 1"].unique())
    #atend1.info()
    atend2 = pandas.Series(df_in["Assunto 2"].unique())
    atend3 = pandas.Series(df_in["Assunto 3"].unique())
    attends = sorted([x for x in pandas.concat([atend1,atend2, atend3]).unique() if x is not
None])
    print("Todos atendimentos", attends )
    return Series(attends)

if __name__ == "__main__":
    ## Test - read the data file and ouputs excel filas with daily counts
    ## Also saves excel with all the counts combined in one time series

    ## Dict com o nome dos arquivos de entrada e o nome da coluna de contagem dos atendimentos
    FileTokenDict = {r'Dados Espaço do Advogado.parquet': 'ea',
                     r'Dados atualizados Balcão Virtual.parquet': 'bv',
                     r'Dados e-mail.parquet': 'email',
                     r'Dados Telefone.parquet': 'tel'}

    fileCount = 0
    for pfile, token in FileTokenDict.items():
        print(">>> Reading data from file " + pfile, "nome_coluna_contagem is ", token, "file
count=", fileCount)
        in_df: DataFrame = pandas.read_parquet('parquet\\' + pfile)
        temp: DataFrame = tiposDeAtendimento(in_df, token)
        #porDia: DataFrame = tiposDeAtendimento(in_df, token)
        #porDia.to_excel("atend_unique_" + token + ".xlsx")
        if fileCount == 0:
            todos: Series = temp.copy(deep=True)
        else:
            todos = pandas.concat([todos, temp], ignore_index=True, sort=True)
            fileCount = fileCount + 1
        todos=Series(todos.unique())
        todos.to_excel(r"atend_all.xlsx")

    exit()

```