

Daniel Coelho Liberali

Desenvolvimento de Produto: Otimizador para Mobile Marketing

São Paulo

2021

Lombada

Daniel Coelho Liberali

Desenvolvimento de Produto: Otimizador para Mobile Marketing

Trabalho de Formatura apresentado à Escola
Politécnica da Universidade de São Paulo para
obtenção do diploma de Engenheiro de
Produção

Orientador: Eduardo de Senzi Zancul

São Paulo

2021

Ficha catalográfica

AGRADECIMENTOS

À minha família, que esteve sempre ao meu lado e me ajudou a cada passo nessa jornada e que terá sempre minha mais profunda gratidão.

Ao Lucas Kok, por sua companhia e companheirismo até quando as coisas estavam mais apertadas.

À minha terapeuta, Kelly, por me ajudar a seguir em frente quando meu maior obstáculo era eu mesmo.

Ao meu orientador, Eduardo Zancul, por me auxiliar por todo esse processo e me ajudar a achar meu caminho quando eu estava mais perdido.

Ao meu companheiro de aventuras, Dani, pelas várias conversas e reuniões sem as quais esse trabalho não teria sido possível.

Ao meu chefe, Victor Lira, que além de mentor e colega de trabalho, eu considero um grande amigo.

A todos da Flowsense, que me dão imenso orgulho da empresa e sem os quais o projeto não faria sentido.

Aos professores da Escola Politécnica, pelos diversos ensinamentos em engenharia e, além disso, por me possibilitarem fazer este trabalho e muito mais.

Aos membros da banca, pelo tempo e esforço de se aprofundar nesse trabalho.

Aos meus amigos que sempre me ajudaram e estiveram lá quando precisava.

RESUMO

Este trabalho tem como objetivo analisar a criação de um mecanismo para adequar a comunicação das empresas com aplicativos para cada usuário, de forma que consigam manter o engajamento do usuário com ele por meio de mensagens que considere mais relevantes. O estudo justifica-se pela necessidade da empresa focal deste trabalho, a Flowsense, auxiliar seus clientes a manterem essa comunicação eficiente com seus usuários. A base teórica do estudo são as teorias de Comunicação e Marketing de Kotler e, mais especificamente, teorias sobre *Mobile Marketing* como as de Kaplan e Shankar, além de técnicas e teorias de criação de produtos como as propostas por Baxter e pela metodologia de *Design Thinking*. O desenvolvimento do experimento foi descrito e analisado, com considerações críticas a respeito de seus resultados. A partir dos resultados, foi feita uma análise sobre o procedimento e então apresentadas as conclusões e os aprendizados.

Palavras-chave: *Mobile Marketing*, Desenvolvimento de Produtos, engajamento de usuários.

ABSTRACT

This work aims to analyze the creation of a mechanism that adapts the communication of companies with mobile apps to each user, so that they can maintain the user's engagement with them through messages that they consider more relevant. The study is justified by the need of the focal company of this work, Flowsense, to help its customers to maintain this efficient communication with their users. The theoretical basis of the study are Kotler's Communication and Marketing theories and, more specifically, Mobile Marketing theories such as Kaplan and Shankar's, as well as product creation techniques and theories such as those proposed by Baxter and the Design Thinking methodology. The development of the experiment was described and analyzed, with critical considerations regarding its results. From the results, an analysis of the procedure was performed and then the conclusions and lessons learned were presented.

Key words: *Mobile Marketing*, Product Development, user's engagement

LISTA DE ILUSTRAÇÕES

Figura 1 - Fluxo de Informações de APIs	33
Figura 2 - Logo da Empresa Alvo	36
Figura 3 - Proposta de Valor da Flowsense.....	37
Figura 4 - Ações para análise e desenvolvimento do produto.....	39
Figura 5 - Diagrama de Parametrização do Tempo.....	45
Figura 6 - Diagrama de Desenvolvimento do Produto Inicial.....	47
Figura 7 - Diagrama de Desenvolvimento do Produto Final.....	57
Figura 8 - Gráfico de Variação do CTR de acordo com CTR original	62
Figura 9 - Comparação do CTR com métrica AUC ROC.....	63

LISTA DE TABELAS

Tabela 1 - Tabela-resumo dos resultados em relação aos conceitos centrais de Marketing	59
Tabela 2 - Tabela-resumo dos resultados em relação aos conceitos centrais de Mobile Marketing	60
Tabela 3 - Tabela-resumo dos resultados em relação aos conceitos centrais de Produto	61
Tabela 4 - Tabela-resumo dos resultados em relação aos conceitos centrais de Desenvolvimento de Produto	61

LISTA DE ABREVIATURAS E SIGLAS

AMA	American Marketing Association
API	Application Programming Interfaces
APP	Application (software para dispositivos eletrônicos)
AUC	Area Under Curve
AWS	Amazon Web Services
CEO	Chief Executive Officer
CTR	Click-Through-Rate (Taxa de Clique)
DP	Desenvolvimento de Produto
EC2	Elastic Compute Cloud
ECR	Elastic Container Registry
GDPR	General Data Protection Regulation
GPS	Global Positioning System
LGPD	Lei Geral de Proteção de Dados
MM	Mobile Marketing
MMS	Multimedia Messaging Service
ROC	Receiver Operating Characteristic
SDK	Software Development Kit
SIM	Sistema de Informações de Marketing
SMS	Short Message Service
SQL	Structured Query Language
S3	Simple Storage Service
TI	Tecnologia da Informação
UTC	Universal Time Coordinated
WOM	Word Of Mouth

SUMÁRIO

1. INTRODUÇÃO	13
2. REVISÃO DA LITERATURA	17
2.1 Marketing	17
2.2 <i>Mobile Marketing</i>	22
2.3 Desenvolvimento de Produto	29
2.4 <i>Machine Learning</i>	31
2.5 <i>Endpoint</i>	32
2.5.1 APIs	33
2.5.2 Contêiner e Docker	33
2.5.3 Flask	34
2.5.4 Cache	34
2.5.5 Sentry	34
2.6 Serviços da AWS	34
2.6.1 EC2	35
2.6.2 S3	35
2.6.3 Lambda	35
2.6.4 ECR	35
2.6.5 SageMaker	35
3. MATERIAIS E MÉTODOS	36
3.1 A empresa Flowsense	36
3.2 Método	38
3.2.1 Análise do problema	39
3.2.2 Levantamento de soluções	40
3.2.3 Criação de protótipo inicial	40
3.2.4 Iteração sobre o protótipo	40
3.3 Materiais	41
3.3.1 Dados internos da Flowsense	41
3.3.2 Ferramentas da AWS	41
3.3.3 Algoritmos de <i>Machine Learning</i>	41
3.3.4 Sentry	41
4. RESULTADOS	42

4.1 Identificação e Definição do Problema	42
4.2 Produto Inicial	43
4.3 Produto Final	48
4.3.1 Problemas do protótipo inicial	48
4.3.2 Soluções Propostas	50
4.3.3 Arquitetura da Solução	53
5. DISCUSSÃO	59
5.1 Resumo dos resultados em relação à teoria estudada	59
5.2 Resultados Atingidos pelo Delivery Inteligente	62
5.3 Avaliação Crítica	63
5.4 Limitações da Solução.....	64
5.5 Continuidade do Projeto	65
6. CONCLUSÕES.....	67
REFERÊNCIAS BIBLIOGRÁFICAS	69

1. INTRODUÇÃO

É notório que, para qualquer tipo de negócio, alguma forma de comunicação com os clientes necessita ser pensada para que conheçam o produto ou o serviço oferecido por determinada empresa. Este é o princípio básico de Marketing que, segundo Kotler (1991, p. 32), pode ser definido como “um processo social e administrativo pelo qual os indivíduos e grupos obtêm o que necessitam e o que desejam através da criação e troca de produtos e valor com outras pessoas”.

Desde os primeiros hominídeos desenvolvendo algum tipo de linguagem para informar seus colegas que havia comida até as elaboradas campanhas de marketing de hoje, muito mudou. Com a evolução das tecnologias e dos costumes, diversos modos e canais diferentes foram utilizados para comunicação: do boca-a-boca, passando por cartazes, anúncios em jornais, telefone, rádio e televisão, até a diversidade de mídias utilizadas nos dias de hoje, inclusive em dispositivos móveis.

Em maio de 1999, o primeiro dispositivo móvel capaz de oferecer serviços baseados em localização foi lançado e, com ele, segundo Kaplan (2012), os usuários conseguiam saber a previsão de tempo e de trânsito de acordo com a localização oferecida pelo CEP. O autor, no entanto, explica que, provavelmente, foi depois do surgimento do iPhone, em 2007, que o Marketing realmente se tornou móvel. Desde então, os *smartphones* têm oferecido acesso a mais de 250.000 aplicativos, proporcionando, por exemplo, leitura de livros (*Kindle app*), pesquisa a informações (*Wikipedia app*) e contato com amigos (*Facebook app*) (KAPLAN, 2012). Assim, é possível afirmar que *smartphones* e outros dispositivos móveis proporcionam a seus usuários acesso quase que ilimitado aos mais diversos tipos de aplicativos, de forma individualizada e quase que constante, o que significa, também, que marcas contam com notificações via e-mail, *push* e *inapp messages*, canais de comunicação presentes nesses *smartphones*, para construir e manter relações com seus clientes (LEANPLUM, 2020).

Esse método de comunicação tem sido denominado *Mobile Marketing*, que, de acordo com Kaplan (2012, p. 130), pode ser definido como “uma atividade de marketing conduzida por meio de uma rede ubíqua na qual consumidores estão constantemente

conectados ao usarem dispositivos móveis pessoais”¹. Essa estratégia, porém, apesar de garantir uma forma de comunicação acessível das marcas com os clientes, pode não bastar. As marcas podem ser ignoradas se a comunicação não for eficiente.

De acordo com pesquisa realizada pela plataforma de *Mobile Marketing* Leanplum (2020), a principal razão de usuários deletarem aplicativos é por receberem notificações excessivas, especialmente de mensagens irrelevantes. Os clientes se sentem incomodados e ignoram ou optam por excluir ou sair do aplicativo.

A empresa focal deste estudo, que também é local de trabalho deste autor, a Flowsense², vem realizando trabalhos com *Mobile Marketing* há cerca de cinco anos, auxiliando empresas de diversos ramos a melhor se comunicarem com usuários de seus aplicativos. Utilizando aprendizados em diversas disciplinas no curso de Engenharia de Produção da Escola Politécnica da Universidade de São Paulo, o autor pôde aplicar vários conceitos estudados, principalmente de Desenvolvimento de Produtos, em uma situação real do contexto da empresa.

O primeiro contato deste autor com o Desenvolvimento de Produtos ocorreu quando cursou a disciplina optativa *0303410 - Desenvolvimento Integrado de Produtos*. Nela, percebeu a importância da criação de um produto a partir das necessidades dos clientes e de diferentes técnicas, tais como: coletar informações sobre os clientes; processar as informações como necessidades dos clientes; converter essas necessidades em requisitos de produtos; e criar produtos a partir desses requisitos. Além disso, incluiu-se também a proposta de múltipla prototipação para poder incorporar testes e feedbacks no produto em si. Essas foram ideias com as quais o autor se identificou durante o curso e o motivou a buscar outras disciplinas com abordagens e temas semelhantes, como *PRO3381 - Design de Serviços* e *0303212 - Criação de Negócios Tecnológicos*.

Em 2018, o autor conheceu a Flowsense depois de o CEO da empresa apresentá-la em uma disciplina que o autor deste trabalho cursava. A ideia de trabalhar com

¹ “(...) we define mobile marketing as any marketing activity conducted through a ubiquitous network to which consumers are constantly connected using a personal mobile device” (KAPLAN, 2012, p. 130). Tradução pelo autor.

² A empresa autorizou o uso de seu nome e de suas informações para o desenvolvimento deste trabalho.

programação e análise de dados em um ambiente amigável pareceram muito interessantes para o autor, que, após conhecer mais sobre a empresa e descobrir que havia uma vaga na equipe, logo buscou realizar lá seu estágio obrigatório.

Na Flowsense, começou seu trabalho como estagiário na área de Ciências de Dados, onde fazia análises a partir de grandes quantidades de dados. Hoje, o autor trabalha na mesma área como efetivo, e adquiriu experiência em trabalhar com diversos tipos de dados.

A partir de análises de dados sobre respostas de campanhas, junto a comentários de clientes e pesquisas externas, foi possível identificar um padrão no comportamento dos usuários. Eles deixavam de interagir com as campanhas de *Mobile Marketing* se houvesse uma sobrecarga de envios. Esses clientes desativavam as notificações ou até desinstalavam o aplicativo quando se sentiam incomodados com o excesso de notificações.

Com isso, a função que o autor exerce trouxe a necessidade de desenvolver um mecanismo para balancear o número de mensagens enviadas, partindo da motivação de possibilitar a melhoria dos serviços da empresa a fim de atender a demanda de seus clientes sem sobrecarregar usuários finais e tendo maior sucesso. Dessa motivação surgiu este trabalho.

Percebeu-se, então, que existia a necessidade de um fino equilíbrio de forma a manter o cliente informado e engajado sem sobrecarregá-lo com dados e mensagens (LEANPLUM, 2020). Neste trabalho, o objetivo principal foi analisar a criação de um mecanismo que mantivesse este equilíbrio, procurando entender o perfil dos usuários, os interesses das empresas e as melhores formas de conciliar os dois, por meio de estratégias de desenvolvimento de produto e conceitos de marketing. Esse mecanismo é um produto oferecido pela Flowsense e seu sucesso é avaliado por meio da métrica Taxa de Clique (*Click-Through-Rate* ou CTR), isto é, o número de cliques recebidos por um anúncio dividido pelo número de vezes que ele foi exibido.

Para expor o trabalho prático desenvolvido na empresa Flowsense e o analisar a partir de relevante base conceitual, o presente estudo está dividido em cinco capítulos distintos. Nesta primeira, foi feita uma breve introdução ao trabalho com detalhamento

da empresa-alvo e a motivação do autor. O segundo capítulo versará sobre a revisão bibliográfica e apresentará a visão geral de marketing que embasa o estudo, assim como discutirá sobre *Mobile Marketing* e fundamentará as metodologias utilizadas para Desenvolvimento de Produto. No terceiro, será descrito, em detalhes, o projeto realizado, exibindo o levantamento e a organização de dados, as alternativas de solução e a solução selecionada, e, também, a implementação da solução e o acompanhamento de sua execução. O quarto capítulo avaliará criticamente a solução e os resultados atingidos ou esperados. Por fim, as considerações finais irão apontar inovações alcançadas e possíveis melhorias no processo, além de elencar as aprendizagens realizadas com o processo de trabalho e apontar os próximos passos do produto.

Ao longo do processo deste trabalho, pôde-se notar que todos os artigos publicados sobre a otimização da comunicação por meio de notificações em *push* estavam alinhados ao interesse do usuário, não aos interesses dos aplicativos que enviam as notificações, como é o caso deste trabalho. Apesar de semelhantes, as métricas de “sucesso” são diferentes. Embora na perspectiva dos aplicativos seja importante que o usuário se sinta satisfeito e confortável, eles também buscam que o usuário interaja com o aplicativo.

Essa diferença de perspectiva também é relevante quando consideramos o tipo de dados disponíveis para a análise. Esses estudos possuíam uma riqueza enorme de dados para cada cliente pois podiam verificar a interação do usuário com diversos aplicativos, além de terem tido permissões de coleta de vários dados sensíveis e até inputs diretos dos usuários. Isso se deu pois os usuários eram voluntários (ou até recebiam compensação). No caso deste trabalho, só se tem as interações do usuário com cada aplicativo cliente da Flowsense individualmente (mesmo que um usuário possua aplicativos de mais de um cliente, análises cruzadas possuem diversas restrições em virtude da LGPD e de contratos com os clientes) e apenas algumas permissões consideradas importantes para o funcionamento do aplicativo são concedidas (em geral apenas dados de GPS, e mesmo esses estão cada vez mais limitados e sendo menos concedidos).

Ao abordar essa outra perspectiva e propor uma solução que busca aumentar as chances de interação com o usuário considerando esses dados que conseguimos acesso, este trabalho se diferencia do conteúdo produzido previamente e explora o processo de criação deste produto.

2. REVISÃO DA LITERATURA

Nesta seção, serão apresentados estudos já realizados sobre os tópicos centrais a este trabalho: Marketing, *Mobile Marketing* (MM) e Desenvolvimento de Produto (DP). A parte inicial sobre Marketing terá como foco uma visão holística da temática, o que permitirá uma melhor compreensão do objeto específico do estudo: *Mobile Marketing*. Um foco especial foi dado ao conceito de Pesquisa de Marketing, que explica a metodologia tradicional pela qual se entendia os consumidores a quem o Marketing é direcionado. Na parte sobre MM, serão apresentadas considerações sobre o que se entende na pesquisa e no mercado sobre MM e como esse tema tem se mostrado central à vida de empresas de vários tipos. Por sua vez, na seção sobre DP, primeiro serão discutidas as principais metodologias e requisitos para esta atividade. Na sequência, de forma específica, estudos sobre DP na área de MM serão descritos e seus resultados interpretados frente às demandas deste estudo.

2.1 Marketing

De acordo com Kaplan (2012, p. 130), *Mobile Marketing* (MM) pode ser definido como “uma atividade de marketing conduzida por meio de uma rede ubíqua na qual consumidores estão constantemente conectados ao usarem dispositivos móveis pessoais”. Levando tal definição em consideração, é necessário, primeiro, entender o que é marketing para, então, discorrer sobre MM.

Conforme Kotler (1991), no sentido mais genérico, marketing seria a busca pelo interesse das pessoas por algum objeto. Nesse sentido, ele apresenta quatro meios de os seres humanos obterem produtos que querem: produção própria, coerção, súplica e troca, sendo a troca o conceito que define marketing. Para existir uma troca, o autor destaca a necessidade de haver ao menos duas pessoas envolvidas e cada uma delas dispor de algo que a outra possa considerar de valor. Ademais, também é condição o fato de cada indivíduo ser capaz de se comunicar e realizar a troca, cada um ter a liberdade de aceitar ou rejeitar a oferta e cada um crer ser desejável ou apropriado negociar com o outro. Kotler (1991) pontua que

a troca pode ocorrer caso sejam satisfeitas essas condições e as pessoas envolvidas concordarem sobre os seus termos e considerarem que ela as deixaria melhor do que antes.

Kotler (1991, p. 38) continua expondo que o conceito de troca leva ao conceito de mercado, uma vez que o mercado é “composto por clientes potenciais que partilham de uma mesma necessidade ou desejo e estão aptos a engajar-se num processo de troca para satisfazer aquela necessidade ou desejo.” Esse conceito torna-se relevante para a discussão do que é marketing à medida que é importante adequar o marketing ao mercado que se procura atingir.

Marketing, então, é o trabalho realizado para que essas trocas ocorram. Em seu livro, Kotler (1991) define marketing como “um processo social e gerencial pelo qual indivíduos e grupos obtêm o que necessitam e desejam através da criação e troca de produtos e valor com outras pessoas”.

Levitt (1960) estabelece, em seu clássico trabalho *Marketing Myopia*, a distinção entre conceitos de venda e marketing. Enquanto vendas foca as necessidades do vendedor, isto é, converter seu produto em dinheiro, marketing foca as necessidades do comprador por meio de ênfase no mercado, orientação para o cliente, marketing integrado (toda empresa trabalhando para o cliente) e rentabilidade (vendedor lucrar).

Os vendedores são entendidos como a indústria e os compradores como o mercado. Vendedores e compradores são ligados por quatro fluxos: vendedores enviam bens ou serviços e comunicação ao mercado e recebem em troca dinheiro e informação.

Lidar com processos de troca requer trabalho e competência. Daí aparece a administração de marketing como um processo de planejamento e execução da concepção, do preço, da promoção e da distribuição de ideias, bens e/ou serviços para criar trocas que satisfaçam objetivos individuais e organizacionais. Tal definição foi concebida pela American Marketing Association (AMA) de acordo com Kotler (1991). Portanto, o principal objetivo da administração de marketing é influenciar o nível, o tempo e a composição da demanda para permitir à organização obter suas metas. Conclui-se que administrar o marketing é administrar a demanda, que pode ser negativa, inexistente, latente, declinante, irregular, plena, excessiva ou mesmo indesejada. Para enfrentar essas diferentes situações da demanda, a administração de marketing utiliza pesquisa de marketing, planejamento, implementação e controle.

Uma das maiores dificuldades dos gestores (administradores) de marketing é não ter as informações adequadas para tomar as devidas decisões. Naisbitt (1982) em *Megatrends* diz que não há perigo de se ficar sem informações, mas sim de ser afogado por elas. Para equilibrar e conciliar o que os profissionais de marketing gostariam de obter com o que eles necessitam e com o que é possível oferecer se desenvolveu o Sistema de Informações de Marketing (SIM) de acordo com Kotler e Armstrong (1999). Os dois autores comentam que os gestores de marketing nem sempre podem esperar que as informações a partir do SIM estejam disponíveis dentro da sua organização. Portanto, solicitam estudos formais de situações específicas. Esses estudos se denominam pesquisas de marketing, que são os instrumentos que conectam o consumidor alvo, o cliente e o público ao gestor de marketing por meio de informações. Tais informações são usadas para identificar e definir oportunidades e problemas de marketing, gerar, refinar e avaliar as atividades, acompanhar seu desempenho e aumentar a compreensão do seu próprio processo.

Kotler e Armstrong (1999) continuam, explicando que todo o profissional de marketing depende dessas pesquisas, que podem ser sobre o potencial e a participação de mercado, avaliação da satisfação do cliente ou seu comportamento de compra. Elas permitem atividades como apreçamento e avaliação do produto, da distribuição e da promoção. A empresa pode realizar pesquisas de marketing por conta própria ou pode contratar terceiros. A decisão da contratação de terceiros depende de recursos da empresa e da capacidade de seus funcionários executarem a contento a atividade objetivada, tanto em termos de qualificação profissional como de disponibilidade de recursos técnicos. As informações coletadas pelas pesquisas de marketing geralmente necessitam análises mais profundas. Essas análises podem requerer ajuda de especialistas para aplicação aos problemas e para tomada de decisão. De acordo com Kotler e Armstrong (1999) são quatro etapas do processo de pesquisa de marketing: (1ª.) definição do problema e dos objetivos da pesquisa; (2ª.) implementação; (3ª.) interpretação e (4ª.) apresentação dos resultados.

A definição do problema e dos objetivos da pesquisa é o primeiro e o passo mais difícil do processo. Após uma cuidadosa definição do problema, passa-se para a definição dos objetivos da pesquisa, que podem ser uma pesquisa exploratória – reunindo informações preliminares que auxiliem a definir o problema e propor hipóteses; ou

pesquisa descritiva – descrevendo fenômenos ou objetos e pesquisa causal – testando hipóteses sobre relações de causa e efeito.

O segundo passo do processo é a identificação da informação necessária, o desenvolvimento de um plano para coletá-la com eficácia e a apresentação do plano ao gestor de marketing. O plano estabelece as fontes de dados existentes e detalha os métodos específicos da pesquisa, assim como as formas de contato, os planos de amostragem e os instrumentos que os pesquisadores usarão para reunir os dados.

Os objetivos da pesquisa precisam ser traduzidos por necessidades específicas de informação. Essas necessidades podem implicar na coleta de dados primários – com o propósito específico da pesquisa; secundários – informações já existentes em algum lugar e coletadas com outro propósito; ou ambos os dados – que geralmente se começa a coleta com dados secundários. Boas decisões necessitam de dados adequados. Assim como o pesquisador precisa avaliar com cuidado a qualidade da informação secundária, precisa também cuidar da coleta dos dados primários para assegurar sua relevância, precisão, atualidade e imparcialidade.

A elaboração de um plano para coleta de dados primários requer decisões sobre o método de pesquisa a ser usado, formas de contato, plano de amostragem e instrumentos de pesquisa.

Segundo Kotler e Armstrong (1999), os métodos de pesquisa mais usados são: **pesquisa por observação** – consiste na coleta de dados primários por meio de observação de pessoas, ações e situações relevantes; **levantamento** – método mais usado à coleta de informações descritivas por meio de perguntas diretas aos consumidores sobre seu conhecimento, atitudes, preferências ou comportamento de compras; **pesquisa experimental** – mais indicada para a coleta de informações do tipo causal, envolvendo seleção de grupos experimentais, tratamento diferenciado desses grupos, controle de fatores aleatórios e checagem das diferenças nas respostas dos grupos.

Os critérios mais utilizados para escolha das formas de contato são flexibilidade, quantidade de dados que podem ser coletados, controle das interferências do entrevistador, controle da amostra, rapidez da coleta de dados, taxa de resposta e custo.

A escolha da melhor forma de contato depende do tipo de informação desejada e do número e dos tipos de entrevistados a serem contatados.

As pesquisas de marketing podem chegar a conclusões sobre grandes grupos de consumidores por meio de uma pequena amostra da população total. Amostra é um segmento da população selecionado para representar a população como um todo. Busca-se uma amostra representativa do universo para que se possa fazer estimativas mais precisas das ideias e dos comportamentos da população. Três decisões se fazem necessárias para definição da amostra: (1) quem será entrevistado; (2) qual o tamanho da amostra; (3) que procedimento amostral será utilizado.

Ao serem coletados os dados primários, há possibilidade de escolha de instrumentos de pesquisa de acordo com o envolvimento direto buscado entre o pesquisador e o pesquisado.

O pesquisador de marketing deve resumir o plano de pesquisa em uma proposta por escrito, sendo importante quando o projeto é grande e complexo ou quando é executado por terceiros. Normalmente se incluem os problemas de gerenciamento levantados e os objetivos da pesquisa, a informação a ser obtida, as fontes de informação secundária ou métodos para coleta de dados primários e a forma pela qual os resultados irão ajudar os gestores a tomar suas decisões, bem como os custos da pesquisa. A apresentação por escrito alinha os aspectos importantes da pesquisa entre pesquisador e cliente, assim como concordam com a sua finalidade e sua forma de ser realizada.

O próximo passo é colocar o plano de pesquisa de marketing em ação, o que envolve a coleta, o processamento e a análise da informação. Finalmente, o pesquisador deve interpretar os resultados, tirando conclusões e as apresentando ao cliente de forma inteligível, para contar com seu envolvimento por conhecer melhor os problemas e por ser responsável em implementar as decisões.

Com o decorrer dos anos, as novas tecnologias e paradigmas de comunicação foram se alterando e, nesse sentido, também o marketing e as suas pesquisas. Hoje, 46% dos gastos em Marketing são especificamente com Marketing Digital (VERBLOW *et al.*, 2020). Segundo Ablas (2020), o Marketing Digital tem como vantagem a alta interatividade, a velocidade e a possibilidade de conhecer melhor as preferências dos

consumidores. O meio digital permite também a coleta de diversas informações que são fornecidas pelo consumidor de maneira indireta. Essas informações são o que Kotler e Armstrong (1999) chamam de dados secundários por não terem sido criados com o propósito específico em questão. Segundo esses autores, apesar de não serem tão específicos ou detalhados quanto dados primários, dados secundários são geralmente mais baratos e adquiridos com maior velocidade.

Vale notar que recentemente há um movimento para a limitação das informações coletadas a fim de preservar a privacidade do usuário. Escândalos envolvendo vazamento de informações pessoais e manipulação de consumidores são cada vez mais divulgados e causam reprovação veemente. Um exemplo com repercussão mundial é o caso da empresa Cambridge Analytica com relações escusas com o Facebook, que implicou em revelações de que os consultores digitais da campanha de Trump usaram indevidamente, isto é, sem permissão, os dados de milhões de usuários do Facebook e que foi decisivo na vitória trumpista da eleição presidencial estadunidense (CONFESSORE, 2018).

Para coibir tais práticas, houve uma reação estruturada dos legisladores em diferentes países que culminaram na criação de diversas leis que protegeriam o usuário. O movimento se iniciou na União Europeia com a *General Data Protection Regulation* (GDPR) e se apresenta no Brasil sob o nome de Lei Geral de Proteção de Dados (LGPD). No entanto, mesmo com essa limitação, o marketing digital continua em alta e atingindo diversas pessoas.

2.2 Mobile Marketing

Como já exposto, com os avanços da tecnologia usada pelas pessoas, a área de Marketing precisou se adaptar aos novos canais. Os aparelhos móveis se tornaram uma ferramenta tão presente na vida das pessoas que uma nova área de Marketing, conhecida como *Mobile Marketing*, teve de ser criada. Considerada uma estratégia de marketing digital multicanal, o *Mobile Marketing* utiliza-se de *websites*, *e-mail*, SMS, MMS, mídias sociais e *apps*, presentes em *smartphones*, *tablets* e outros aparelhos móveis, para atingir seu público alvo (KANAGAVALLI, 2019).

Shankar e Sridhar (2009) citam como principais características do *Mobile Marketing* a especificidade de localização, a portabilidade e a conectividade sem fio. Essas características fazem com que surjam diferenças significativas entre o *Mobile Marketing* e o Marketing de massa tradicional, com o *Mobile Marketing* possuindo a possibilidade de uma comunicação em duas vias, no lugar da comunicação unidirecional do marketing de massa, além de poder medir respostas e direcionar a comunicação de forma mais fácil e barata. O problema, porém, é que a audiência do *Mobile Marketing* é reduzida (apenas consideram-se usuários que possuem aparelhos móveis e em geral requerem algum tipo de *opt-in*) e a mídia utilizada em geral precisa ser mais simples por conta das limitações de espaço e banda.

Com isso em mente, pode-se então considerar a definição de *Mobile Marketing* utilizada por Kaplan (2012, p. 130) mencionada no início do capítulo: “qualquer atividade de marketing conduzida por meio de uma rede ubíqua na qual os consumidores estão conectados constantemente através de um aparelho móvel pessoal”. Dessa definição pode-se compreender que uma comunicação de *Mobile Marketing* assume que o canal de comunicação está sempre aberto.

É importante notar, porém, que existem diferentes tipos de comunicação mobile. Kaplan (2012) divide os tipos de comunicação de acordo com duas vertentes: grau de conhecimento sobre o usuário e gatilho da comunicação. O grau de conhecimento do usuário é o quanto uma empresa sabe sobre um cliente, o que permite a customização da mensagem ao usuário específico. O gatilho da comunicação é dividido entre *Push* e *Pull*, sendo *Push* quando a empresa inicia a comunicação e *Pull* quando o cliente vai em busca dela. É importante notar que *Push*, nesse contexto, não tem o mesmo significado das Notificações em *Push* (apesar de ser daí que vem o nome). Um SMS que foi enviado sem que o usuário solicitasse é um exemplo de uma comunicação com gatilho *Push*.

Além disso, o *Mobile Marketing* também pode ser compreendido por seu potencial de enfrentar o grande desafio dos profissionais de marketing hoje: obter atenção de consumidores. Para Barwise e Strong (2002) e Pousttchi e Wiedemann (2007), essa atenção pode ser obtida usando três estratégias básicas - *push*, *pull* (também abordadas por Kaplan) e virais. Para esses autores, em estratégias *push*, os profissionais de marketing direcionam os consumidores para distribuir ou comunicar conteúdo. Em estratégias *pull*, os consumidores solicitam explicitamente o conteúdo dos profissionais

de marketing. Já nas estratégias virais, a troca de informações ocorre entre consumidores. Essa última estratégia é conhecida como mobile boca-a-boca (WOM) ou marketing viral móvel (WIEDEMANN, 2007).

Na prática, segundo Leppäniemi e Karjaluoto (2008), existem três maneiras pelas quais uma empresa pode optar por lidar com suas campanhas de marketing para celular. A empresa pode usar recursos internos ou contar com os serviços de uma ou mais agências (por exemplo, agência de publicidade, agência de publicidade móvel e agência de mídia) ou usar algum tipo de combinação (ou seja, usar agências e dar sequência ao trabalho interno). Como muitas empresas não possuem a tecnologia e os conhecimentos necessários para executar campanhas de marketing móvel, a última opção é o mais comum modo de ação. Além disso, campanhas de marketing móvel geralmente envolvem várias mídias e, portanto, menos ainda as empresas são capazes de desenvolver e manter todas as atividades relacionadas à campanha de marketing móvel sem a necessidade de conhecimento externo. No entanto, um entendimento básico de mídia móvel e campanha de marketing móvel é necessário para desenvolver uma campanha de marketing móvel bem-sucedida.

Uma das métricas mais comuns, que tem atraído a atenção da academia e da indústria, quando se fala de *Mobile Marketing* é o CTR. O *Click-Through-Rate* (CTR), uma medida comumente usada para mensurar o sucesso de uma campanha de marketing digital, apresenta a razão de usuários que clicam em um link específico sobre o número total de usuários que viram a página, o email ou o anúncio (OUYANG *et al.*, 2019). O cálculo do CTR se expressa como a seguir:

$$CTR = \frac{\text{Número de Cliques}}{\text{Número de Exibições}}$$

Segundo Gao (2012), O CTR pode ser calculado de duas formas: a) *Individual Click Through Rate*, em que a porcentagem é apresentada como o número total de usuários que clicaram seus canais de Marketing Digital e b) *Holistic Click Through Rate*, que considera todos os cliques de um de seus canais digitais para um destino mesmo que esses cliques tenham sido feitos por um mesmo indivíduo.

Na proposta de Gao (2012) para emails, considera-se também que os possíveis problemas no número de cliques de uma campanha abaixo da expectativa possam estar conectados a:

1. número insuficiente de links para o usuário clicar,
2. links invisibilizados em imagens que os usuários não conseguem acessar,
3. links formatados de modo precário,
4. *calls-to-action* precários que não engajam ou motivam os usuários,
5. ofertas e conteúdos não atraentes,
6. qualidade ruim da lista dos recipientes da mensagem.

Para melhorar a qualidade do CTR, Gao (2012) sugere algumas ações:

1. auditoria de templates para CTR por melhores práticas para otimizar a atividade,
2. revisão da redação do *Call-to-Action*,
3. testagem e monitoramento de diferentes ofertas e conteúdos dos tipicamente utilizados,
4. avaliação da lista de contatos, considerando seus interesses e engajamento com o produto.

Essas ações permitem que comunicações digitais encontrem maior sucesso.

Sobre notificações em *push* em específico, existem muitas variáveis que podem influenciar no sucesso da comunicação. Diversos artigos buscam compreender quais variáveis são essas e como é possível tirar proveito para aumentar a satisfação dos usuários frente ao envio de *pushes*.

No artigo publicado em 2015 para a UbiComp '15, *Designing Content-driven Intelligent Notification Mechanisms for Mobile Applications*, Mehrota *et al.* detalham um projeto que visava encontrar “momentos oportunos” para enviar notificações, sendo esses momentos em que o usuário leva menos de 10 minutos para interagir com a notificação. Foi notado que o estudo identificava momentos oportunos com maior precisão quando considerava tanto o conteúdo quanto o contexto em que a notificação era enviada.

Outro aprendizado importante desse estudo foi que os modelos de previsão eram mais assertivos quando consideravam as informações individuais de cada usuário, não apenas as informações agregadas.

Em um outro trabalho, *Understanding smartphone notifications' user interactions and content importance*, Visuri *et al.* (2019) criam um aplicativo para tentar entender o interesse de diversos usuários em notificações. O aplicativo pede para que o usuário avalie notificações recebidas de acordo com a importância e o timing. O aplicativo também permite que após 50 avaliações o usuário faça o uso de um administrador de notificações que decide se a mensagem é importante ou não e só mostra as mensagens relevantes.

Essa decisão é baseada tanto em uma série de fatores do contexto da notificação (localização, estado da tela, atividade física, entre outros) como também no conteúdo e na origem da notificação e nas pesquisas feitas com o usuário (com classificações de importância e percepção sobre o momento em que o *push* foi enviado).

Após a análise, uma das informações que pareceu ter um efeito relevante em como o usuário interagia com as notificações era a hora do dia. O efeito era relevante pois no período das 16h às 18h o usuário está mais propenso a clicar.

Outro fator relevante encontrado foi o número de mensagens recebidas anteriormente. É perceptível uma queda na taxa de interação quando o usuário recebe mais de 100 notificações diárias. Também foi possível notar que as chances de um usuário interagir com a notificação é significativamente maior quando ele não recebeu nenhuma outra notificação. Essa análise se repetiu para diversos períodos de tempo analisados (de 1 minuto a 4 horas). Outro efeito notado foi que, se o usuário recebe muitas notificações seguidas, o painel de notificações pode ficar saturado e o usuário pode não interagir com notificações que ele normalmente interagiria (em especial se o usuário deixa de interagir e filtrar informações desnecessárias).

O trabalho de Visuri *et al.* (2019) busca se aprofundar no entendimento comum de que clique=bom e *dismiss*=ruim, mostrando que diversas vezes o usuário não interage com a notificação independente da importância que ele vê nela. Também é apresentado que, apesar de a origem da notificação ser um fator importante nas preferências do usuário, fatores individuais são muito mais relevantes.

Outro artigo de interesse para entender o funcionamento do *Mobile Marketing* é *Dismissed! A Detailed Exploration of How Mobile Phone Users Handle Push Notifications* por Pielot *et al.* (2018). Ele também busca compreender os comportamentos de usuários com notificações de diferentes aplicativos. Esse estudo, porém, é menos detalhado quanto aos fatores e compara principalmente o tempo e o tipo de cada interação de acordo com o tipo de aplicativo que é.

Os resultados da análise feita por Pielot *et al.* (2018) revelam que não houve um número maior de notificações quando comparado a estudos anteriores, eliminando a hipótese de que o número de envios estaria crescendo com o tempo. Apenas aplicativos de mensagens possuíam uma quantidade mais expressiva de notificações.

Outro achado foi que a taxa de conversão (calculada nesse estudo como a porcentagem de usuários que abriram o app antes da mensagem ser removida) de aplicativos que não são de mensagem é muito inferior à de aplicativos de mensagens.

Outro resultado relevante é que entre 20% e 35% das notificações chegam enquanto o celular está travado.

Além disso, foi possível notar com o estudo desses autores que não havia uma diferença significativa entre as diferentes categorias de aplicativos (com exceção dos aplicativos de mensagens) para a conversão. A principal diferença estava na velocidade em que as mensagens eram visualizadas.

Pode-se também observar o artigo *PrefMiner: Mining User's Preferences for Intelligent Mobile Notification Management* de Mehrota *et al.* (2016a) que, assim como o de Visuri *et al.* (2019), relata a criação de um aplicativo que busca reduzir a quantidade de notificações com informações vistas como irrelevantes ao usuário. Nele, Mehrota *et al.* (2016a) apresentam a importância de que um sistema desse tipo funcione corretamente, a fim de evitar que notificações importantes sejam escondidas. Para obter essa garantia, esse trabalho de Mehrota *et al.* (2016a) tem como um diferencial que as regras criadas sejam de fácil interpretação e que os usuários possam optar por aceitá-las ou não. Essas regras são criadas a partir da mineração de associações entre diferentes notificações.

Para considerar as notificações que usuários podem ter interesse em receber, mas não necessariamente interagem, Mehrota *et al.* (2016a) criou a classificação de notificações como lembretes. Para uma regra ser considerada um lembrete, o aplicativo tem que ter um CTR zero (nenhuma notificação clicada).

A classificação das notificações se dá então pelo conteúdo de cada uma. São extraídos os títulos das notificações, que são então limpos e então *clusterizados*. Foram testadas também diferentes informações de contexto, como o horário, localização e atividade do usuário, mas foi verificado como mais eficiente a combinação de informações *clusterizadas* e dados de localização.

Por fim podemos analisar *My Phone and Me: Understanding User's Receptivity to Mobile Notifications*, também de Mehrota *et al.* (2016b). Esse artigo é interessante por buscar entender melhor quais fatores influenciam o tempo de resposta e percepção de interruptibilidade de notificações e ter um foco especial nas características e no contexto do usuário. O estudo tem como variáveis não apenas qualidades da notificação, como horário de recebimento, resposta, conteúdo e procedência da notificação, mas também a coleta de dados como modo de alerta do celular (som, vibrar, mudo) e contexto (localização, intensidade de luz, som ambiente e atividade física), além de dados obtidos por meio de questionários com o usuário (reação à notificação, fatores que levaram ao clique, relação com quem enviou, sensação de quão disruptiva foi e que tipo de tarefa estava sendo realizada) e até dados sobre a personalidade do usuário a partir de testes de personalidade que alguns participantes responderam.

O artigo então lista relações percebidas entre esses dados, especialmente qual o impacto que diferentes variáveis têm na percepção de interruptibilidade, no tempo para que o usuário veja a notificação e o tempo para que ele reaja (clique ou dispense). O texto de Mehrota *et al.* (2016b), então, torna-se em uma série de descrições nessas relações. Algumas das mais relevantes/surpreendentes são:

- Usuários demoram mais para ver notificações quando ociosos
- Quanto mais complexa a atividade que o usuário estiver realizando, mais rapidamente a notificação é vista

- Usuários percebem que uma notificação chegou independente do modo de alerta do celular, mas ele afeta o tempo para o usuário ver a notificação

- Notificações são consideradas mais disruptivas quando chegam quando o usuário está no meio de uma tarefa ou a finalizando

- A personalidade do usuário tem um efeito estatisticamente relevante tanto no tempo de visualização quanto no tempo de resposta e na percepção de interruptibilidade

Com todas essas análises em mente, se torna possível a criação de um produto que leve em conta os diferentes aspectos de Marketing e *Mobile Marketing* para melhorar a comunicação com usuários. Torna-se então necessário voltar a atenção para estratégias e metodologias de desenvolvimento de produto para que este ocorra da melhor maneira possível.

Deve-se destacar que problemas com a implementação (ou produção) de *Mobile Marketing* têm um impacto direto sobre o desenvolvimento e design de uma campanha de *Mobile Marketing* específica. Portanto, é importante fornecer uma breve visão geral dos principais problemas relacionados à implementação de *Mobile Marketing*. O planejamento da implementação do *Mobile Marketing* começa com a identificação e avaliação de potenciais provedores de serviço de marketing móvel. Esta é uma etapa vital no processo de planejamento e prestadores de serviços devem ser avaliados quanto ao serviço que fornecem, os equipamentos utilizados e as referências que eles possuem. Depois que o provedor de serviços for escolhido, o plano de implementação de marketing móvel pode ser desenvolvido.

2.3 Desenvolvimento de Produto

Em seu livro *Projeto do Produto*, Mike Baxter (2000) detalha diversas metodologias que podem ser utilizadas durante o desenvolvimento de produtos. Primeiramente, é necessária a compreensão do problema. Para tal, Baxter (2000) descreve uma ferramenta de Análise do Problema que visa pesquisar a origem do problema, questionando o porquê dos problemas e do interesse em resolvê-los. Com isso, busca-se identificar as Metas e Fronteiras do problema.

Outro aspecto importante é o estabelecimento de metas para o produto de forma a não permanecer em uma eterna busca por oportunidades melhores. Essas metas devem estar alinhadas à estratégia da empresa e em geral se relacionam com a viabilidade comercial do produto.

Baxter (2000) menciona também que as três fontes principais para pesquisar por oportunidades de produtos são:

- Necessidades de mercado (foco deste trabalho)
- Análise de concorrentes
- Auditorias tecnológicas

As necessidades de mercado podem ser levantadas tanto por conhecimentos internos da empresa quanto por pesquisa tanto bibliográfica quanto de campo.

Para o planejamento do projeto, Baxter (2000) diz que se deve primeiro transformar necessidades do consumidor em requisitos do projeto, para, então, transformar requisitos do projeto em especificações do produto e, em seguida, especificações do produto em controle de qualidade. O método formal para se fazer isso é usando desdobramentos da função de qualidade.

Quanto ao produto em si, Mike Baxter (2000, p.233) comenta que “um produto pode ser descrito em termos funcionais ou físicos”. Em seu livro, descreve também arquitetura modular como um produto em que cada módulo (conjunto de elementos físicos) é responsável por um ou mais elementos funcionais de forma completa.

O autor (2000) diz que só se deve fazer protótipos se já tiverem sido esgotadas todas as outras fontes de informação, porém outras metodologias discordam dessa abordagem visto o quanto protótipos são interessantes e ricos em informações desde que realizados corretamente. Para isso, uma abordagem mais semelhante à do *Design Thinking* foi utilizada quando considerando protótipos.

O *Design Thinking* é uma metodologia que foi muito influente para a realização deste estudo. Como o nome sugere, ela é uma metodologia de criação de produtos baseada em conceitos da área de design. Em seu artigo *Design Thinking*, Tim Brown (2008) define *Design Thinking* como:

uma disciplina que utiliza a sensibilidade e métodos do designer para casar as necessidades das pessoas com aquilo que é tecnologicamente possível e que uma estratégia de negócios pode converter em valor para o consumidor e em oportunidade de mercado.

Essa definição evidencia que a criação do produto parte das necessidades reais das pessoas para então considerar o que é possível e viável, e não o contrário, como é comum.

Brown (2008) divide *Design Thinking* em três etapas: Inspiração, Ideação e Implementação. Na Inspiração, entende-se qual é o problema a ser resolvido, quais as necessidades dos clientes e quais as limitações do projeto. Na etapa de Ideação, existe um esforço para gerar, desenvolver e testar possíveis soluções. A Implementação, por sua vez, consiste na transformação dessas ideias em uma solução de mercado.

O método de *Design Thinking* é cíclico e iterativo. Conforme as ideias e soluções são testadas e novos aprendizados são tomados a partir dos protótipos criados, novos ciclos de Inspiração e Ideação são feitos para buscar novas ideias e melhorias. Liedtka (2018) explica que o processo de prototipação não trata de ajustes finos em um serviço já pré-estabelecido, mas sim em experiências iterativas dos usuários com produtos ainda em produção.

Além de estratégias de design de produtos é importante também que sejam exploradas diferentes técnicas, tecnologias e serviços que podem ser utilizados para a criação de um produto que atenda aos objetivos do presente trabalho.

2.4 Machine Learning

De acordo com Jordan e Mitchell (2015, p. 255), a definição de *Machine Learning*, um dos campos técnicos de crescimento mais rápido da atualidade, pode partir de duas questões inter-relacionadas:

Como construir sistemas de computador que melhorem automaticamente com a experiência? E quais são as leis teóricas da informação estatística computacional fundamentais que governam todos os sistemas de aprendizagem, incluindo computadores, humanos e organizações?³

³ Tradução livre de: “How can one construct computer systems that automatically improve through experience? and What are the fundamental statistical-computational-information-theoretic laws that govern all learning systems, including computers, humans, and organizations?”

Esse tipo de aprendizagem é fundamental para abordar questões científicas e para considerar a produção de softwares de computador com múltiplas aplicações. Além disso, é uma área situada na interseção da ciência da computação e da estatística, e no cerne da inteligência artificial e ciência de dados. Esse tipo de aprendizagem é impulsionado pelo desenvolvimento de novos algoritmos e pela explosão contínua de disponibilidade de dados online e de baixo custo.

Machine Learning é descrita por Mitchell (1997) como um campo que estuda a criação de programas de computador que melhoram sua performance por meio de experiência. Ele também menciona como perspectiva interessante a de que *Machine Learning* é uma busca sobre um grande número de hipóteses para determinar a que melhor se adequa aos dados coletados.

Segundo Dietterich (1997), *Machine Learning* oferece técnicas aplicáveis a novos tipos de problema, incluindo descoberta de conhecimento em bancos de dados, processamento de linguagem, controle de robô e otimização combinatória, reconhecimento de fala, reconhecimento de rosto, reconhecimento de caligrafia, análise de dados médicos e jogos.

Para o uso de um produto que emprega *Machine Learning* é importante entender como essa estará conectada ao resto da arquitetura do produto. Um dos métodos possíveis é o uso de um endpoint que exponha o modelo treinado para a previsão ou classificação de dados fornecidos.

Em termos técnicos, entre os paradigmas de *Machine Learning* mais comuns estão o Gradiente Descendente Estocástico, as Redes Neurais e a Previsão Estruturada.

2.5 Endpoint

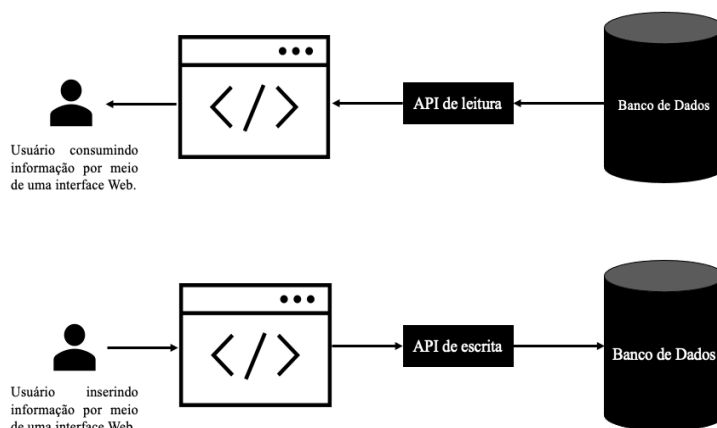
Um *endpoint* é um dos pontos terminais de uma comunicação (SMARTBEAR, 2021). No contexto deste trabalho o *endpoint* é a estrutura que fornecerá as informações necessárias para outros sistemas por meio de uma API. Alguns dos principais aspectos do Endpoint tratado neste trabalho estão descritos ao longo desta seção.

2.5.1 APIs

Application Programming Interfaces (APIs) organizam grandes volumes de informações (JANETZKO, 2017). As APIs são como "colas dinâmicas" que unem diferentes sistemas computacionais. Pode-se dizer, dessa forma, que APIs são intermediários que permitem que duas ou mais aplicações troquem informações.

Uma aplicação moderna é tipicamente construída sobre algumas APIs, enquanto estas ainda podem depender de outras APIs. O objetivo dessa construção é prover uma interface lógica à funcionalidade de um componente ao mesmo tempo que esconde os detalhes de implementação (REDDY, 2011). Em um exemplo simplificado, é possível desenvolver APIs para leitura de arquivos ou ainda para escrever novas informações em um Banco de Dados para interagir com um usuário por meio de uma aplicação web. A Figura 1 abaixo busca apresentar esse fluxo de informações.

Figura 1 - Fluxo de Informações de APIs



Fonte: Autor

2.5.2 Contêiner e Docker

Um contêiner é uma unidade de software que contém código e todas suas dependências, de forma que permite que sua aplicação rode de maneira eficiente (DOCKER, 2021). Lachman (2017) explica que os contêineres usam um aplicativo cliente-servidor chamado de engine de containerização que permite a separação entre as

camadas de contêineres e o kernel da máquina. O Docker é não apenas uma engine de containerização como todo um ecossistema de ferramentas.

2.5.3 Flask

O Flask é um *framework* para aplicativos de web baseado na linguagem *Python*. Ele faz parte do *Pallets Project*, um conjunto de bibliotecas para desenvolvimento web criadas originalmente por Armin Ronacher e mantidas atualmente pela comunidade. Grandes nomes como Netflix, Patreon e Airbnb utilizam esse *framework* para seus serviços (FLASK, 2021; THE PALLETS PROJECTS, 2021a, 2021b).

2.5.4 Cache

Cache é definido pelo dicionário Michaelis Online (2021) como “Área de memória rápida que armazena os últimos dados manuseados (carregados na memória RAM), deixando-os disponíveis para serem acessados rapidamente”. A documentação do Mozilla (MDN WEB DOCS, 2021) explica que cache é uma técnica que guarda uma cópia de um dado recurso e o serve quando requisitado. A próxima vez que esse recurso for chamado ao invés de solicitar ao servidor o recurso é copiado do cache, o que permite velocidade muito maior e diminui a carga de solicitações no servidor.

2.5.5 Sentry

O Sentry é um serviço de monitoramento de software que permite um acompanhamento das ocorrências de erros com mais detalhes sobre o contexto em que eles ocorreram. Ele permite a criação de alertas para que se tenha mais visibilidade e fornece uma visualização de forma organizada que permite o tratamento mais eficiente dos erros (SENTRY, 2021).

2.6 Serviços da AWS

Para a implementação de todas essas peças da arquitetura, foram utilizados diversos serviços da Amazon Web Services (AWS). Muitos desses serviços já eram utilizados pela Flowsense e foram expandidos para acomodar o produto aqui descrito. É importante para a compreensão do trabalho obter um certo grau de familiaridade com algumas das ferramentas que foram utilizadas e, para isso, informações sobre elas foram coletadas em seu site.

2.6.1 EC2

O Amazon Elastic Compute Cloud (Amazon EC2) é um serviço de computação na nuvem. A própria Amazon define computação na nuvem como “a entrega de recursos de TI sob demanda por meio da Internet com definição de preço de pagamento conforme o uso.” O EC2 possui uma série de recursos que permitem flexibilidade ao uso desejado e segurança nas conexões à máquina virtual (AMAZON, 2021a,2021b).

2.6.2 S3

O Amazon Simple Storage Service (Amazon S3) é um serviço de armazenamento na nuvem. Ele possui diferentes configurações de acordo com o tipo de acesso que se necessita, além de permitir escalabilidade, disponibilidade e segurança (AMAZON, 2021c).

2.6.3 Lambda

O AWS Lambda, assim como o EC2, é um serviço de computação na nuvem, mas tem como diferencial que não utiliza servidores. Isso permite que o dimensionamento possa ser contínuo, adequando-se à carga de trabalho requerida, além de simplificar o gerenciamento por fazer o gerenciamento e provisionamento de recursos automaticamente (AMAZON, 2021f).

2.6.4 ECR

O Amazon Elastic Container Registry (ECR) é um registro de contêiner totalmente gerenciado que facilita o armazenamento, o gerenciamento, o compartilhamento e a implantação de imagens e artefatos de contêiner em qualquer lugar (AMAZON, 2021e).

2.6.5 SageMaker

O Amazon SageMaker é um serviço de *Machine Learning* que auxilia em todas as etapas de um ambiente de *Machine Learning*, desde o tratamento dos dados, passando pela criação e escolha de algoritmos, treinamento e ajustes de modelos e gerenciamento e implantação (AMAZON, 2021d).

3. MATERIAIS E MÉTODOS

Neste capítulo, serão apresentados a descrição da empresa focal e os métodos usados no processo de produção deste trabalho.

3.1 A empresa Flowsense⁴

Figura 2 - Logo da Empresa Alvo



Fonte: Flowsense

A Flowsense, localizada na cidade de São Paulo, é uma empresa de engajamento mobile que auxilia outras empresas de diversos ramos a melhor se comunicarem com usuários de seus aplicativos (FLOWSENSE, 2020). Nasceu com um grupo de três alunos da Escola Politécnica da Universidade de São Paulo (EP-USP) dentro da incubadora da universidade, e, desde então, tornou-se um nome reconhecido no mercado de Marketing Digital.

Em aproximadamente seis anos, seu time cresceu de três engenheiros para mais de 20 pessoas de diferentes formações, e seu faturamento anual hoje é superior a R\$1MM. Em 2019, o banco de dados cresceu 10 vezes e a empresa conta hoje com mais de 75 milhões de usuários analisados e mais de 150 *apps*.

A Flowsense procura entender o perfil e a jornada dos usuários de modo a permitir a comunicação das empresas com eles de forma flexível e customizada. Com isso, é possível conhecer melhor a base de usuários, além de condicionar a comunicação com eles de acordo com quem eles são, sua interação com o aplicativo, o horário do recebimento e a localização geográfica do usuário. Assim, sua proposta de valor congrega dados, inteligência e otimização, e engajamento no app, como apresentado na Figura 3.

⁴ Os dados apresentados foram retirados do site da empresa e de relatórios internos recolhidos com os executivos da empresa.

Figura 3 - Proposta de Valor da Flowsense



Fonte: site da empresa em CTR/2021 - <https://flowsense.com.br/>

Para isso, as empresas clientes da Flowsense incluem em seus aplicativos um trecho de código que as permite enviar para o banco de dados da Flowsense informações como a localização do usuário e sua interação com o aplicativo, desde que o usuário aceite compartilhar esses dados. Tais informações são, então, processadas para entender diversos aspectos do usuário, como sua classe social, a frequência com que ele visita as lojas físicas do cliente, em que etapa da jornada do usuário ele está, entre outras análises.

Essas informações são disponibilizadas para as empresas por meio de um dashboard, no qual é possível criar campanhas de *Mobile Marketing* segmentadas por qualquer um desses dados coletados. O envio de Campanhas de Marketing via *dashboard* torna possível enviar notificações mais relevantes para cada usuário individualmente. Nele também é possível acompanhar dados de campanhas, como o número de usuários afetados e o número de usuários que clicou na mensagem, além de visualizar outras análises como o progresso pela jornada do usuário ou a performance de campanhas anteriores.

A Flowsense possui mais de 150 aplicativos integrados de diversas áreas de negócio, incluindo grandes bancos, empresas de varejo e programas de fidelidade. Dentre

seus clientes estão nomes como Bradesco, UOL, Americanas, VR Benefícios, Carrefour, Banco Next, Allya, Dotz, Ame, e ChefsClub.

Ao longo dos anos, a empresa focal foi reconhecida com diversas premiações, como o prêmio de campeã na categoria *Adtech* da Innovation Awards Latam, o 1º lugar na lista de empresas de *Big Data* da 100 Open Startups, TOP 5 Startups no Emerge Americas 2019 e 1º lugar no Judge Awards na NTT Data Open Innovation 2019. Mais recentemente, a Forbes Brasil incluiu, na lista dos Under 30 2020, os fundadores da Flowsense que foram destaque na categoria Marketing e Publicidade.

A Flowsense tem como seu objetivo auxiliar seus clientes a melhor se comunicarem com os usuários de seus aplicativos. Para tal, ela oferece vários serviços para que a empresa possa entregar a mensagem certa para o consumidor certo, possibilitando entender melhor quem são os seus clientes e enviar mensagens personalizadas para cada grupo e situação. Pensando na definição de Marketing apresentada por Kotler (1991), pode-se dizer que a Flowsense atua para facilitar a troca entre os aplicativos e os usuários, logo a Flowsense realiza Marketing.

Levando em conta essa conclusão, torna-se interessante considerar a metodologia de pesquisa de marketing, como apresentada por Kotler e Armstrong (1999), no contexto da Flowsense. Assim como em uma pesquisa de marketing e seguindo também o conceito do *Design Thinking* descrito por Brown (2008) de criar produtos a partir das necessidades do usuário, esse trabalho iniciou-se pela definição do problema que se buscava solucionar.

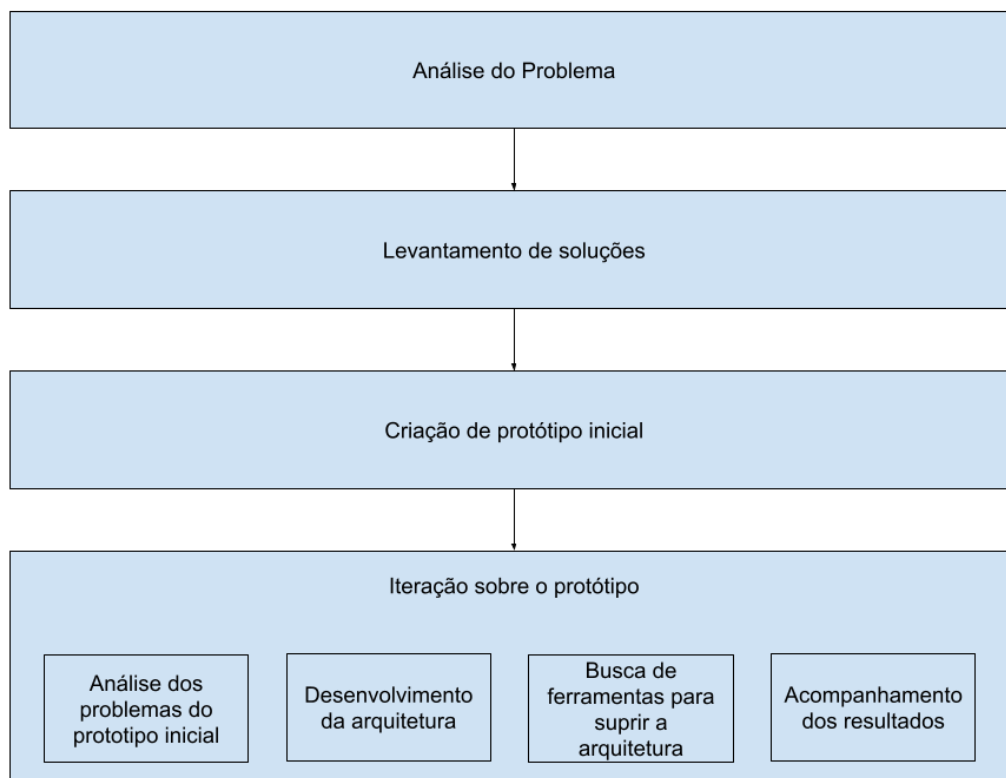
3.2 Método

O trabalho se organizou a partir de um conjunto variado de ações definidas pela equipe da empresa e sustentadas pelo aporte teórico apresentado anteriormente. Assim, foram realizadas as seguintes ações como parte do processo de análise e de desenvolvimento do produto:

1. Análise do problema;
2. Levantamento de soluções;
3. Criação de protótipo inicial;

4. Iteração sobre o protótipo;
 - 4.1. Análise dos problemas do protótipo inicial;
 - 4.2. Desenvolvimento da arquitetura;
 - 4.3. Busca de ferramentas para suprir a arquitetura;
 - 4.4. Acompanhamento dos resultados.

Figura 4 - Ações para análise e desenvolvimento do produto



Fonte: Autor

3.2.1 Análise do problema

Para iniciar o processo, foram realizadas a identificação do problema e sua descrição em um paralelo com proposta de *Design Thinking* sobre imersão. Nessa atividade, foi possível realizar um aprofundamento no tema e identificar várias perspectivas e situações particulares que compõem o problema em foco.

3.2.2 Levantamento de soluções

Em seguida, possíveis soluções para o problema definido foram levantadas. Diferentes formas para resolver o problema foram elencadas e uma foi decidida com base em parâmetros do projeto.

3.2.3 Criação de protótipo inicial

Em resposta à compreensão do problema, foi desenvolvido um produto inicial como um protótipo. A base do produto foi *Machine Learning*. A partir de então, algumas ações foram empreendidas: definição de parâmetros a serem utilizados para otimização do produto, criação de um modelo de acordo com o algoritmo escolhido, construção de previsões, realização de treinamentos, armazenamento dos resultados, processamento dos dados, avaliação dos clientes e levantamento para melhoria do produto.

3.2.4 Iteração sobre o protótipo

Para criar robustez para o produto, foram realizados alguns processos tais como: levantamento dos principais problemas do protótipo inicial, criação de possíveis soluções para cada um dos problemas e implementação das soluções.

- **Análise dos problemas do protótipo inicial**

O início da iteração foi a análise dos problemas do protótipo inicial. A partir do entendimento do que poderia ser melhorado foram levantadas possíveis soluções para cada um dos problemas.

- **Desenvolvimento de arquitetura**

O próximo passo foi agregar as soluções para as limitações do protótipo em uma arquitetura unificada que compreendesse todas ou pelo menos as principais.

- **Busca de ferramentas para suprir a arquitetura**

A partir da arquitetura definida, foi então necessário construir o produto. Para tal foi feita uma busca das melhores ferramentas e serviços para suprir as necessidades apontadas.

- **Acompanhamento dos resultados**

Com o produto montado foi necessário manter um acompanhamento de seus resultados para entender o sucesso do produto. Para tal foi utilizado um teste AB.

O teste AB é uma metodologia de pesquisa de experiência dos usuários, aplicando um teste de hipótese estatística. Organiza-se como um experimento aleatório com duas variantes de uma única variável A e B. Neste trabalho, foi realizada distinção de *pushes* pertencentes ao grupo de controle e ao de tratamento.

3.3 Materiais

Para a execução do projeto foram necessários, além dos métodos descritos, uma série de materiais.

3.3.1 Dados internos da Flowsense

O ponto central do projeto foi o aproveitamento dos dados que a Flowsense coleta dos usuários. Esses dados possuem uma enorme diversidade de informações, como dados sobre a localização física dos usuários, suas ações dentro dos aplicativos e suas respostas a *pushes* enviados pela Flowsense.

3.3.2 Ferramentas da AWS

Um dos materiais mais utilizados ao longo da execução do produto foram as ferramentas da AWS. Diversas ferramentas foram utilizadas, desde computação na nuvem, passando por armazenamento e até ferramentas de *Machine Learning*.

3.3.3 Algoritmos de *Machine Learning*

Outro material que foi utilizado foram algoritmos de *Machine Learning*. Esses algoritmos foram o que permitiram que os dados pudessem se transformar em uma informação aplicável para otimizar notificações.

3.3.4 Sentry

Além das ferramentas da AWS outra ferramenta que foi de grande importância foi o Sentry. Este serviço permitiu o monitoramento dos outros sistemas para que quaisquer problemas fossem evidenciados.

4. RESULTADOS

Nesta seção, os resultados serão discutidos.

4.1 Identificação e Definição do Problema

Ao longo dos anos de atividade da Flowsense, foi possível notar que muitas vezes as empresas não tinham certeza de qual a melhor maneira de se configurar uma mensagem. Qual o melhor horário? Qual a melhor mensagem? Vale apenas usar uma linguagem mais coloquial? Havia também a necessidade de adequar essas comunicações ao mercado que se busca envolver. Dentro da definição de mercado de Kotler (1991) os diferentes usuários se assemelham quanto ao interesse pelo serviço oferecido pelo aplicativo, mas as necessidades de comunicação podem variar drasticamente. Foi pensando em ajudar com essas questões que surgiu a resolução de montar uma nova solução.

O time de *Data Science*, como o próprio nome já indica, tem sua expertise em trabalhar com dados. Sempre foi importante para a Flowsense coletar e armazenar dados sobre as diferentes interações que o aplicativo mantém com o usuário, incluindo as mensagens enviadas e as ações que o cliente toma em relação a essas mensagens (clica, ignora, etc). Seria possível, então, partir dessa base de dados para tentar compreender quais as melhores condições para enviar uma notificação de forma que ela tenha sucesso. Mas, para isso, é importante antes decidir o que significa uma mensagem ter sucesso.

Há diversas formas de se considerar o sucesso de uma mensagem, desde métricas mais gerais sobre o sucesso do aplicativo como um todo até métricas mais ligadas diretamente às mensagens. Idealmente, a métrica adotada indica que o objetivo da mensagem foi atingido, representando uma ação tomada pelo usuário. Hoje, é possível cadastrar diversos gatilhos que são representativos de ações que o usuário realiza dentro do aplicativo por meio do uso dos chamados *In-App Events*, mas é necessário que o cliente inclua manualmente esse gatilho, o que, além de ser uma dificuldade em si, significa também que não há dados sobre esses eventos antes dessa inclusão, impossibilitando o uso de dados históricos. Esses eventos também são mais difíceis de serem associados diretamente à mensagem que foi enviada, já que a verdadeira causa do *In-App Event* não é conhecida. Por isso, a métrica que faria mais sentido é o CTR. Ela está diretamente

atrelada à mensagem, é muito utilizada pelo mercado (o que significa que o cliente verá mais valor em sua melhora) e está incluída nos dados históricos da Flowsense.

4.2 Produto Inicial

Após um bom aprofundamento no problema, estava então no momento de seguir para a próxima etapa tanto ao considerar o processo de criação de produto segundo a metodologia de *Design Thinking* descrita por Brown (2008), ou seja, a etapa de prototipação, quanto ao considerar a metodologia de Pesquisa de Marketing exposta por Kotler e Armstrong (1999), ou seja, a implementação.

Partindo da meta de escolher os melhores parâmetros para uma mensagem a fim de que ela seja clicada, o time de *Data Science* começou a buscar uma solução. Ao analisar o problema que buscava-se resolver, a possibilidade de empregar um algoritmo de *Machine Learning* foi imediatamente aparente. Esse tipo de algoritmo é altamente eficiente em encontrar os melhores parâmetros para otimizar um resultado esperado. Quanto maior a riqueza dos dados de treinamento, melhor o treinamento para que se consiga encontrar os melhores parâmetros.

Com a metodologia de como otimizar os parâmetros definida, foi necessário escolher quais parâmetros seriam otimizados. Muitos parâmetros podem ser ligados diretamente ao clique. Dentre os parâmetros possíveis, é possível mencionar, por exemplo, o tamanho da mensagem, a linguagem utilizada, o uso de emojis, imagens e botões, o conteúdo textual da mensagem. Ademais, existem parâmetros que dizem respeito à relação que o usuário manteve com o aplicativo. Dentre os exemplos, pode-se citar o histórico de cliques, outras ações que o usuário já tomou no aplicativo e quantas mensagens esse usuário já recebeu. Além disso, pode-se fazer uso das características próprias do *Mobile Marketing* como apontados por Shankar e Sridhar (2009): a especificidade de localização, a portabilidade e a conectividade sem fio. Apesar de não serem parâmetros da mensagem em si, poderiam ser considerados para otimizar uma mensagem por descreverem o contexto no qual a mensagem é recebida. A localização física do usuário, o horário local e local de residência do usuário são apenas alguns dos exemplos de valores que podem ser levantados por conta do Marketing ser *Mobile*.

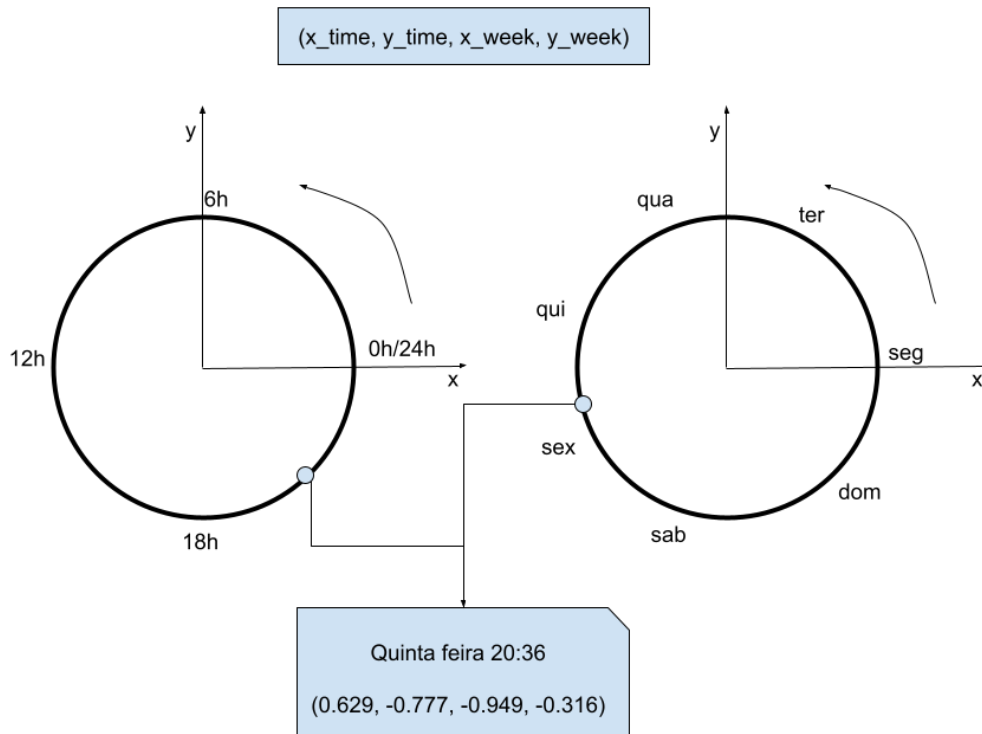
Para poder testar o uso de uma ferramenta de otimização, a escolha foi pautada em algumas condições. A primeira foi que o parâmetro deveria ser facilmente convertido em

um valor numérico, de forma que fosse mais fácil trabalhar com ele diretamente nos algoritmos, sem grandes esforços de pré-processamento. Outra condição foi que o parâmetro pudesse ser alterado sem necessidade de um grande envolvimento com o cliente (o cliente não precisa de um conhecimento muito avançado nem de esforço do time de desenvolvimento para poder usar a ferramenta) ou que mudasse radicalmente a mensagem. Isso permitiria a criação de um protótipo inicial que poderia ser mostrado ao cliente já entregando valor.

Com essas condições, foi possível selecionar como parâmetros para o protótipo o horário e o dia da semana do envio das notificações. De acordo com o conhecimento do time, já era claro que a taxa de clique era influenciada pelo horário. Os resultados do artigo publicado por Visuri *et al.* (2019) também corroboravam com a hipótese de que o horário do dia era um parâmetro importante para a taxa de clique. Era difícil, porém, dizer exatamente quando era o melhor horário, ou se isso variava muito de aplicativo para aplicativo. Alterar o horário de envio também não era um grande problema: bastava, na hora em que o cliente selecionaria o horário, ele, em vez disso, selecionaria que gostaria de utilizar o otimizador de horário. Depois disso bastaria que o cliente fornecesse limites mínimos e máximos para o dia e horário de envio e o serviço tomaria conta do resto.

Foi necessário, então, definir como representar numericamente esses parâmetros. A maneira mais simples seria usar um número no intervalo $[0,24[$ para o horário e outro no intervalo $[0,7[$ para o dia da semana, mas esses valores não representam bem a ciclicidade do parâmetro (os horários 23:59 e 00:01 estariam extremamente distantes nessa representação, por exemplo). Para poder então considerar esse artefato foi definido um processo de pré-processamento que transformaria as informações de horário e dia da semana em coordenadas em um círculo, semelhante ao diagrama expresso na Figura 6.

Figura 5 - Diagrama de Parametrização do Tempo



Fonte: Autor

Com os parâmetros definidos, era necessário, então, definir um algoritmo de modelagem. Esse algoritmo iria receber grandes quantidades de dados com os parâmetros escolhidos e os resultados (se o *push* foi ou não clicado), gerando um modelo que, a partir dos parâmetros, gerava uma pontuação para o quanto ele acreditava que esse *push* seria clicado. Após alguns testes com diversos modelos, tais como regressão linear, regressão logística, árvore de decisão, entre outros, foi definido por apresentar o melhor resultado: o algoritmo *Gradient Boosting*, mais precisamente, a versão otimizada utilizada pela biblioteca XGBoost.

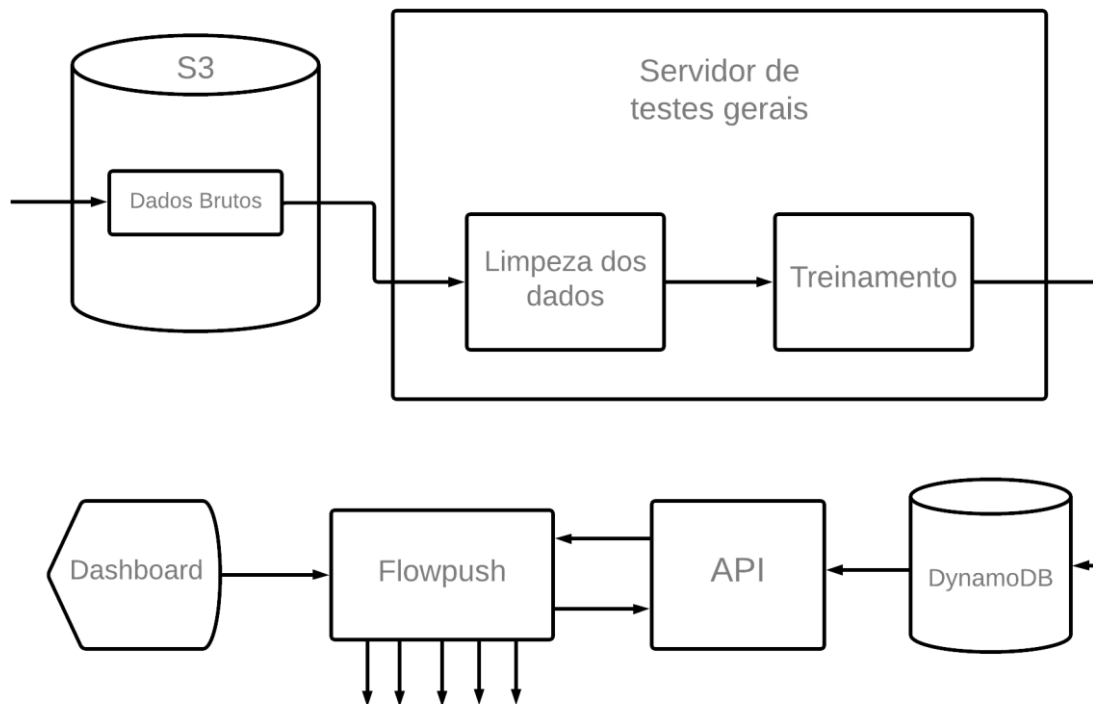
Com os parâmetros definidos e o modelo treinado de acordo com o algoritmo escolhido, foi possível fazer previsões para diferentes horários e dias da semana. Para isso, realizou-se um treinamento para os dados de cada cliente e os resultados foram salvos num banco de dados do DynamoDB (serviço de banco de dados da Amazon Web Services, AWS, já utilizado para outras necessidades na empresa), para que pudessem ser acessados com facilidade. Bastava, então, criar alguma interface que permitisse que

outros serviços da empresa pudessem requisitar qual o melhor horário para um *push* e devolvesse a opção ideal.

Para criar essa interface, foi utilizado o mesmo recurso que era utilizado para outras interfaces na empresa: um servidor rodando dentro de um container no Serviço de Contêineres Elásticos (ECS) da AWS que recebe chamadas de API via Flask (um framework para aplicativos web em Python). Esse servidor receberia, via Flask, as informações de qual o cliente e qual o horário e o dia máximos e mínimos para o envio do *push*. O servidor puxaria, então, o modelo treinado do DynamoDB e o rodaria em todos os horários possíveis dentro dos limites colocados, com intervalos de 5 minutos cada um. Em seguida, ele colocaria os horários em ordem decrescente de pontuação dada pelo modelo e retornaria os horários nessa ordem.

Com esse servidor rodando, o time de tecnologia da Flowsense pôde incluir no *dashboard* (principal ferramenta de comunicação das empresas com os serviços da Flowsense) uma nova aba que definiria o melhor horário para um *push*, bastando o cliente inserir o horário máximo e o mínimo. Essa aba fazia uma solicitação para o servidor, que respondia com o melhor horário, e, então, encaminhava o *push* para o serviço de envio de *pushes* (chamado de Flowpush) com o horário definido. Esse serviço de otimização do horário foi nomeado de Delivery Inteligente. A Figura 6 apresenta o funcionamento do Delivery Inteligente de forma simplificada.

Figura 6 - Diagrama de Desenvolvimento do Produto Inicial



Fonte: Autor

O diagrama expresso na Figura 6 pode ser entendido da seguinte maneira:

- Resultado dos *pushes* enviados é armazenado no S3
- Esses resultados são baixados de forma manual para o servidor de testes da Flowsense
- Nesse mesmo servidor são realizados os treinamentos
- Os modelos treinados são enviados para um banco do DynamoDB
- Quando um cliente utiliza o Delivery Inteligente, o pedido é enviado ao Flowpush, que solicita o melhor horário à API
- A API pega o modelo do DynamoDB e faz a avaliação do melhor horário de acordo e envia de volta para o Flowpush, que faz o envio dos *pushes*

O Delivery Inteligente foi bem recebido pelos clientes, que ficaram satisfeitos por não precisarem escolher um horário quando eles não sabiam qual o melhor horário para o

envio de *pushes*. Ainda assim, o produto tinha muitos problemas e seria necessário repensá-lo para que ele pudesse se manter ativo e expandir.

4.3 Produto Final

Este primeiro produto descrito na seção anterior serviu bem como uma prova de conceito, mas, para que seu uso pudesse ser implementado de forma robusta para se tornar parte do portfólio de produtos da Flowsense, seriam ainda necessárias grandes mudanças.

4.3.1 Problemas do protótipo inicial

Apesar de já auxiliar na resolução do problema definido, essa solução ainda carecia de aperfeiçoamento. Os problemas levantados estão descritos a seguir.

A. Os treinamentos foram feitos manualmente

Os modelos criados para predição de *push* precisavam ser atualizados com alguma regularidade. Isso se deu por dois principais motivos: as tendências de clique mudarem com o tempo e os modelos com poucos dados poderem ser melhorados com o fornecimento de novos dados recentes.

O Delivery Inteligente não possuía uma forma de atualizar esses modelos por conta própria, o que tornava necessário que algum integrante do time manualmente fizesse essa atualização, baseando-se somente em sua percepção de se já estava na hora de atualizar.

B. Os treinamentos foram feitos num servidor geral que era usado para testes e análises gerais

Apesar de a API do Delivery Inteligente ter rodado em um servidor próprio, o treinamento dos modelos ainda rodava no servidor utilizado para testes e análises gerais. Esse servidor tinha limites de memória e de computação, que podiam ser utilizados por outros programas que a Flowsense rodava, impedindo, assim, o sucesso do treinamento.

C. Não havia um acompanhamento do sucesso do treinamento de modelos

Mesmo que o treinamento dos modelos fosse executado de forma automática, haveria ainda o problema de que não havia nenhum acompanhamento quanto ao sucesso desses treinamentos. Se erros ocorressem, seria difícil verificá-los e resolvê-los.

Como o treinamento era feito manualmente, não havia uma necessidade de acompanhamento formal, pois os erros ocorriam na frente do operador, mas, no caso da automatização, esse acompanhamento se tornaria imprescindível.

D. Os dados brutos eram muito pesados e não foram armazenados de uma maneira que fosse de fácil ingestão

Os dados que são necessários para o treinamento do modelo eram armazenados sem grande preocupação sobre a facilidade para conseguir consumi-los no futuro. Isso significa que não apenas era necessário procurar por uma quantidade muito maior de arquivos, mas os arquivos em si eram muito grandes, com diversas informações que não eram relevantes para o Delivery Inteligente.

Para outros serviços da Flowsense, esses dados estavam disponíveis em um banco de dados do MongoDB, mas esse era um banco de dados quente, ou seja, apenas os dados mais recentes ficavam salvos nele, o que o tornava insuficiente para o êxito do treinamento.

E. Não havia um acompanhamento da qualidade dos modelos do Delivery Inteligente

Para poder entender se as variáveis utilizadas, hiperparâmetros definidos e algoritmos escolhidos estavam fornecendo um resultado esperado, seria necessário que algum tipo de acompanhamento de métricas de sucesso para o treinamento fosse feito, de forma a entender se algum desses parâmetros deveria ser alterado.

F. O Delivery Inteligente poderia ser demorado se o chamado tivesse que ser feito individualmente por *push*

Nesse protótipo inicial, o resultado do modelo era único para cada aplicativo. Então, para cada nova campanha, era realizada apenas uma solicitação para a API do Delivery Inteligente. No futuro, porém, havia planos para que esse resultado fosse único

por usuário, o que iria aumentar muito a quantidade de chamadas (de uma para milhares ou até milhões).

O tempo de resposta precisaria ser muito menor, portanto, para dar conta dessa demanda, mas puxar o modelo do DynamoDB não era tão rápido (se uma chamada durasse 100ms, por exemplo, uma campanha para 1M de usuários demoraria mais de um dia para receber todos os horários).

G. A escolha de novas variáveis ou novos algoritmos significaria refazer tudo do zero

Diversos algoritmos foram testados antes de ser feita a escolha de utilizar o XGBoost. Porém, mesmo com esse algoritmo definido, era reconhecido que, de acordo com novos parâmetros ou novas necessidades de modelo, seria possível que um novo algoritmo se tornasse mais desejável (inclusive com planos de poder, no futuro, testar diferentes algoritmos periodicamente para escolher o melhor). Desta forma, era importante que a estrutura fosse agnóstica ao algoritmo sendo utilizado, permitindo que sua troca fosse feita sem grandes dificuldades.

H. A dificuldade de ter um ponto de comparação para o sucesso dos *pushes* enviados com e sem o Delivery Inteligente

Mesmo que o treinamento dos modelos e as chamadas da API funcionassem perfeitamente, ainda seria necessário saber se o Delivery Inteligente realmente teve um impacto positivo no sucesso dos *pushes*.

Os dados sobre os *pushes* que eram salvos nos diferentes bancos de dados não permitiam a diferenciação entre um *push* enviado utilizando os parâmetros do Delivery Inteligente e um *push* enviado de forma comum, tornando impossível saber qual o impacto efetivo da ferramenta.

4.3.2 Soluções Propostas

Para poder criar um produto mais robusto e confiável, foram definidas soluções para cada um dos problemas levantados anteriormente. As soluções estão descritas abaixo e a implementação de cada uma está descrita na próxima seção.

A. Automatização do fluxo de treinamento

Para garantir que os modelos sempre estivessem em dia, foi criado um fluxo de automação que atualiza todos os modelos uma vez por semana. As versões anteriores continuavam salvas para análises futuras, mas a API utilizaria apenas a mais recente.

B. Criação de arquitetura e servidores próprios para o treinamento que fossem independentes do resto da estrutura da empresa, mas pudesse se comunicar com ela

Para ter mais controle dos recursos disponíveis para o Delivery Inteligente, foi criada uma arquitetura independente para todas as etapas necessárias. Cada parte da arquitetura pôde ter suas especificações de recursos de acordo com a necessidade.

C. Criação de log das métricas do treinamento e de alertas para a ocorrência de erros

Para ter mais controle do estado de cada uma das partes, que se mostrou ainda mais necessário quando o processo se tornou automatizado, foram criados dois sistemas: um para guardar métricas e informações que descreviam o treinamento dos modelos para verificação posterior e outro para alertar o time em caso de erros ou imprevistos.

D. Elaboração de uma pré-seleção dos dados, coletando apenas as informações necessárias e carregando-as em um *database* separado

Para não necessitar consultar o banco de dados de base, que leva muito tempo, as informações relevantes foram compiladas e salvas em outro banco de dados separado, o que permitiu uma escolha de índices mais relevantes e arquivos mais concisos. Isso possibilitou que as consultas fossem mais rápidas e consumissem menos memória. Além disso, como o custo dessas consultas era de acordo com o tamanho dos arquivos verificados, o gasto com a implantação de um novo banco de dados foi balanceado por consultas mais baratas.

E. Criação de log das principais métricas de sucesso e de gráficos que permitissem a fácil visualização da evolução dessas métricas

Como forma de acompanhar a qualidade dos modelos, métricas de sucesso foram armazenadas junto às métricas de treinamento. Ademais, gráficos de sua evolução foram

criados para que essas métricas pudessem ser verificadas com facilidade. Além disso, também foram coletados e compilados os dados do uso do modelo em produção, o que permitiu saber o sucesso real do modelo.

F. Confeção de um cache na API que fez com que a chamada fosse muito rápida caso o mesmo modelo fosse solicitado várias vezes seguidas

Como o principal gargalo da chamada da API era a solicitação do modelo no DynamoDB, um *cache* permitiu que chamadas consecutivas para um mesmo modelo (como no caso de vários usuários do mesmo cliente) fossem muito mais rápidas sem gastar muito espaço na memória.

G. Com uma arquitetura bem definida e bem documentada, além de uma modularização de necessidades, inclusão de novos parâmetros ou novos algoritmos com mais facilidade

Retomando o conceito de arquitetura modular proposto por Baxter (2000), pode-se dividir os diferentes elementos funcionais em módulos. Esses módulos possuem responsabilidade por completo dos elementos funcionais associados a eles, de forma que caso seja necessário trocar alguma delas por quaisquer motivos, precisaria-se apenas garantir que o novo módulo aceitasse os outputs e enviasse os inputs que os outros módulos esperavam. Adotando esse modelo de arquitetura para o Motor de *Pushes*, ficou muito mais fácil trocar ou até testar diferentes configurações. Para saber exatamente quais eram esses inputs e outputs para cada módulo, uma documentação bem detalhada se tornou essencial.

H. Uso da estratégia do teste AB, ou seja, seleção de uma parte dos *pushes* para não serem enviados de acordo com o Delivery Inteligente e comparação entre o sucesso dos que usaram e o dos que não usaram

Um dos métodos mais comuns (e por bom motivo) para testar se uma alteração faz diferença em um dado resultado é o teste AB. Ele permite que você compare o resultado de duas versões garantindo que a única diferença entre elas seja a alteração que se quer verificar. O uso do teste AB nesse caso permitiu entender o real impacto do uso do Delivery inteligente, quando comparado a *pushes* enviados da maneira anterior.

4.3.3 Arquitetura da Solução

A partir desses novos requisitos, então, foi criada uma nova arquitetura de software que desse conta de atender a todas as necessidades. A essa arquitetura foi atribuído o nome “Motor de *Pushes*”. O primeiro desenho foi feito de acordo com como se imaginaria que a arquitetura deveria funcionar, complementando e alterando a partir de como os dados já eram obtidos e trabalhados na primeira versão do Delivery Inteligente. É importante ressaltar que algumas das soluções descritas na seção anterior foram sendo desenvolvidas junto à criação da arquitetura, então não constavam no planejamento inicial.

Usando esse rascunho como base, foram sendo pesquisadas quais eram as melhores opções de software para cada uma das etapas, e adaptando o desenho de acordo com elas.

A primeira parte definida foi como os dados seriam armazenados. Essa não foi uma escolha difícil, já que, como uma empresa que trabalha com uma grande quantidade de dados, a Flowsense tem conhecimento sobre diferentes bancos de dados e serviços e, mais que isso, utiliza-os para outras partes de sua operação. Dessa forma, o armazenamento escolhido foi o S3 da AWS, que é utilizado para várias outras ferramentas e pode armazenar todo tipo de arquivo, sendo fácil e rápida tanto a inserção como a extração. Além disso, por seu uso no armazenamento para outras funções da empresa, a economia de escala faz com que o preço seja bem reduzido.

Como fazer *queries* no S3 para mais de um arquivo não é possível, já que ele não é um banco de dados e, sim, um serviço de armazenamento, para alguns pedidos mais complicados, como o de compilar os resultados, foi utilizado um *framework* que se associa ao S3 chamado Athena (junto a uma ferramenta que transforma os arquivos do S3 em tabelas, chamada Glue). O Athena permite que sejam feitas *queries* no formato de *queries* para bancos de *query* estruturada (*Structured Query Language* ou SQL).

A verdadeira dificuldade do armazenamento foi criar um detalhamento de quais arquivos seriam guardados e qual a organização deles dentro dessa estrutura.

Para definir como seriam feitos os treinamentos dos modelos, foi feito um estudo comparativo dos principais *frameworks* para *Machine Learning*. Os *frameworks*

comparados foram: MLFlow, Kubeflow, Paquiderm, AWS Sagemaker, Google Cloud AI Platform e Azure Machine Learning.

Primeiramente, fez-se a distinção entre os serviços gerenciados e os não-gerenciados. Os serviços gerenciados possuíam um custo de operação maior, mas eram mais fáceis de se utilizar. Os serviços não gerenciados possuíam uma flexibilidade maior e eram gratuitos.

Um primeiro teste foi feito com o MLFlow e foi bem positivo, com os treinamentos sendo bem organizados. Mesmo assim, a opção final foi pelo Sagemaker, por possuir como oferta a possibilidade de realizar os treinamentos em instâncias temporárias, o que teria que ser feito manualmente por outro serviço em um *framework* não-gerenciado. Houve, também, com a escolha desse *framework*, uma redução de trabalho e de custos.

O algoritmo usado continuou sendo o XGBoost, mas o Sagemaker permitiu a troca por outro algoritmo de maneira simples e rápida. Para o acompanhamento dos treinamentos foi escolhida como métrica de sucesso a Area Under Curve Receiver Operating Characteristic (AUC ROC). Essa métrica foi escolhida, pois a curva ROC permite analisar o *trade-off* entre a sensibilidade (quantos *pushes* que seriam clicados foram previstos como tal) contra a especificidade (quantos *pushes* que *não* seriam clicados foram previstos como tal). Ao procurar modelos que otimizassem a área sob a curva, o que se fez foi procurar um modelo que avaliasse bem quando se atingia o máximo de usuários que iriam clicar e perturbasse o mínimo possível de usuários que não teriam interesse no *push*.

Para o controle e a orquestração das métricas, assim como seu acompanhamento, foi definido que haveria uma máquina independente que seria responsável por isso. A opção por usar o serviço EC2 da AWS foi fácil, por ser um sistema que a equipe já tinha experiência e que comunicaria bem com o resto das estruturas da empresa, uma vez que boa parte dos produtos também estavam em serviços da AWS.

Nessa parte, a maior dificuldade foi escolher o modelo de instância que seria utilizado, visto que poderia ter um impacto significativo no custo. No fim, escolheu-se utilizar um dos modelos mais baratos, já que, a partir da escolha de um serviço gerenciado

para os treinamentos, as operações que exigiriam muita memória ou muito processamento foram externalizadas. Essa escolha também foi feita junto à de usar funções Lambda para a limpeza dos dados. Nesse tópico, cabe destacar que funções Lambda constituem um serviço da AWS e, portanto, integrável às demais soluções da empresa que permite executar pequenos trechos de código em um servidor não gerenciável de alta disponibilidade.

Na sequência foram realizadas algumas mudanças na API que já existia para que essa se adequasse ao novo contexto da arquitetura e às necessidades comentadas. A primeira foi a criação de um sistema de *cache* para eliminar a necessidade de puxar o modelo do DynamoDB para todo novo chamado.

A segunda mudança foi que, como o gargalo de coletar o modelo para cada usuário havia sido resolvido, a necessidade de ter um *database* dedicado a fornecer modelos à API se tornou obsoleta. A API passou, então, a puxar os modelos diretamente do S3. Isso também simplificou o processo, já que o Sagemaker não salvava diretamente para o DynamoDB e era necessária uma etapa a mais para essa transferência.

Por fim, era necessário que se criasse uma forma de acompanhar os dados. Essa necessidade era tanto para o time de Data Science que precisava acompanhar o sucesso e evolução dos modelos, quanto para mostrar para os clientes o impacto que o uso do Delivery Inteligente teria em seus *pushes*.

Para poder mostrar que o Delivery Inteligente auxiliava a melhorar o sucesso dos *pushes*, optou-se por utilizar um teste AB. Esse teste consistia em separar um grupo aleatório e representativo de todos os usuários, chamado de grupo de controle, enquanto o resto dos usuários seria o grupo de tratamento. Para os usuários do grupo de controle, a escolha do horário dos *pushes* não era feita pelo Motor de *Pushes*. Em vez disso, um horário dentro de um intervalo razoável (sem envios tarde da noite ou de madrugada) seria escolhido aleatoriamente.

Antes de fazer a comparação do grupo de controle com o de tratamento, porém, era necessário que se pudesse diferenciar os *pushes* enviados pelo Motor de *Pushes* dos enviados regularmente. Era necessário, portanto, que fosse feita uma distinção no momento em que os *pushes* eram armazenados. Foi feito, então, um pedido para o time

de *Back-End* para que se incluísse nos registros de *pushes* enviados se o *push* era para o grupo de tratamento ou de controle, além de qual o modelo que foi utilizado. Essa segunda informação era para poder diferenciar tanto os diferentes treinamentos do mesmo algoritmo quanto, no futuro, possibilitar a comparação de algoritmos distintos.

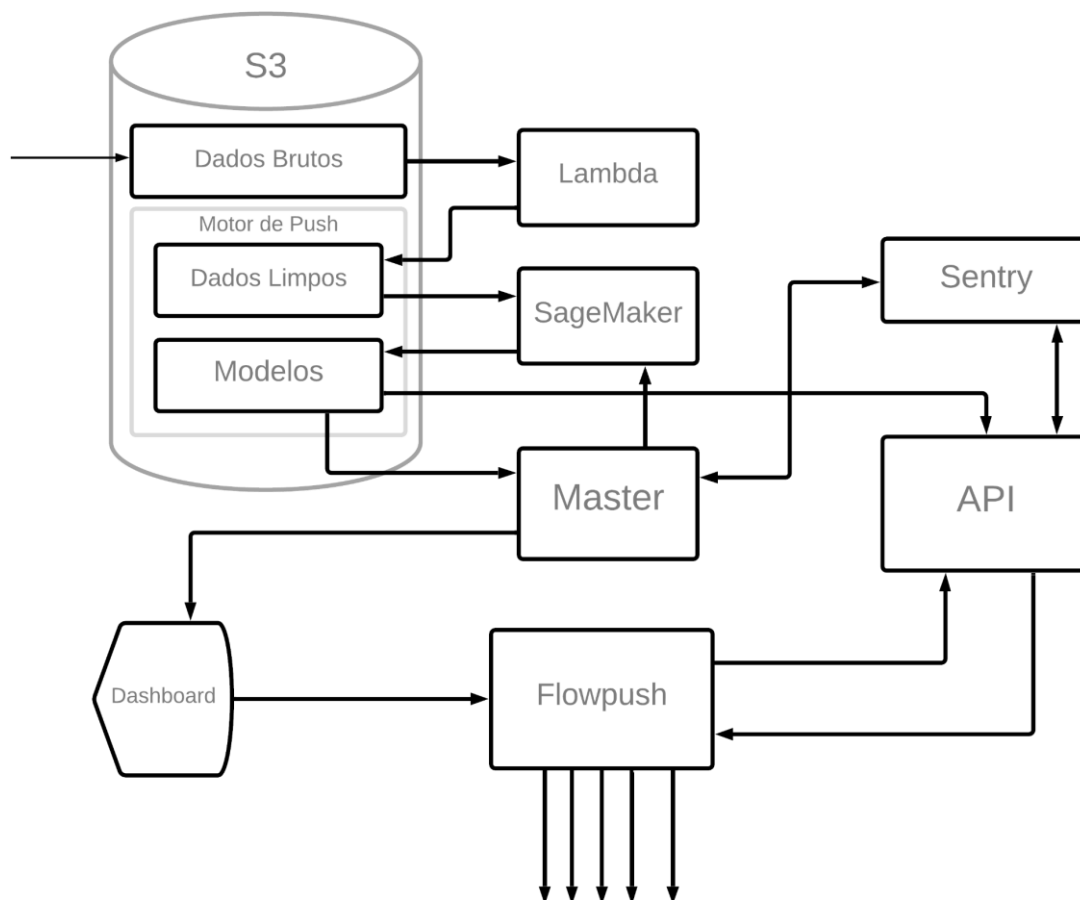
A tarefa seguinte era agregar os resultados para a análise. Para melhor entender a evolução dessas métricas, definiu-se que essa agregação seria feita por dia, coletando o total de envios e de cliques e diferenciando-os quanto pertencentes ao grupo de tratamento ou ao de controle.

O próximo passo foi conversar com o time de *Front-End* para que fosse feita a visualização desses dados. Ela estaria disponível para o cliente no mesmo painel que ele já utilizava para interagir com os outros serviços da Flowsense. Para que fosse criada, porém, era antes necessário que os dados pudessem ser consumidos. Para tal, criou-se uma rotina na máquina Master para regularmente fazer a agregação e o *upload* desses dados em um outro banco de dados usado em muitos outros serviços da Flowsense, o MongoDB.

Como o sistema passou a ser bem complexo, era importante que quaisquer erros que aparecessem fossem vistos com maior facilidade. Para tal, foi utilizado um serviço que já era adotado em outros projetos, o Sentry. Ele permite que erros sejam relatados e resolvidos de forma mais organizada, juntando-os em um único *dashboard*, com detalhamentos sobre o código que gerou o erro e com e-mails enviados aos membros da área de *Data Science* chamando atenção para a ocorrência dos erros.

Sendo assim, o diagrama apresentado na Figura 7 busca apresentar, de maneira resumida, os principais tópicos abordados ao longo do Desenvolvimento do Produto.

Figura 7 - Diagrama de Desenvolvimento do Produto Final



Fonte: Autor

O diagrama expresso na Figura 7 pode ser entendido da seguinte maneira:

- Resultado dos *pushes* enviados é armazenado no S3;
- Esses resultados são limpos por uma Lambda, que tem como gatilho a chegada dos *pushes*;
- Os dados limpos são armazenados em um outro *Bucket* do S3;
- A Máquina Master, periodicamente, faz uma solicitação ao SageMaker para que ele faça o treinamento de novos modelos com base nos dados limpos;
- O Sagemaker salva o modelo e as informações sobre o treinamento no S3;
- Quando um cliente faz uma solicitação via *Dashboard* para enviar um *push* usando o Delivery Inteligente, o *Dashboard* faz uma solicitação ao *Flowpush* para fazer o envio, que, por sua vez, faz um chamado para a API do Motor de *Push*;

- Quando recebe uma nova chamada, a API do Motor de *Push* consome o modelo mais recente do S3, faz a previsão da probabilidade de clique de acordo com cada intervalo de cinco minutos dentro dos limites colocados pelo cliente e retorna ao *Flowpush* o melhor horário, que é usado para o agendamento de 90% dos envios (os outros 10% sendo o grupo de controle, para os quais o envio é realizado em horários aleatórios dentro do que o cliente coloca como limite).

5. DISCUSSÃO

Para poder compreender as soluções e os resultados que elas possibilitaram, é importante que, além da coleta dos resultados, seja feita sobre eles uma análise crítica, buscando entender o que fez com que a solução levasse a esses resultados e o que esses resultados significaram para a solução do problema proposto.

Para isso, nesta seção, é apresentado, primeiramente, um resumo dos resultados em correlação com a fundamentação teórica. Em conexão com as tabelas, são apontados resultados atingidos pelo delivery inteligente para, na sequência, ser realizada uma avaliação sobre esse produto. A partir de então, são apontadas as limitações e as possibilidades futuras de encaminhamento.

5.1 Resumo dos resultados em relação à teoria estudada

As tabelas a seguir resumem os principais resultados e os relaciona aos conceitos abordados na fundamentação teórica, buscando criar compreensões sobre o processo desenvolvido e as considerações a serem apresentadas na sequência.

Tabela 1 - Tabela-resumo dos resultados em relação aos conceitos centrais de Marketing

MARKETING		
Definição do Problema		Dificuldade em enviar notificações com os melhores parâmetros
Objetivo da Pesquisa de Marketing		Compreender as melhores condições para enviar notificações com sucesso
Informações Necessárias		Influência de diferentes parâmetros no sucesso das notificações
Plano de Coleta	Método da pesquisa	Uma mistura de pesquisa por observação e experimental
	Critérios de Escolha	Dados já coletados, que podem ser flexíveis quanto a análise dos mesmos permitir
	Amostragem	População completa
	Instrumentos para coleta	Informações coletadas pelos aplicativos dos usuários usando o SDK da empresa

Fonte: Autor

Conforme observado na Tabela 1, a partir das ideias de Kotler e Armstrong (1999), com relação aos aspectos ligados de marketing, a definição do problema estava ligada à dificuldade que os clientes têm em enviar notificações com os melhores parâmetros. Isso levou à definição do objetivo da pesquisa de marketing como a compreensão das melhores condições para enviar notificações para que elas tivessem sucesso. Para alcançar este objetivo foram levantadas como informações necessárias a

influência de diferentes parâmetros no sucesso das notificações. A fim de coletar essas informações, foram utilizados dados históricos assim como dados do resultado das notificações enviadas. Esses dados foram coletados pelos aplicativos usando o SDK da Flowsense.

Tabela 2 - Tabela-resumo dos resultados em relação aos conceitos centrais de *Mobile Marketing*

MOBILE MARKETING		
Características Aproveitadas	Especificidade de localização	Não foi utilizado nessa primeira versão
	Portabilidade	Permite o acesso constante e ininterrupto de aparelhos
	Conectividade sem fio	Permite o envio de notificações independente da localização do usuário
Tipo de Comunicação Mobile	Gatilho de Comunicação	Push
	Grau de conhecimento do usuário	Baixo, dados existem, mas só estão sendo usados agregados
Recursos de Marketing Utilizados		Internos ao cliente quanto ao criativo, externo (feito pela Flowsense) quanto ao canal

Fonte: Autor

Em termos de *Mobile Marketing* (Tabela 2), foi possível verificar que, das características levantadas por Shankar e Sridhar (2009), a única que não foi aproveitada foi a especificidade da localização. A portabilidade e a conectividade sem fio, por outro lado, são o que permitem que se considere que o canal está sempre aberto. Quanto às vertentes explicadas por Kaplan (2012), o gatilho de comunicação utilizado foi o de *push* e o grau de conhecimento do usuário foi baixo. A partir dos dados já possuídos seria possível enviar mensagens que considerassem informações específicas de cada usuário, mas o trabalho de análise para tal comunicação ainda não foi feito e os dados foram agregados. Em relação aos apontamentos de Leppäniemi e Karjlouto (2008), pode-se dizer que os clientes utilizam recursos internos para a criação da parte da criação das mensagens e externos quando se considera que o canal de envio é fornecido pela Flowsense.

Tabela 3 - Tabela-resumo dos resultados em relação aos conceitos centrais de Produto

CARACTERÍSTICAS DO PRODUTO		
Machine Learning	Tipo de modelo	Modelo de predição: previsão do CTR a partir do horário
	Categoria de Algoritmo	Supervised learning: treinamento a partir de dados históricos
	Algoritmo	Gradient Boosting
Endpoint	Estrutura	Container do docker em uma máquina virtual da EC2
	Framework da API	Flask
Armazenamento		Dados e modelos salvos no S3

Fonte: Autor

Sobre o produto final em si pode-se começar a entender suas características a partir dos elementos de *Machine Learning* que ele utiliza (Tabela 3). O Delivery Inteligente cria um modelo de previsão do CTR a partir de algoritmos de *supervised learning* sobre dados históricos de horário de envio e clique de notificações. Outra característica importante é o *endpoint*. Ele tem como estrutura um contêiner de docker montado em uma máquina virtual no EC2 da AWS. O *framework* utilizado para a API foi o Flask. O armazenamento dos dados é feito no S3.

Tabela 4 - Tabela-resumo dos resultados em relação aos conceitos centrais de Desenvolvimento de Produto

DESENVOLVIMENTO DE PRODUTO		
Oportunidades		Empresas não sabem a melhor forma de se comunicar com seus usuários
Compreensão do Problema		Empresas não sabem quais os melhores parâmetros para que uma notificação seja mais efetiva
Estabelecimento de Metas		Tentativa intuitiva de resolução do problema a partir de possíveis aspectos passíveis de serem trabalhados para aumentar o CTR
Design Thinking	Inspiração	Histórico com clientes criando bases para compreensão do problema e de suas possíveis soluções
	Ideação	Busca por possíveis parâmetros a serem trabalhados: horário
	Implementação	Desenvolvimento do Delivery Inteligente

Fonte: Autor

No que concerne ao desenvolvimento do produto (Tabela 4), a partir do apontado por Baxter (2000), foi notado, a partir de anos de experiência com os clientes, uma oportunidade no fato de que as empresas não sabem a melhor forma de se comunicar com seus usuários. A partir daí, buscou-se otimizar parâmetros que influenciassem no sucesso,

que foi medido pelo CTR. Apesar de não ter sido rigorosamente seguida, as etapas de design thinking serviram de inspiração para o trabalho.

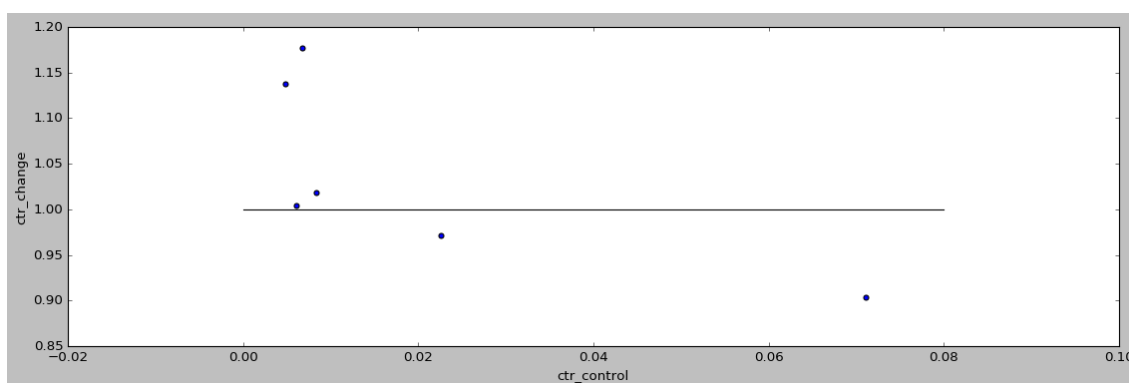
5.2 Resultados Atingidos pelo Delivery Inteligente

O Delivery Inteligente foi para produção em julho de 2020. Desde então, considerando dados coletados no dia 26 de junho de 2021, seis clientes diferentes já fizeram o uso da ferramenta para enviar um total de 1.383.791 *pushes* dos quais 31.064 foram clicados. Isso representa um CTR de 2,24%. Concomitantemente, os grupos de controle de cada campanha receberam um total de 76.332 *pushes*, dos quais 1.842 foram clicados, o que representa um CTR de 2,41%. Isso significa que, como um todo, o efeito do Delivery Inteligente no CTR foi uma redução de 6,97%. Porém esses números não contam toda a história.

O impacto no CTR de cada cliente foi muito diferente. Em clientes que possuíam o CTR baixo (abaixo de 1%), o Delivery Inteligente foi benéfico, aumentando o CTR entre 0,4% e 17%. Já para clientes que possuíam CTR mais significativo (entre 2% e 7%), a ferramenta teve um efeito negativo. No pior dos casos, o CTR do grupo de tratamento foi 9,7% pior que o do grupo de controle.

Nota-se pelo gráfico da Figura 9 que o efeito no CTR é menor quanto maior o CTR original.

Figura 8 - Gráfico de Variação do CTR de acordo com CTR original



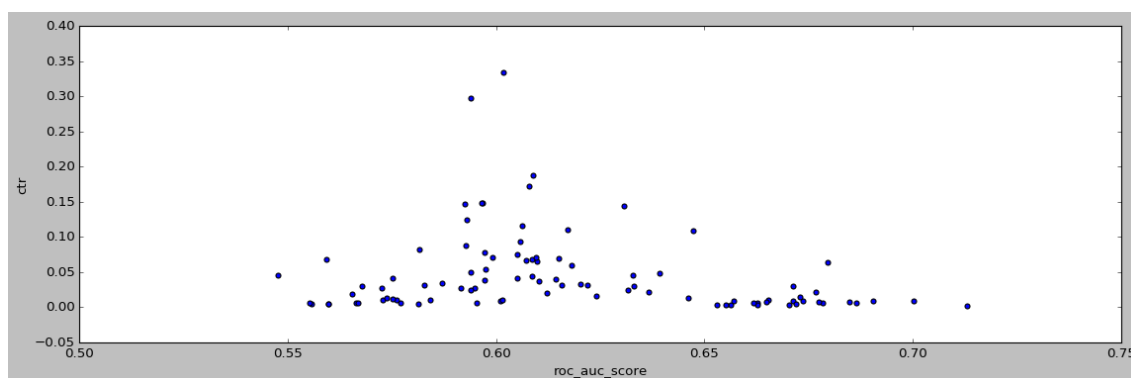
Fonte: Autor

O gráfico da Figura 9 também evidencia que para a maior parte dos clientes o efeito do Delivery Inteligente foi positivo. Isso pode parecer paradoxal quando o efeito

total foi negativo, mas o que aconteceu foi que o cliente que possuía o maior número de usuários foi também o que teve o pior efeito.

Outro resultado interessante é a comparação da métrica de sucesso de treinamento (AUC - ROC) com a variação do CTR obtida para cada modelo treinado. Pode-se notar, ao examinar a Figura 10, que essas duas medidas não estão muito correlacionadas. O valor do coeficiente de correlação de Pearson $r = -0.20$ evidencia que, apesar de a correlação dessas duas métricas ser negativa, ela é fraca.

Figura 9 - Comparação do CTR com métrica AUC ROC



Fonte: Autor

5.3 Avaliação Crítica

No momento de desenvolvimento da pesquisa, o Motor de *Push* teve pouco sucesso. Ao analisar os diferentes clientes que o usaram, como previamente exposto, pôde-se notar que clientes que normalmente tinham *pushes* que performavam pior encontraram relativo benefício no uso do Delivery Inteligente. Já para clientes mais maduros, o efeito do Motor de *Push* não foi significativo. Uma possível hipótese para esse resultado é que as possibilidades de ação do Motor de *Push* ainda eram muito limitadas, não permitindo o uso do real potencial do *Mobile Marketing* apontado por Shankar e Sridhar (2009).

Ademais, a fraca correlação entre a variação do CTR e a métrica de sucesso de treinamento adotada indica que a AUC ROC pode não ser a melhor métrica para identificar se um modelo terá a performance desejada.

Além disso, apesar de já ter sido possível deduzir alguns resultados, o uso do Delivery Inteligente ainda não foi tão significativo. Algumas análises mais gerais foram

possíveis dado que o número total de *pushes* foi grande, mas esses *pushes* foram todos de um número pequeno de campanhas, o que dificulta análises mais complexas como a de séries temporais, além de deixar os resultados menos confiáveis.

Apesar disso, é importante levar em consideração que o foco deste trabalho mais do que criar um produto que obtivesse um grande sucesso em aumentar o CTR foi criar a estrutura de um sistema que permitisse fazer modificações e melhorias com facilidade. A criação dessa estrutura segue em linha com a ideia de ciclos de inovação descritos por Brown (2008) e a criação de um produto de forma iterativa como aponta Liedtka (2018).

As possibilidades de ação e diferentes parâmetros levantados ao longo dos estudos de Mehrotra *et al.* (2015 e 2016 a e b), Visuri *et al.* (2019) e Pielot e Park (2018) se tornam possibilidades de serem implementadas com muito mais facilidade, uma vez que dependem apenas da coleta e transformação em parâmetros de um modelo, sem a necessidade de montar novamente todo a estrutura de treinamento, servicionamento do modelo, manutenção dos sistemas e gerenciamento de dados.

5.4 Limitações da Solução

Para poder ter uma visão mais crítica e entender como melhorar a solução, é importante discutir as limitações que ela apresentou. Um dos focos deste projeto foi criar uma estrutura que permitisse o crescimento e a expansão da solução, mas, para isso, é importante ver onde o projeto estava e o que deveria ser mudado.

O principal problema foi que, no momento do trabalho, o modelo não considerava informações muito descritivas de um indivíduo. As variáveis apenas descreviam o aplicativo, o horário e o sistema operacional. Apenas com essas variáveis não foi possível fazer um bom discernimento dos diferentes comportamentos e preferências dos indivíduos. Isso indicou que ainda não era possível afirmar que as mensagens eram customizadas a nível de usuário, algo avaliado como uma das características mais importantes de uma notificação.

Além de os *inputs* do modelo terem sido limitados, os *outputs* também foram. Ao final do trabalho, a única mudança que o modelo pôde fazer foi alterar o horário em que

os *pushes* eram enviados. Ele não teve nenhum efeito sobre o conteúdo ou mesmo sobre a decisão de enviar ou não uma notificação.

Ademais, houve, ainda, o problema de todos os horários estarem baseados no horário de Greenwich (também conhecido como *Universal Time Coordinated* ou UTC) e não no horário local do dispositivo. Apesar de grande parte dos usuários estar no mesmo fuso horário (ou em fusos próximos), seria importante ter levado esse fator em consideração, especialmente considerando a crescente diversidade de usuários conforme a Flowsense se expande. Esse problema não poderia ser resolvido de maneira simples, já que seria necessário que o aplicativo solicitasse por uma série de permissões de acesso ao dispositivo, o que, por não auxiliarem diretamente na função principal do *app*, poderia passar uma imagem ruim ao usuário.

5.5 Continuidade do Projeto

Com essas limitações em mente, para dar continuidade ao projeto, seria importante manter o monitoramento dos resultados para saber quais mudanças no modelo seriam, de fato, interessantes, assim como considerar outras variáveis que podem afetar o clique ou não em uma notificação. Considerar variáveis que levam em consideração o conteúdo da mensagem ou o histórico de cliques do usuário, por exemplo, seria interessante, visto que não exigiriam mudanças na parte do cliente e permitiriam uma personalização muito maior.

Ao final do trabalho, percebeu-se que o avanço mais importante do modelo seria na linha de conseguir que o Delivery Inteligente funcionasse de maneira individualizada. Dentro de um mesmo aplicativo, diferentes usuários têm comportamentos bem distintos, então considerar apenas os dados agregados acaba por desconsiderar essas variações. Essa conclusão também segue em linha com os resultados apresentados por estudos anteriores como o de Mehrotra *et al.* (2015). Criar a possibilidade para que a análise seja feita individualmente, porém, traria a uma série de novas complicações, como a transformação contínua (ou periódica) das interações dos usuários em métricas que pudessem ser utilizadas para avaliar quais valores atribuir às variáveis de *output* (por enquanto, o horário de envio).

Além disso, outra melhoria que seria interessante é conseguir considerar outras métricas de sucesso no Delivery Inteligente. Como demonstrado por Visuri *et al.* (2019),

o clique ou dismiss não necessariamente indicam o sucesso de um *push*, com muitos *pushes* que não são clicados sendo considerados importantes. Com o crescimento, dentro da Flowsense, do uso de *in-app events*, que incluem registros de diferentes gatilhos que os clientes cadastram, como passagens por telas e cliques em botões específicos, é possível que ações concretas do usuário dentro do aplicativo, que tragam resultados direto para o cliente (como compras, renovação de inscrições, etc), possam ser consideradas como a métrica de sucesso para o modelo. Esse trabalho já está sendo feito em outras áreas da empresa para analisar o sucesso de outros produtos oferecidos e boa parte dele poderia ser aproveitado para a continuidade do trabalho aqui proposta.

6. CONCLUSÕES

Ao final deste trabalho, é importante revisitar os conceitos e os objetivos discutidos no seu decorrer. Este trabalho teve como objetivo analisar a criação de um mecanismo para adequar a comunicação de empresas para cada usuário de forma que o aplicativo conseguisse manter o engajamento do usuário por meio de mensagens consideradas mais relevantes. Revisitando o conceito de Marketing descrito por Kotler (1991), pôde-se avaliar que o produto aqui desenvolvido facilitou a troca entre o aplicativo e o usuário, como era o foco do estudo, para a maior parte dos aplicativos.

A otimização do horário de envio contribuiu de forma a permitir que o usuário se comunicasse com o aplicativo no momento em que mais lhe fosse conveniente. Além disso, garantir que o *push* chegasse nesse horário melhorou as chances que, dentre outros *pushes* recebidos, esse tivesse interação.

Ainda assim, no momento final deste trabalho, todos os usuários que fossem do grupo alvo da campanha recebiam a mesma mensagem. A comunicação ainda pode ser melhorada considerando qual a mensagem e qual a frequência de comunicação, considerando tanto o contexto quanto o conteúdo como recomenda Mehrotra *et al.* (2015).

É possível também dizer que o trabalho alcançou em parte seu objetivo de criar um mecanismo que adequasse a comunicação das empresas para cada usuário para desenvolver maior possibilidade de engajamento, mas, mais do que isso, criou fundamentos robustos para que soluções mais completas e complexas possam ser criadas. Apesar de o motor de push já estar sendo utilizado, sua arquitetura modular permite que cada módulo passe por experiências iterativas junto ao usuário, de forma análoga ao processo de prototipação descrito por Liedtka (2018).

No entanto, apesar de o produto deste trabalho ter apresentado algumas limitações, seus ganhos práticos, já mencionados, são inegáveis. Além deles, com esse projeto, muitos aprendizados também foram vivenciados pelo autor, tais como a implantação de um sistema completo de aprendizado de máquina, o uso de diversas ferramentas (em especial da AWS) e modos de codificar atributos do usuário em métricas mensuráveis.

O futuro da Flowsense já conta com inovações a partir de 2021. Em 17 de junho, a empresa foi integrada a uma rede de empresas do Grupo DigitalReef que compõem um conglomerado de empresas de *Mobile Marketing* inovadoras. Sua proposta de prover serviços potentes para a América Latina implica impulsionar o *marketplace* global. Essa característica coloca demandas sobre o tipo de estudo aqui realizado e pode significar expansão e novas potencialidades para o futuro deste desenvolvimento e da própria empresa.

REFERÊNCIAS BIBLIOGRÁFICAS

ABLAS, B. O que é marketing digital? Entenda e veja dicas para aplicar na empresa. In: **TechTudo**, jul. 2020. Disponível em:

<<https://www.techtudo.com.br/noticias/2020/07/o-que-e-marketing-digital-entenda-e-veja-dicas-para-aplicar-na-empresa.ghml>>. Acesso em: 26 de junho de 2021.

AMAZON, AWS. **Amazon EC2**. Disponível em: <<https://aws.amazon.com/pt/ec2/?ec2-whats-new.sort-by=item.additionalFields.postDateTime&ec2-whats-new.sort-order=desc>>. Acesso em: 23 de junho de 2021a.

AMAZON, AWS. **What is Amazon EC2?** Disponível em: <<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html>>. Acesso em: 23 de junho de 2021b.

AMAZON, AWS. **Amazon S3**. Disponível em: <<https://aws.amazon.com/pt/s3/>>. Acesso em 23 de junho de 2021c.

AMAZON, AWS. **Amazon SageMaker**. Disponível em: <<https://aws.amazon.com/pt/sagemaker/>>. Acesso em: 23 de junho de 2021d.

AMAZON, AWS. **Amazon Elastic Container Registry**. Disponível em: <<https://aws.amazon.com/pt/ecr/>>. Acesso em: 23 de junho de 2021e.

AMAZON, AWS. **AWS Lambda**. Disponível em: <<https://aws.amazon.com/pt/lambda/>>. Acesso em: 23 de junho de 2021f.

BAXTER, M. **Projeto do Produto**: guia prático para o design de novos produtos. Tradução Itiro Iida. 2. ed. rev. São Paulo: Blancher, 2000.

BARWISE, P.; STRONG, C. Permission-Based Mobile Advertising. **Journal of Interactive Marketing**, v. 16, n. 1, p. 14–24, 2002. Disponível em: <<https://doi.org/10.1002/dir.10000>>. Acesso em: 23 de junho de 2021.

BROWN, T. Design Thinking. **Harvard Business Review**, Boston, p. 85-92, jun. 2008.

CACHE. In: **Michaelis Online**. Disponível em: <<https://michaelis.uol.com.br/moderno-portugues/busca/portugues-brasileiro/cache/>>. Acesso em: 26 de junho de 2021.

CONFESSORE, N. Cambridge Analytica and Facebook: The Scandal and the Fallout So Far. **The New York Times**, abr. 2018. Disponível em: <<https://www.nytimes.com/2018/04/04/us/politics/cambridge-analytica-scandal-fallout.html>>. Acesso em: 25 de junho de 2021.

DIETTERICH, T. G. Machine-learning research: Four current directions. **AI magazine**, v. 18, n. 4, p. 97-136, dez. 1997.

DOCKER. **What is a container?** Disponível em: <<https://www.docker.com/resources/what-container>>. Acesso em: 26 de junho de 2021.

FLOWSENSE. **Flowsense**. Disponível em: <flowsense.com.br> Acesso em: 02 de Fevereiro de 2020.

GAO, K. Click Through Rates: Click Through Rates Numbers and Their Meaning. In: **Email Marketing eBook**. 2012. Disponível em: <<https://web.archive.org/web/20121218223529/http://emailmarketing.comml100.com/email-marketing-ebook/click-through-rates.aspx>>. Acesso em: 23 de junho de 2021.

JANETZKO, D. The role of APIs in data sampling from social media. In: SLOAN, L.; QUAN-HAASE, A. (Eds.). **The SAGE Handbook of Social Media Research Methods**. Reino Unido: SAGE Publications LTD, 2017.

JORDAN, M. I.; MITCHELL, T. M. Machine learning: Trends, perspectives, and prospects. **Science**, v. 349, n. 6245, p. 255-260, jul. 2015. Disponível em: <science.sciencemag.org>. Acesso em: 23 de junho de 2021.

KANAGAVALLI, G. Mobile Marketing in India. In: ONE DAY NATIONAL SEMINAR ON IMPACT OF TECHNOLOGY IN BUSINESS, 2019, Devakottai, India. **Proceedings...** Devakottai, Índia: Shalax Publications, 2019. p. 6-10.

KAPLAN, A. If you love something, let it go mobile: Mobile marketing and mobile social media 4x4. **Business Horizons**, v. 55, p.129-139, 2012.

KOTLER, P. **Administração de Marketing: Análise, Planejamento, Implementação e Controle**. 2. ed. São Paulo: Atlas, 1991.

KOTLER, P.; ARMSTRONG G. **Princípios de Marketing**. 7. ed. Rio de Janeiro: LTC Editora, 1999.

LACHMAN, F. **Building docker images without accessing/build API endpoint**. 2017. 61f. Dissertação (Graduação em Informática) - Faculty of Informatics, Masaryk University, Brno, República Tcheca.

LEANPLUM. **Best User Communications Brands Report**. Disponível em: <www.leanplum.com/trends/best-user-communications-brand>. Acesso em: 28 de março de 2020.

LEPPÄNIEMI, M.; KARJALUOTO, H. Mobile Marketing: From Marketing Strategy to Mobile Marketing Campaign Implementation. **International Journal of Mobile Marketing**, v. 3, n. 1, jun. 2008.

LIEDTKA, J. Why Design Thinking Works. **Harvard Business Review**, Boston, p.72-79, set./out. 2018.

LEVITT, T. Marketing Myopia. **Harvard Business Review**, Boston, v. 38, n. 4, p. 45-56, jul./ago. 1960.

MDN Web Docs. **HTTP caching**. Disponível em: <<https://developer.mozilla.org/en-US/docs/Web/HTTP/Caching>>. Acesso em: 26 de junho de 2021.

MEHROTRA *et al.*. PrefMiner: Mining User's Preferences for Intelligent Mobile Notification Management. In: 2016 ACM INTERNATIONAL JOINT CONFERENCE ON PERVASIVE AND UBIQUITOUS COMPUTING (UBICOMP '16): 2016, Heidelberg, Alemanha. **Proceeding...** Birmingham, Reino Unido, University of Birmingham Research, 2016a. p. 1223-1234.

MEHROTRA *et al.*. My Phone and Me: Understanding User's Receptivity to Mobile Notifications. In: 2016 CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2016b, San Jose, CA, EUA. **Proceedings...** ACM Digital Library, 2016b. p. 1021-1032.

MEHROTRA, A. *et al.*. Designing Content-driven Intelligent Notification Mechanisms for Mobile Applications. In: 2015 ACM INTERNATIONAL JOINT CONFERENCE ON

PERVASIVE AND UBIQUITOUS COMPUTING (UBICOMP '15), 2015, Osaka, Japão. **Proceedings...** Osaka, Japão: ACM Digital Library, 2015. p. 813–824.

MITCHELL, T. **Machine Learning**. Nova York, EUA: McGraw Hill, 1997.

NAISBITT, J. **Megatrends**. New York: Warner Books, 1982.

OUYANG, W.; ZHANG, X.; REN, S.; QI, C.; LIU, Z.; DU, Y. Representation Learning-Assisted Click-Through Rate Prediction. In: INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE (IJCAI-19), 28., 2019, Macau, China. **Proceedings...** International Joint Conferences on Artificial Intelligence Organization, 2019. p. 4561-4567.

PIELOT, M.; VRADI, A.; PARK, S. Dismissed! A Detailed Exploration of How Mobile Phone Users Handle Push Notifications. In: INTERNATIONAL CONFERENCE ON HUMAN-COMPUTER INTERACTION WITH MOBILE DEVICES AND SERVICES (MOBILEHCI '18), 20., 2018, Barcelona, Espanha. **Proceedings...** Nova York, EUA: Association for Computing Machinery, 2018. p. 1-11.

POUSTTCHI, K.; WIEDEMANN, D.G. Success Factors in Mobile Viral Marketing: A multi-case study approach. In: INTERNATIONAL CONFERENCE ON MOBILE BUSINESS (ICMB), 6., 2007, Toronto, Canadá. **Proceedings...** Munique, Alemanha, IEE Computer Society Press, 2007. p. 1-8.

REDDY, M. **API Design for C++**. Burlington, MA, EUA: Morgan Kaufmann Publishers, 2011. Disponível em: <https://www.google.com/books/edition/API_Design_for_C++/IY29LylT85wC>. Acesso em: 24 de junho de 2021.

SENTRY. **Sentry**. Disponível em: <<https://sentry.io/>>. Acesso em: 26 de junho de 2021.

SHANKAR,V.; BALASUBRAMANIAN, S. Mobile Marketing: A Synthesis and Prognosis. **Journal of Interactive Marketing**, v. 23, n. 2, p. 118–129, 2009.

SMARTBEAR. **API Endpoints** - What Are They? Why Do They Matter? Disponível em: <<https://smartbear.com/learn/performance-monitoring/api-endpoints/>>. Acesso em: 28 de junho de 2021.

STACKSHARE. **Flask**. Disponível em <<https://stackshare.io/flask>>. Acesso em 26 de junho de 2021.

THE PALLETS PROJECTS. **Flask**. Disponível em:
<<https://palletsprojects.com/p/flask/>>. Acesso em: 26 de junho de 2021a.

THE PALLETS PROJECTS. **Pallets**. Disponível em: <<https://palletsprojects.com/>>.
Acesso em: 26 de junho de 2021b.

VERBLOW, B.; KUMAR, S.; REITSMA, R.; JOSHI, H. **Q3 2020 Digital Marketing Tracker Global**. Forrester. Disponível em:
<<https://www.forrester.com/report/Q3+2020+Digital+Marketing+Tracker+Global/-/E-RES163955?objectid=RES163955>>. Acesso em: 10 de junho de 2021.

VISURI *et al.*. Understanding smartphone notifications' user interactions and content importance. **International Journal of Human-Computer Studies**, v. 128, p 72-85, 2019.

WIEDEMANN, D.G. Exploring the Concept of Mobile Viral Marketing through Case Study Research. In: CONFERENCE ON MOBILITY AND MOBILE INFORMATION SYSTEMS, 2., 2007, Aachen, Alemanha. **Proceedings...** Bonn, Alemanha, Lecture Notes in Informatics, 2007. p. 49–60.