



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CAMPUS DE CHAPECÓ
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

THIAGO HENRIQUE FERREIRA CORREA

**DESENVOLVIMENTO DE APLICAÇÕES WEB QUE POSSIBILITEM COLETAR
DADOS UTILIZANDO O TAGUEAMENTO**

**CHAPECÓ
2023**

THIAGO HENRIQUE FERREIRA CORREA

**DESENVOLVIMENTO DE APLICAÇÕES WEB QUE POSSIBILITEM COLETAR
DADOS UTILIZANDO O TAGUEAMENTO**

Trabalho de conclusão de curso apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.
Orientador: Prof. Dra. Raquel Pegoraro

CHAPECÓ
2023

Correa, Thiago Henrique Ferreira

DESENVOLVIMENTO DE APLICAÇÕES WEB QUE POSSI-
BILITEM COLETAR DADOS UTILIZANDO O TAGUEAMENTO /
Thiago Henrique Ferreira Correa. – 2023.

50 f.: il.

Orientador: Prof. Dra. Raquel Pegoraro.

Trabalho de conclusão de curso (graduação) – Universidade Federal
da Fronteira Sul, curso de Ciência da Computação, Chapecó, SC, 2023.

1. Tagueamento. 2. Google Analytics. 3. Front-end. 4. Web
Analytics. I. Pegoraro, Prof. Dra. Raquel, orientador. II. Universidade
Federal da Fronteira Sul. III. Título.

THIAGO HENRIQUE FERREIRA CORREA

**DESENVOLVIMENTO DE APLICAÇÕES WEB QUE POSSIBILITEM COLETAR
DADOS UTILIZANDO O TAGUEAMENTO**

Trabalho de conclusão de curso apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.

Orientador: Prof. Dra. Raquel Pegoraro

Este trabalho de conclusão de curso foi defendido e aprovado pela banca avaliadora em: .

BANCA AVALIADORA

Prof. Dra. Raquel Pegoraro – UFFS

Prof. Me. Marina Girolimetto – UFFS

Profa. Me. Andressa Sebben – UFFS

RESUMO

No contexto digital, atualmente empresas e corporações dependem cada vez mais de informações precisas e relevantes para tomada de decisões estratégicas. A *web analytics* surgiu com a necessidade de coletar dados provenientes do tráfego de usuários em websites e se tornou uma prática fundamental nesse cenário. Para áreas como comércio eletrônico, marketing e gestão de negócios digitais, esses dados tornaram-se essenciais possibilitando a criação de estratégias de marketing de campanhas e melhorias dentro de uma plataforma web. O tagueamento de websites se tornou uma ferramenta estratégica para obtenção desses dados. Esse projeto tem como objetivo geral a elaboração de uma descrição da implementação do tagueamento em aplicações web utilizando o framework Javascript React.

Palavras-chave: Tagueamento, Google Analytics, Front-end, Web analytics

ABSTRACT

In the digital context, companies and corporations currently depend more and more on accurate and relevant information to make strategic decisions. *web analytics* emerged with the need to collect data from user traffic on websites and has become a fundamental practice in this scenario. For areas such as e-commerce, marketing and digital business management, this data has become essential, enabling the creation of marketing strategies for campaigns and improvements within a web platform. Website tagging has become a strategic tool for obtaining this data. This project has as general objective the elaboration of a description of the implementation of tagging in web applications using the Javascript React framework.

Keywords: Tagging, Google Analytics, Front-end. Web Analytics.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplos de dados conforme origem	21
Figura 2 – Componente em React	23
Figura 3 – Gerenciador de tags do Google	25
Figura 4 – Fluxo demonstrativo do processo de tagueamento	25
Figura 5 – Fluxograma de como funciona o Google Analytics	29
Figura 6 – Fases e atividades da proposta	31
Figura 7 – Layout da aplicação exemplo - Página de vendas	34
Figura 8 – Layout da aplicação exemplo - Página de contato	34
Figura 9 – arquitetura da aplicação	36
Figura 10 – Tipagem de dados typescript	36
Figura 11 – Função para disparar um evento no React	37
Figura 12 – Fluxo inicial para conexão do Google analytics com a aplicação	38
Figura 13 – Chave de métricas que será conectada no código	39
Figura 14 – Chave de métricas fornecido pelo Google analytics na ferramenta	39
Figura 15 – Função de disparo de clique com a tag que será enviada ao Google analytics	40
Figura 16 – Arquivos de rotas da aplicação	40
Figura 17 – Arquivos de rotas da aplicação	41
Figura 18 – Função de disparo de clique com a tag que será enviada ao Google analytics	41
Figura 19 – Formulário com componentes de input e botão	42
Figura 20 – Variável de estado que recebe texto de pesquisa	42
Figura 21 – Funções de input e envio de formulário	43
Figura 22 – Textos digitados no campo de pesquisa	44
Figura 23 – Gráfico com total de usuários e tempo médio de engajamento	45

LISTA DE TABELAS

Tabela 1 – Eventos coletados pelo Google analytics	43
--	----

SUMÁRIO

1	INTRODUÇÃO	15
1.1	PROBLEMÁTICA E JUSTIFICATIVA	16
1.2	APRESENTAÇÃO	16
2	OBJETIVOS	19
2.1	OBJETIVO GERAL	19
2.2	OBJETIVOS ESPECÍFICOS	19
2.3	DELIMITAÇÃO DA PESQUISA	19
3	REVISÃO BIBLIOGRÁFICA	21
3.1	BIG DATA	21
3.1.1	Origem dos dados	21
3.1.2	Estrutura dos dados	22
3.2	DESENVOLVIMENTO FRONT-END	23
3.2.1	REACT.JS	23
3.2.2	CONCEITOS BÁSICOS DE REACT E COMPONENTIZAÇÃO	23
3.2.3	BENEFÍCIOS DO REACT PARA O TAGUEAMENTO	24
3.3	WEB ANALYTICS E TAGUEAMENTO	24
3.3.1	TIPOS DE TAGUEAMENTO	26
3.3.2	BENEFÍCIOS E OBJETIVOS DO TAGUEAMENTO PARA ANÁLISE DE COMPORTAMENTO E MONITORAMENTO DE MÉTRICAS.	26
3.3.3	CONSIDERAÇÕES ÉTICAS E DE PRIVACIDADE RELACIONADAS AO TAGUEAMENTO	27
3.4	GOOGLE ANALYTICS	27
3.4.1	VISÃO GERAL DO GOOGLE ANALYTICS COMO UMA FERRA- MENTA DE ANÁLISE DE DADOS	28
3.4.2	COMO FUNCIONA	28
4	DESENVOLVIMENTO	31
4.1	FASES E ATIVIDADES PARA O DESENVOLVIMENTO DO PLANO DE ATIVIDADES DE TAGUEAMENTO	31
4.1.1	Etapa 1 - Planejamento da aplicação	32
4.1.2	Etapa 2 - Desenvolvimento do front-end e tagueamento do site	32
4.1.3	Etapa 3 - Configuração da ferramenta de coleta das ações de tagueamento	32
4.1.4	Etapa 4 - Análise dos dados coletados	32
4.2	IMPLEMENTAÇÃO DA PROPOSTA UTILIZANDO REACT	33
4.2.1	Planejamento da aplicação	33
4.2.2	Desenvolvimento do Front-end e tagueamento do site	35
4.2.3	Iniciando a aplicação com create-react-app	35
4.2.4	Configuração da ferramenta de coleta das ações de tagueamento	37

4.2.5	Tag de rastreo de cliques	39
4.2.6	Tag de visualizações da página	41
4.2.7	Tag de input de pesquisa	42
4.3	ANÁLISE DOS DADOS COLETADOS	43
5	CONCLUSÃO	47
	REFERÊNCIAS	49

1 INTRODUÇÃO

Com o surgimento da internet e o aumento do volume de dados conhecido como big data, empresas da área de tecnologia passaram a armazenar grandes volumes de dados, momentos após o qual diversas ferramentas surgiram para coletar essas informações permitindo que grande variedade e tipos de dados fossem apresentados, de forma não estruturada, ou seja não necessariamente de formato único ou origem única. A web, por exemplo, permitiu que essas informações fossem cruzadas, trazendo métodos estatísticos e algoritmos para que fosse descoberta um novo jeito de padronizar os dados advindo do comportamento dos usuários na internet (15).

Com esse grande volume de dados sem estruturação, big techs notaram a importância dessas informações para elaborar planejamento de negócios e, a geração de valor em torno dos dados dos seus usuários, trazendo consigo otimizações em seus sites e plataformas, possibilitando o planejamento de negócio e receita monetária a partir do planejamento dos dados de seus clientes (18). Atualmente temos diversas áreas interdisciplinares que são movidas a dados oriundo de informações capturadas pela interação dos usuários com suas plataformas na internet. As redes sociais foram um grande ponto de partida para definir a captura de dados como algo relevante. Outras áreas como grandes e-commerces, pequenos negócios online ou departamento de data science e marketing enxergam a importância de boas estratégias no contexto web associado ao sucesso empresarial (11).

O uso de ferramentas para capturar esses dados se tornou essencial, pois, torna-se possível analisar grandes volumes de informações de usuários, além prático a inserção de tags nas linhas de código inseridas por programadores durante o desenvolvimento da aplicação, o que também traz segurança no lado do servidor para que dados sensíveis não vazem. Empresas como o Google implementaram o Google Tag Manager (GTM) e o Google Analytics tornando o tagging, que trata da prática de marcação de trechos de códigos em componentes de Front-end para capturar informações de uso da plataforma pelos visitantes, mais acessível de forma gratuita, porém com limitações (11).

A linguagem Javascript associada a bibliotecas de front-end como o React.js também possibilitou que metadados fossem mapeados através de ações dos usuários em uma aplicação, onde é possível extrair informações através de simples interações como cliques, *rollouts*, tempo de tela e digitação, o que consolidou ainda mais a área de web analytics. Hoje os dados se tornaram de suma importância para planejamento de aplicações, realização de modelos de negócios, alcance de público e realização de vendas, podendo assim abrir diversas possibilidades para estudos e trabalhos em torno da área de web analytics (15).

1.1 PROBLEMÁTICA E JUSTIFICATIVA

O planejamento das informações devem ser construído com o objetivo de extrair o máximo de relatórios. Dados mal planejados geram incertezas na tomada de decisão, também chamado de fenômeno GIGO (*garbage-in, garbage-out*, ou lixo entra, lixo sai). Em aplicações web, esses problemas tornam-se ainda maiores, pois além dos dados de negócio também há a necessidade de coletar dados de comportamento dos usuários para obter *insights* e realizar modelos de predição.

Hoje aplicações web que não realizam a captura de informações comportamentais de usuário, pode enfrentar uma série de problemas em relação a UX (Experiência de Usuário), fluxos que não fazem sentido, componentes que não agregam à página e pouca intuitividade. Grandes corporações como Facebook, Twitter, Google, realizam tomada de decisões com base no comportamento do usuário em suas respectivas plataformas, obtendo sucesso em suas tomadas de decisões.

Planejar aplicações web é um processo complexo, dado que existem inúmeros frameworks, linguagens e paradigmas para estruturar essas plataformas, um dos princípios relacionados a desenvolvimento é "respeite os dados"(21), tais informações serão importantes para a construção das interfaces de usuário e serviços relacionados ao Back-end, ou seja a camada que receberá os dados é posteriormente deve armazená-los.

Em aplicações web, aplicar tagueamento otimiza ainda mais a forma de como o dado é capturado, trazendo mais flexibilidade e tornando a captura mais específica, considerando que os trechos de código serão escritos na menor parte dos componentes web, permitindo assim ter o maior controle da informação que será capturada.

Nesse contexto, este trabalho irá contribuir elaborando uma descrição da implementação de tagueamento em aplicações web, assim podendo ser replicado com o objetivo de obter métricas com os dados que forem coletados.

1.2 APRESENTAÇÃO

A motivação para este trabalho surgiu a partir da experiência profissional do autor atuando no desenvolvimento de um projeto para uma empresa brasileira que atua na área de Edtech, classificada como unicórnio por possuir um valor de mercado superior a \$ 1 bilhão de dólares. O projeto em questão se tratava de uma aplicação web com Front-end desenvolvido em React, e tinha como necessidade rastrear comportamentos dos usuários a partir da personalização de tags embutidas no código fonte. Na ocasião, por não haver referências consistentes, o projeto foi desafiador e gerou muitos aprendizados.

As aplicações que necessitam de analisar comportamentos de usuário através de tagueamento podem ser desenvolvidas com diversas tecnologias, como por exemplo o framework React para Front-end, sendo este amplamente utilizado pelo mercado.

Essas aplicações possuem em comum a necessidade de conhecer o comportamento dos usuários ao acessar a sua aplicação. Neste contexto, a pergunta de estudo foi definida da seguinte forma: “Como descrever um plano de atividades para realizar tagueamento em aplicações web no desenvolvimento Front-end com React permitindo gerar informações via Google Analytics?”

A decisão de utilizar os serviços do Google Analytics como solução de tagueamento foi baseada em diversos motivos. Primeiramente, o Google Analytics oferece uma ampla gama de recursos que permitem a coleta de dados detalhados sobre o comportamento dos usuários em uma aplicação web. Isso inclui a obtenção de informações como número de visitantes, páginas mais acessadas, taxas de conversão, entre outros indicadores relevantes para análises e tomada de decisões.

Além disso, o Google Analytics é amplamente reconhecido como uma das principais soluções de análise de dados do mercado. Sua popularidade se deve não apenas à sua capacidade de fornecer relatórios detalhados, mas também à sua interface amigável e à integração com outras ferramentas e serviços do Google, como o Google Ads (1).

A decisão de utilizar React foi motivada pela experiência do autor do trabalho com a biblioteca e por sua abordagem baseada em componentes. Com a biblioteca é possível criar componentes reutilizáveis, possibilitando taguear partes menores de cada componente. Outra vantagem da utilização do React para o tagueamento, foi sua grande aderência no mercado atual de desenvolvimento Front-end, onde a diversas bibliotecas e ferramentas complementares ajudam a tornar o tagueamento no *framework* mais prático.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Descrever um plano de atividades para realizar tagueamento em aplicações web no desenvolvimento Front-end com React permitindo gerar informações via Google Analytics.

2.2 OBJETIVOS ESPECÍFICOS

Para atingir o objetivo geral foram definidos os seguintes objetivos específicos:

- Definir um plano composto por fases e atividades para implementar tagueamento em aplicações web possibilitando capturar dados do comportamento específicos no Front-end da aplicação, e integra-las com o Google Analytics;
- Descrever passo a passo a implementação do plano utilizando o framework React como Front-end.
- Demonstrar os dados das tags implementadas e coletadas pelo Google Analytics.

2.3 DELIMITAÇÃO DA PESQUISA

Este trabalho delimita-se a tratar os elementos de Front-end desenvolvido em React com a finalidade de demonstrar as etapas de tagueamento e a integração com o Google Analytics.

3 REVISÃO BIBLIOGRÁFICA

3.1 BIG DATA

Big Data é a expressão que descreve conjuntos de dados grandes demais para serem analisados pelos sistemas tradicionais de processamento, e que, portanto, exigem novas maneiras de tratamento e tecnologias de armazenamento, processamento, análise, visualização e integração (4). Ciências como a astronomia e a genômica, que vivenciaram uma explosão informacional nos anos 2000, cunharam o termo big data relacionado aos grandes volumes de dados. O termo foi citado primeiramente no relatório *"Data, data, everywhere: a special report on managing information"*, do periódico britânico *The Economist*(3).

Com tecnologias cada vez mais miniaturizadas, podemos colocar inteligência nos limites mais externos das redes, permitindo que os processos de negócio sejam mais descentralizados, com decisões sendo tomadas localmente, melhorando o seu desempenho, escalabilidade e aumentando a rapidez das decisões. Resumindo, Big Data é igual a velocidade, variedade, veracidade e o valor agregado a desses dados (18).

3.1.1 Origem dos dados

São gerados petabytes de dados a cada dia. e estima-se que este volume dobre a cada dezoito meses. Aumentam também a variedade, sendo os dados estruturados a grande minoria e não estruturados a imensa maioria, gerados por e-mails, mídias sociais (Facebook, Twitter, YouTube e outros), documentos eletrônicos, mensagens instantâneas, sensores, etiquetas RFID, câmeras de vídeo (18).

Em 2013, a United Nations Economic Commission for Europe (UNECE), ou Comissão Econômica das Nações Unidas para a Europa, propôs uma taxonomia para classificar os dados com base em diferentes critérios, entre os quais a origem, como vemos no figura 1 (4).

Figura 1 – Exemplos de dados conforme origem

Origem	Exemplo
Dados gerados por seres humanos	Registros de experiências humanas, anteriormente gravados em livros e obras de arte e, posteriormente, em fotografias, áudio e vídeo, disponibilizados em redes sociais, blogs, documentos pessoais, aplicativos de mensagens.
Dados medidos e processados	Registros produzidos por sistemas de negócios de agências públicas ou privadas, como registros médicos, transações comerciais e financeiras, dados de e-commerce etc.
Dados gerados por máquinas	Dados gerados por sensores fixos (de clima, poluição, tráfego, segurança, automação doméstica etc.) ou de sensores móveis (geolocalização de aparelhos celulares, carros e imagens de satélites), além de registros de sistemas computacionais (como logs e Web logs)

Fonte: Adaptado de (15)

As redes sociais geram dados fornecidos por seres humanos, enquanto a Internet das Coisas é baseada em máquinas. Os dados mediados por processos de sistemas de negócios

registram, por exemplo, dados de transações, o que torna as organizações uma terceira fonte de dados (4).

3.1.2 Estrutura dos dados

Além do enorme volume de dados na web, eles também se caracterizam pelo uso de dados de diferentes naturezas, em cada um deles atribuem-se diferentes técnicas e ferramentas (4).

- Dados estruturados: dados de formato fixo e com tipos bem definidos (números, textos, datas etc.), organizados rigidamente na forma de tabelas (com linhas e colunas) em bancos de dados ou arquivos do Excel.

Todas as entidades do mesmo grupo (ou coluna, se pensarmos em tabelas) têm os mesmos atributos, formato e comprimento, além de seguirem a mesma ordem. Tudo é rotulado, fácil de acessar e pode ser facilmente processado por máquinas.

Os dados estruturados representam apenas 20 por cento dos dados gerados nos principais domínios de conhecimento ou prática (4).

- Dados semiestruturados: dados que podem ter certa estrutura, embora nem todas as informações coletadas tenham estrutura idêntica, resultantes do cruzamento entre as formas não estruturadas e estruturadas.

Marcadores ou descritores (os chamados metadados) são usados para identificar determinados elementos dos dados, mas os dados em si não têm uma estrutura rígida. É o que acontece com as páginas Web, os *feeds* do Twitter e as tags (palavras-chave) nos vídeos.

- Dados não estruturados: dados armazenados em um formato que facilita a leitura por seres humanos, mas que são dificilmente compreendidos por um computador, dadas suas irregularidades e ambiguidades. Como não seguem um formato ou padrão particular, geralmente não são armazenados em um banco de dados no formato tradicional de linhas e colunas.

Normalmente são textos produzidos em forma de documentos Word, mensagens de e-mail ou serviços de mensagens instantâneas, mas também de arquivos de logs, datas, números e eventos, bem como de arquivos de imagem, vídeo e áudio.

Os dados não estruturados representam a maior parte das informações disponíveis hoje, com tendência a crescimento exponencial. Esses dados não possuem nenhum modelo subjacente nem uma estrutura interna totalmente ordenada. Existem ferramentas maduras para tratar os dados estruturados, mas as ferramentas para mineração e análise de dados não estruturados são incipientes e ainda estão em desenvolvimento (4).

3.2 DESENVOLVIMENTO FRONT-END

Podemos definir como a parte visual de um site, aquilo que conseguimos interagir. Quem trabalha com Front-end é responsável por desenvolver por meio de código uma interface gráfica, normalmente com as tecnologias base da Web (HTML, CSS e JavaScript).

3.2.1 REACT.JS

React JS é uma biblioteca JavaScript para a criação de interfaces de usuário — ou UI (user interface). Criado em 2011 pelo time do Facebook, o React surgiu com o objetivo de otimizar a atualização e a sincronização de atividades simultâneas no *feed* de notícias da rede social, entre eles chat, status, listagem de contatos e outros (16).

3.2.2 CONCEITOS BÁSICOS DE REACT E COMPONENTIZAÇÃO

Para compreender a forma de implementação é necessário compreender alguns conceitos básicos apresentados a seguir:

- Componentes: os componentes permitem a divisão da interface do usuário em partes independentes e reutilizáveis e pense em cada parte isoladamente. Conceitualmente, componentes são como funções JavaScript. Eles aceitam entradas arbitrárias (chamadas “props”) e retornam elementos React que descrevem o que deve aparecer na tela (16).

A figura 2 mostra um exemplo de componente React que retorna um texto junto com a propriedade de nome originada do Javascript.

Figura 2 – Componente em React

```
function Welcome(props) {  
  return <h1>Olá, {props.name}</h1>;  
}
```

Fonte:(16)

- Estado: componentes comumente precisam mudar o que está na tela como resultado de uma interação. Por exemplo, ao se digitar em um formulário, deve-se atualizar o campo de entrada; clicar em “próximo” em um carrossel de imagens deve mudar qual imagem é exibida; clicar em “comprar” deve colocar um produto no carrinho, etc. Componentes precisam “lembrar” de coisas: o valor de entrada atual, a imagem atual, o carrinho de compras. Em React, esse tipo de memória do componente é chamado de estado (16).
- Eventos: são ações específicas que ocorrem dentro de um componente, como um clique num botão, a submissão de um formulário ou a digitação em um campo *input* de texto.

Esses eventos são disparados pelo usuário e são manipulados pelas funções de *handlers* (manipuladores) (16).

- Hooks: são funções que permitem “ligar-se” aos recursos de estado e ciclo de vida do React a partir de componentes funcionais. Hooks não funcionam dentro de classes — eles permitem que seja utilizado o React sem classes (16), ou seja, Hooks são funções prontas feitas pelos desenvolvedores do framework para facilitar o uso da ferramenta em trabalhos específicos como o carregamento de conteúdo dinâmico.

3.2.3 BENEFÍCIOS DO REACT PARA O TAGUEAMENTO

O React oferece uma série de benefícios ao realizarmos o tagueamento, pois existem bibliotecas específicas dentro do framework voltadas à coleta de dados e implementação de tags por meio do Google Analytics, tornando prática a aplicação de tags em nosso código. A biblioteca React-ga4 fornece diversos eventos e tags que podem ser adicionadas nas rotas da aplicação ou em funções de cliques e inputs para esses dados serem coletados pelo Google Analytics (14).

Por ser um framework Javascript voltado a componentização e o ciclo de vida dos componentes no Front-end, isolando os componentes e os tornando reutilizáveis, é possível aplicar as tags em apenas um componente e replicar por diversas páginas, apenas alterando as propriedades condizentes com a rota e seus parâmetros, reduzindo consideravelmente o retrabalho de recriar as tags em cada rota (16).

3.3 WEB ANALYTICS E TAGUEAMENTO

A web analytics é uma abordagem que envolve coletar, medir, monitorar, analisar e relatar dados de uso da Web para entender as experiências dos visitantes. A análise pode ajudar a otimizar sites da web para atingir metas de negócios e/ou melhorar a satisfação e a fidelidade do cliente (9).

Existem dois métodos comuns usados por ferramentas de análise da web para coletar dados de tráfego da web. O primeiro envolve o uso de arquivos de log baseados no servidor, e o segundo requer marcação de página baseada no cliente conhecido como tagueamento. A web analytics começou com a análise de dados de tráfego da web coletados por servidores da web e mantidos em arquivos de log (1).

A abordagem de marcação de página envolve adicionar algumas linhas de script às páginas de um site para coletar estatísticas delas. Os dados são coletados quando as páginas são carregadas no navegador do visitante à medida que as tags de página são executadas (1).

Tagueamento é a capacidade de uma plataforma coletar dados quantitativos e qualitativos de interação do usuário, como por exemplo, o clique em um botão, o preenchimento de uma *newsletter*, a confirmação de compra de um produto, sem no entanto, infringir o direito de

Figura 3 – Gerenciador de tags do Google

```

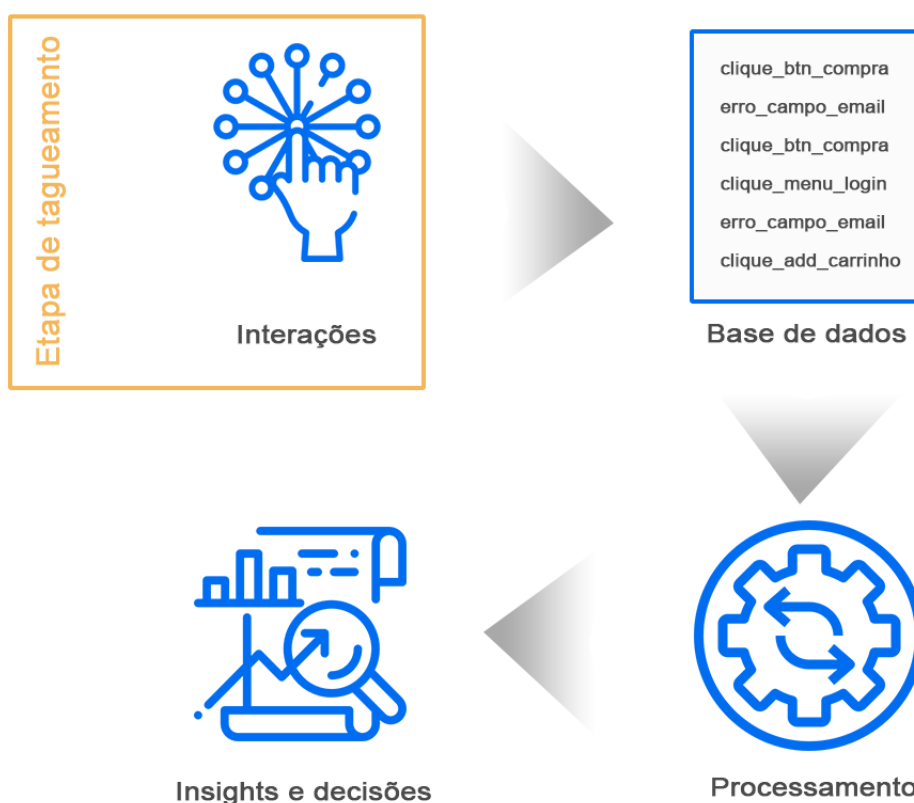
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-XXXX');
<!-- End Google Tag Manager -->

```

Fonte: (7)

anonimização deste usuário, ou seja, coletar informações sem identificar quem ele é. Uma tag é basicamente um trecho de código destinado a observar alguma interação do usuário com algum elemento específico (10).

Figura 4 – Fluxo demonstrativo do processo de tagueamento



Fonte: (10)

Para conduzir o processo de tagueamento, o desenvolvedor deve ser capaz de planejar e se antecipar a questões técnicas importantes que irão garantir um processo assertivo para trazer a capacidade de gerar uma nova dimensão de informações relevantes para serem analisadas (10).

3.3.1 TIPOS DE TAGUEAMENTO

Conforme (1), existem alguns tipos de tagueamento que podem ser implantados em uma aplicação web. Dependendo dos objetivos específicos do projeto e dos resultados esperados pode-se implementar um ou vários tipos de tags.

- Tagueamento de eventos: o tagueamento de eventos é utilizado para rastrear ações específicas dentro de uma aplicação web. As ações mapeadas com esse tipo de tagueamento são: cliques, *rollouts*, preenchimento de formulários, reprodução de vídeos e visualização de imagens (1).
- Tagueamento de conversão: as tags de conversão do Google Ads ajudam a criar relatórios que mostram o que acontece depois que um cliente clica nos seus anúncios e realiza uma ação, como comprar um produto, inscrever-se na *newsletter*, ligar para a empresa ou fazer o download do seu app. Quando o cliente conclui uma ação que você definiu como valiosa, dizemos que ele realizou uma conversão (6).
- Tagueamento de personalização: podem ser criadas tags personalizadas para que outras pessoas na organização possam usá-las. A natureza desses modelos, que se baseia em permissões e *sandbox*, possibilita criar TAGs e variáveis personalizadas de maneira mais eficiente e segura do que ao usar tags HTML e variáveis JavaScript personalizadas (6).

3.3.2 BENEFÍCIOS E OBJETIVOS DO TAGUEAMENTO PARA ANÁLISE DE COMPORTAMENTO E MONITORAMENTO DE MÉTRICAS.

O tagueamento para efetuar a análise de comportamento e monitorar métricas objetiva compreender e otimizar o desempenho de uma aplicação web ou aplicativo móvel. Os principais benefícios do tagueamento são:

- Coleta de dados: a conexão da ferramenta de tagueamento com as aplicações permite o envio de logs que possibilitem analisar, processar e segmentar dados reais do comportamento do usuário dentro de uma aplicação em específico (1).
- Análise de comportamento: a partir da coleta e da mensuração das métricas e com um conjunto de dados bem formados torna-se possível a análise comportamental do usuário na plataforma, rastreando essas ações através de eventos mais simples como cliques e outras interações (9).
- Tomada de decisões embasadas em dados: quando quantificadas as métricas e obtidos os resultados sobre o comportamento do usuário na plataforma, atingindo os objetivos propostos para o tagueamento, torna-se possível aplicar medidas em gestão e desenvolvimento e software para elaborar melhorias na plataforma utilizando os dados coletados. Isso possibilita a tomada de decisões mais assertivas, e embasadas em dados reais (1).

O tagueamento, quando utilizado de forma correta para análise de comportamento e monitoramento de métricas, traz uma base sólida de *insights* para entender como o usuário interage com o conteúdo de uma plataforma, fornecendo subsídios para tomada de decisões (1).

3.3.3 CONSIDERAÇÕES ÉTICAS E DE PRIVACIDADE RELACIONADAS AO TAGUEAMENTO

De acordo com (5), sites e aplicativos que utilizam o Google Analytics coletam e armazenam dados dos usuários, onde torna-se importante advertir aos visitantes o aceite e o armazenamento desses dados através de *cookies*. Cabe a cada país e suas autoridades a aplicação de leis que descrevem a forma como esses dados devem ser utilizados. Empresas que utilizam ferramentas próprias para realizarem o tagueamento também devem estar ciente das leis locais, e devem advertir ao usuário a captura e utilização dos dados de comportamento durante o uso da plataforma.

O ministério público brasileiro implementou diretrizes próprias para a Lei Geral de Proteção de Dados conhecida como LGPD. Trazendo diversas garantias aos usuários como por exemplo: poder solicitar que seus dados pessoais sejam excluídos; revogar o consentimento de acesso ao dados; transferir os dados para outros fornecedor de serviços, entre outras ações (12).

O Google (5) em sua documentação adverte ao desenvolvedor a gerenciar o consentimento do usuário com as seguintes regras:

- Saber se os visitantes autorizam ou não o armazenamento de informações sobre o comportamento realizado por eles;
- Informar ao Google qual foi a escolha do consentimento do usuário;
- Garantir que as tags do Google que armazenam ou leem os *cookies* obedeçam à escolha do usuário.

3.4 GOOGLE ANALYTICS

A principal função do Google Analytics é coletar dados de acesso, comportamento e navegação em sites e aplicativos por meio de códigos de rastreamento e organizar essas informações em relatórios diversificados (2). O google Analytics surgiu em 2005, e desde então se tornou referência no mercado, pela facilidade de integração com multiplataformas e por ter recursos poderosos, que trazem diversos benefícios para quem o utiliza (8).

3.4.1 VISÃO GERAL DO GOOGLE ANALYTICS COMO UMA FERRAMENTA DE ANÁLISE DE DADOS

Podemos listar algumas características que tornam o Google Analytics uma ferramenta poderosa para área de web analytics:

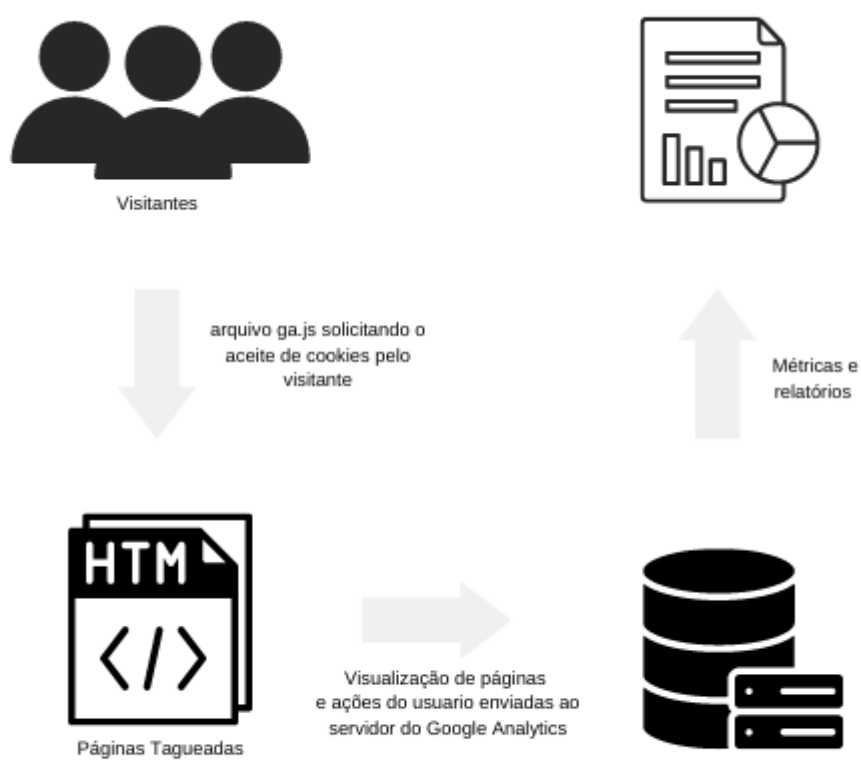
- Rastreamento dos dados: o Google Analytics coleta dados sobre os visitantes de um site, que incluem diversas informações como geolocalização, origem do tráfego, comportamento e interação do usuário com a aplicação, utilizando tags colocadas na plataforma de origem (5);
- Intuitivo: o Google Analytics oferece um painel de controle intuitivo, onde é possível visualizar os principais dados e métricas de um site de forma personalizável (2);
- Relatórios: o Google Analytics fornece uma ampla variedade de relatórios, gráficos e descrições pré-configuradas, os quais são gerados de forma automática pela ferramenta e ajudam a encontrar possíveis melhorias para um site ou aplicativo (8).
- Integração com outras ferramentas: o Google Analytics se integra com outras ferramentas Google, como o Google Ads para direcionamento de publicidade e criação de campanhas. Isso permite que a ferramenta possa combinar dados de diversas origens.

3.4.2 COMO FUNCIONA

De acordo com (1), é feito um pedido automático do arquivo em <http://www.google-analytics.com/ga.js>. Este é o arquivo mestre do Google Analytics, um arquivo JavaScript de 18 Kb que é baixado apenas uma vez durante a sessão do visitante. Outras solicitações serão recuperadas do cache do navegador do visitante.

Com o arquivo ga.js instalado, as informações do referenciador e outros dados do visitante (por exemplo, URL da página, carimbo de data/hora, ID exclusivo, resolução de tela, profundidade de cor) são coletados e um conjunto de cookies primários é criado para identificar o visitante — ou atualizado se o visitante for um visitante recorrente. Para cada exibição de página, o GATC envia essas informações aos servidores de coleta de dados do Google. Todo esse fluxo leva segundos e podem ser detectados em tempo real pelo dashboard do Google Analytics (1).

Figura 5 – Fluxograma de como funciona o Google Analytics



traduzida e adaptada de (1)

4 DESENVOLVIMENTO

O objetivo deste trabalho é descrever um plano de atividades para realizar tagueamento em aplicações web no desenvolvimento Front-end com React permitindo gerar informações via Google Analytics. Para atender ao objetivo, foi estruturado um plano composto por fases e atividades que permitem ao desenvolvedor implementar tagueamento a partir do front-end, capturar dados de componentes e integrar com o Google Analytics. Também foi implementado o exemplo de aplicação do plano desenvolvido em React. Nesta lógica de desenvolvimento do trabalho, este capítulo está estruturado da seguintes forma:

- Seção 4.1. Fases e atividades para o desenvolvimento do plano de tagueamento;
- Seção 4.2. Exemplo da implementação da proposta utilizando React;
- Seção 4.3. Análise dos resultados;

4.1 FASES E ATIVIDADES PARA O DESENVOLVIMENTO DO PLANO DE ATIVIDADES DE TAGUEAMENTO

O processo de implementação do plano de atividades foi dividido em etapas que possibilitem agregar valor ao resultado final seguindo um escopo estruturado. A estrutura do desenvolvimento está subdividida nas seguintes etapas, apresentadas na figura 6:

- Etapa 1 - Planejamento da aplicação
- Etapa 2 - Desenvolvimento do Front-end
- Etapa 3 - Configuração da ferramenta de coleta das ações de tagueamento
- Etapa 4 - Análise dos dados coletados

Figura 6 – Fases e atividades da proposta



Fonte: Desenvolvida pelo Autor

4.1.1 Etapa 1 - Planejamento da aplicação

Nesta etapa é necessário identificar os objetivos do negócio e as metas que serão alcançadas através da implementação de um plano de tagueamento. Também definir as categorias e tags que serão usadas para classificar os dados do tagueamento, e as métricas serão coletadas, como taxa de conversão, tempo médio de navegação, entre outros.

Também é necessário definir os dados de entrada e de saída do usuário, como por exemplo: inputs de pesquisa, cadastro de endereço de e-mail, páginas mais acessadas, geolocalização, preenchimento de formulários, avaliações, etc. após esses passos devemos mapear ações do usuário a serem coletadas: cliques, eventos, tempo de acesso, tempo de duração de sessões e a quantificação de visitas.

A partir dessas definições pode-se seguir o planejamento e seleção das tags que serão utilizadas no desenvolvimento do Front-end, conforme os dados e comportamentos que foram definidos anteriormente. O Google analytics disponibiliza diversas tags, eventos e *cookies* para adicionar ao site.

4.1.2 Etapa 2 - Desenvolvimento do front-end e tagueamento do site

Esta etapa visa garantir que a implementação do código seja feita de acordo com as especificações do plano de tagueamento. Isso inclui a implementação de tags em botões, formulários e outras áreas do site que irão coletar informações sobre as interações do usuário. Nesta etapa deve-se testar a implementação para garantir que as informações estão sendo coletadas corretamente.

4.1.3 Etapa 3 - Configuração da ferramenta de coleta das ações de tagueamento

Nesta etapa é definida qual a ferramenta de coleta de dados será utilizada para obter as informações sobre as interações do usuário. A configuração da ferramenta inclui a definição de variáveis de tagueamento que serão usadas para coletar informações específicas sobre as ações do usuário, como o clique em um botão ou o preenchimento de um formulário. Existem várias ferramentas no mercado. Neste trabalho será demonstrado o uso do Google Analytics e Google Tag Manager. Após esta fase estar concluída a aplicação web poderá ser colocada em produção, ou seja, disponibilizado para acesso ao público.

4.1.4 Etapa 4 - Análise dos dados coletados

Nesta etapa é realizada a análise dos dados coletados para entender o comportamento dos usuários e identificar oportunidades de melhoria no site. Isso inclui a análise das métricas definidas na etapa de planejamento da aplicação, como por exemplo taxa de conversão, tempo médio de navegação e entre outras. Com os dados coletados é possível criar relatórios

para acompanhar o desempenho do site e fazer ajustes no plano de tagueamento sempre que necessário.

Na seção 4.2 é apresentado um exemplo de implementação da proposta utilizando o framework React.

4.2 IMPLEMENTAÇÃO DA PROPOSTA UTILIZANDO REACT

React JS é uma biblioteca JavaScript para a criação de interfaces de usuário — ou UI (user interface). Criado em 2011 pelo time do Facebook, o React surgiu com o objetivo de otimizar a atualização e a sincronização de atividades simultâneas no *feed* de notícias da rede social, entre eles chat, status, listagem de contatos e outros (16). Quando se trata de implementar um plano de tagueamento, o uso de componentes reutilizáveis pode ser muito útil, pois permite a implementação de várias tags com objetivos diferentes em todo o site.

Hoje temos algumas bibliotecas para React voltadas a integrar o Google Analytics de forma facilitada em projeto web, utilizando a biblioteca NPM. O NPM é um poderoso gerenciador de pacotes utilizado para administrar as bibliotecas e frameworks utilizados em uma aplicação (17). O pacote NPM React-ga traz funções que podem ser facilmente integradas aos componentes de Front-end para realizar o tagueamento (14).

4.2.1 Planejamento da aplicação

Para demonstrar o passo a passo é apresentado um exemplo de uma página de venda de um e-commerce por ser amplamente utilizados pela área de marketing para aplicar tais técnicas de implementação (9).

Como mostrado anteriormente, React é uma biblioteca de Front-end que traz flexibilidade para personalização e possibilita adicionar tags e criar componentes voltados a eventos e coleta de dados de forma eficiente e rápida. Assim foi planejado criar uma página com finalidade similar de uma loja virtual para desenvolver botões de "adicionar carrinho" e páginas de "entre em contato" para realizar a mensuração e análise comportamental.

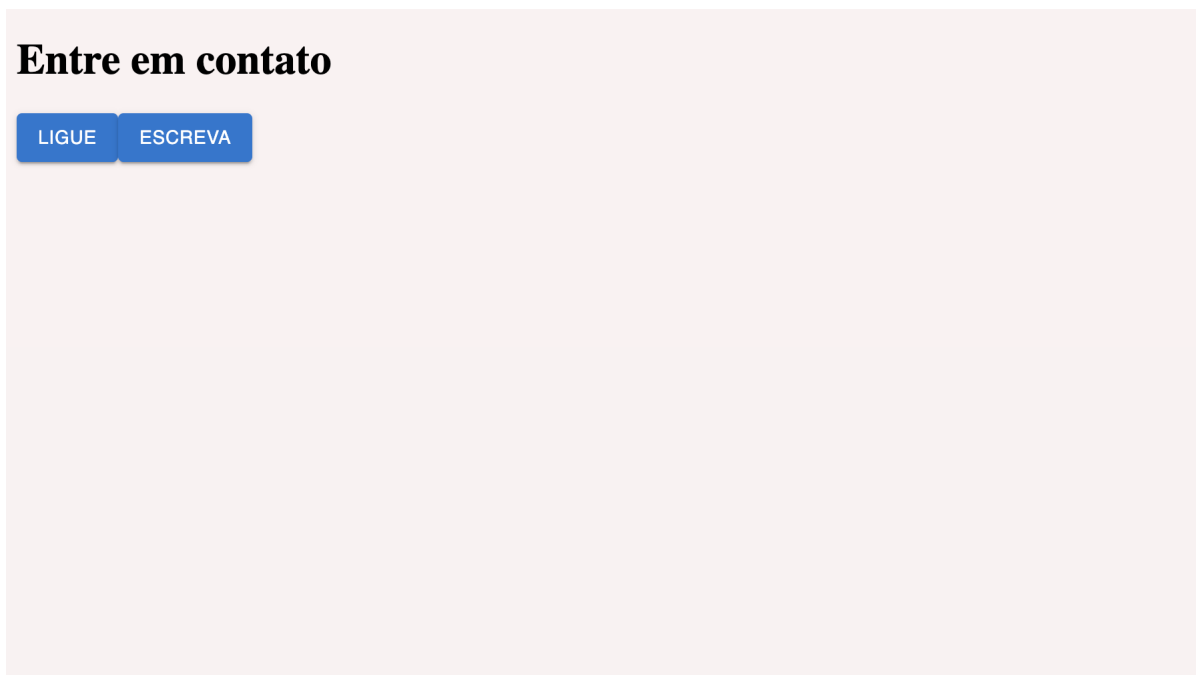
Para a utilização e captura de comportamento na plataforma teste, foi disponibilizada a aplicação via serviço Ngrok, que permite o acesso de aplicações locais em ambiente de desenvolvimento para a internet. Esse serviço cria uma URL encriptada e segura para acesso aos usuários (13).

Figura 7 – Layout da aplicação exemplo - Página de vendas



Fonte: Desenvolvida pelo autor

Figura 8 – Layout da aplicação exemplo - Página de contato



Fonte: Desenvolvida pelo autor

O objetivo é demonstrar a implementação de tagueamento em uma aplicação, e não a criação completa de uma plataforma de e-commerce. Neste exemplo, foram definidos três comportamentos de usuário que desejam ser monitorados, que são:

- Comportamento A - Vista do usuário da página. O componente a ser desenvolvido e a rota principal da aplicação.
- Comportamento B - Clique. O componente a ser desenvolvido são os botões que irão disparar as ações para a ferramenta de análise.
- Comportamento C - Input. O componente a ser desenvolvido é um campo de texto que recebe o que o usuário digitou no campo de pesquisa.

4.2.2 Desenvolvimento do Front-end e tagueamento do site

Esta etapa visa garantir que o desenvolvimento do código fonte, esteja conforme as regras definidas na etapa de planejamento do tagueamento. O tagueamento tem caminhos que podem ajudar o desenvolvedor e as equipes interessadas a realizar o mapeamento e a geração de relatórios posteriormente(1).

Deve-se iniciar o projeto pelo terminal do computador com os comandos específicos para alocar os arquivos do projeto e iniciar o desenvolvimento com a biblioteca Javascript React que nos facilita extrair informações específicas do uso da plataforma através das ações dos usuários.

O projeto é iniciado em um servidor local, possibilitando o recarregamento automático e a integração com a ferramenta que será utilizada para realizar o tagueamento. Ao criar os componentes do Front-end podem ser inseridas algumas estruturas que são utilizadas para obter ações dos usuários, e também estruturas que os desenvolvedores Front-end possam utilizar para facilitar extrair informações e ações diretamente no componente. O typescript é amplamente utilizando junto ao React para tipar os dados de entrada do usuário, por exemplo, quando o usuário cadastra um número de telefone temos a certeza que o tipo de dado recebido é um número inteiro conhecido na programação com "int", ou quando esse mesmo usuário cadastrar seu nome ou endereço vamos receber uma *string*. Isso torna a aplicação mais segura e confiável, reduzindo a possibilidade de ataques e trazendo mais segurança aos dados que serão analisados posteriormente (20).

4.2.3 Iniciando a aplicação com create-react-app

Para utilizar a ferramenta é necessário o a instalação anterior da biblioteca Node.js que gerencia os pacotes do Javascript. Na sequência entrar na pasta do projeto, e rodar a aplicação, conforme comandos abaixo:

```
npx create-react-app my-app --template typescript
```

```
cd my-app
```

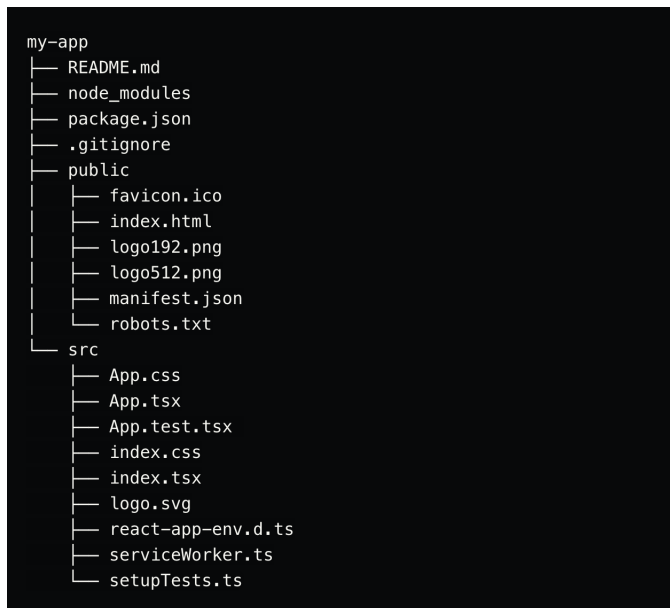
```
npm start
```

Após realizada a execução desses comandos se obterá uma estrutura de arquivos semelhante ao apresentado na figura 9.

Ao finalizar os passos anteriores pode-se adicionar ao arquivo *App.js* os dados que esse componente deve receber utilizando o Typescript. A partir desse ponto deve-se iniciar o desenvolvimento dos componentes que foram planejados e receberão as tags conectadas ao Google Analytics.

Os dados tipados podem ser utilizados posteriormente em outros componentes adicionados à aplicação ou em eventos e funções nas configurações de rotas. Para carregar uma página é

Figura 9 – arquitetura da aplicação



Fonte: Desenvolvida pelo autor

Figura 10 – Tipagem de dados typescript

```

type IHomeProps = {
  event?: MouseEvent<HTMLImageElement, MouseEvent>
  baseUrl?: string,
  idImg: string,
  typeEvent: string
  user: string,
  name: string,
  data: number,
  idUser: number
}
  
```

Fonte: Desenvolvida pelo autor

realizado um chamado de rota que será acionada quando o usuário desejar adentrar em um fluxo de páginas. O React contém estruturas que chamam ações das rotas previamente definidas.

O React trás por padrão a variável `'(e)'` ou `'(event)'` que pode ser utilizada para "ouvir" os eventos realizados no componente e capturar a forma que o usuário utilizou tal item como no input.

Ao configurar as tipagens dos componentes na página inicial é necessário adicionar o trecho de código que será renderizado ao usuário, e criar a função que ficará responsável por disparar o evento ao clique do usuário. Ele irá efetuar o preenchimento dos parâmetros e enviar os dados ao Google analytics. A figura 11 exemplifica como o componente deve ser desenvolvido.

Ao aplicar um console no navegador é possível obter meta dados dos eventos que foram

Figura 11 – Função para disparar um evento no React



```

<Button
  className="button-add"
  variant="contained"
  onClick={event => handleClickButton(event)}
>
  Adicionar no carrinho
</Button>

```

The image shows a code editor snippet for a React Button component. Two red arrows are present: one points from the top right to the `variant="contained"` attribute, and another points from the bottom right to the `onClick={event => handleClickButton(event)}` prop.

Fonte: Desenvolvida pelo autor

acionados. Possibilitando extrair informações como data, hora, o que foi digitado, *host*, ip, página do clique, id do componente, nome do componente, entre outros metadados que o React disponibiliza.

Esses dados podem ser utilizados com uma série de alterações e em diversos componentes, como captura de inputs, textos, cliques, *rollout*, tempo de tela e estado da aplicação. O React disponibiliza outras bibliotecas que ajudam a manter esses dados escondidos e de forma segura quando rodados em um servidor web.

4.2.4 Configuração da ferramenta de coleta das ações de tagging

O componente exibido na figura 11 representa um componente de botão que dispara uma função chamada `'handleClickButton(event)'` e o nome exibido em tela é "Adicionar ao carrinho". Essa função podemos disparar para o Google Analytics a tag que vai mapear o clique nesse botão, utilizando a biblioteca React-ga4.

- A. Após criar o componente de botão deve-se inicializar o terminal do computador na pasta onde está armazenado o projeto e digitar o comando:

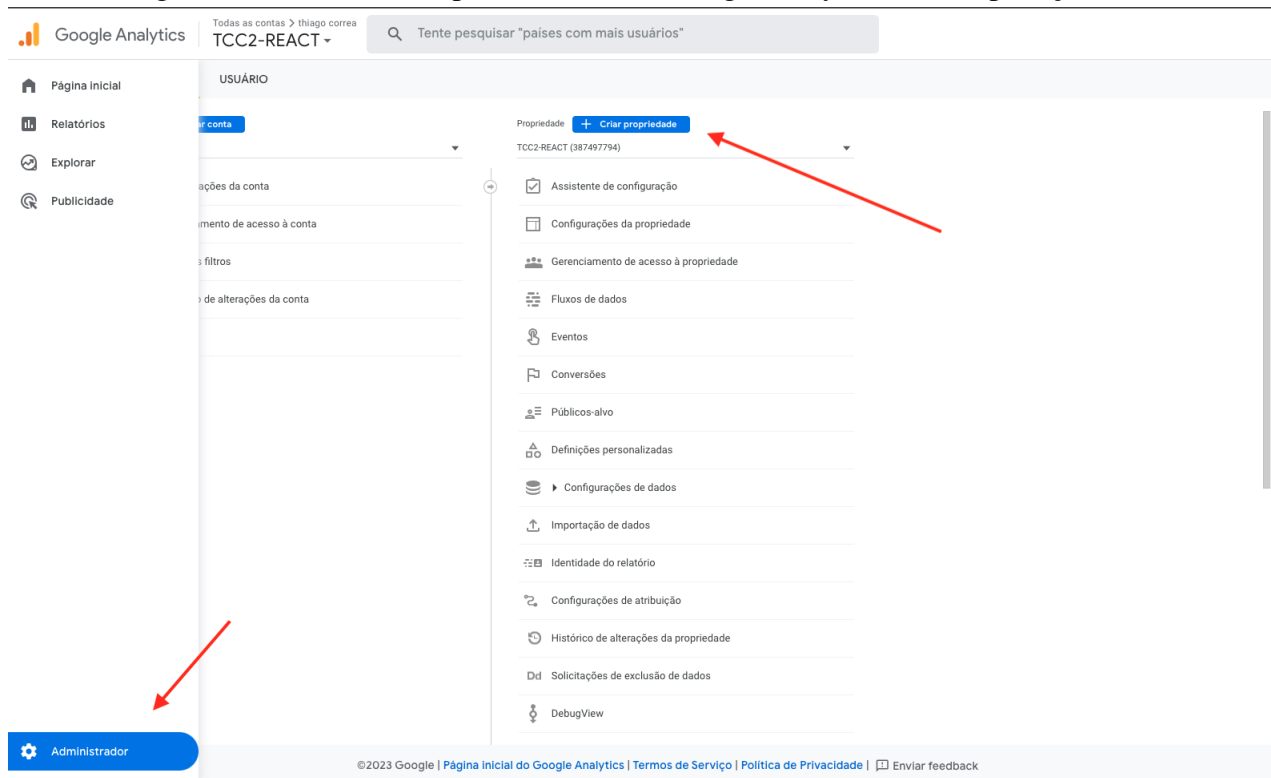
```
npm i react-ga4
```

- B. Ao realizar o login no Google Analytics deve-se iniciar a configuração dos dados que serão coletados pela ferramenta. Após acessar como Administrador selecionar "criar propriedade", para entrar com o fluxo de criação do painel de dados e suas respectivas configurações.

A seguir na Figura 12 mostra o primeiro fluxo ao clicar no botão "administrador" as setas vermelhas indicam os botões que devem ser acionados.

- C. Após dar um nome a propriedade e responder os itens de pesquisa do Google, pode-se adicionar em qual plataforma o site ou aplicativo vai utilizar as análises.

Figura 12 – Fluxo inicial para conexão do Google analytics com a aplicação



Fonte: Desenvolvida pelo autor

- D. Após selecionar a plataforma web podemos digitar o domínio em que o site está registrado, no caso deste trabalho vamos utilizar o serviço Ngrok para disponibilizar as páginas de teste. Após digitar a URL do site a plataforma fornece algumas opções de capturas automáticas e apresenta a chave de métrica que será colocada diretamente no código para autenticar o *dashboard* online do Analytics a biblioteca React-ga4.

Ao fazer as configurações iniciais no Google Analytics pode-se retornar ao código para realizar a conexão com a chave de autenticação fornecida pela ferramenta na plataforma do Google Analytics.

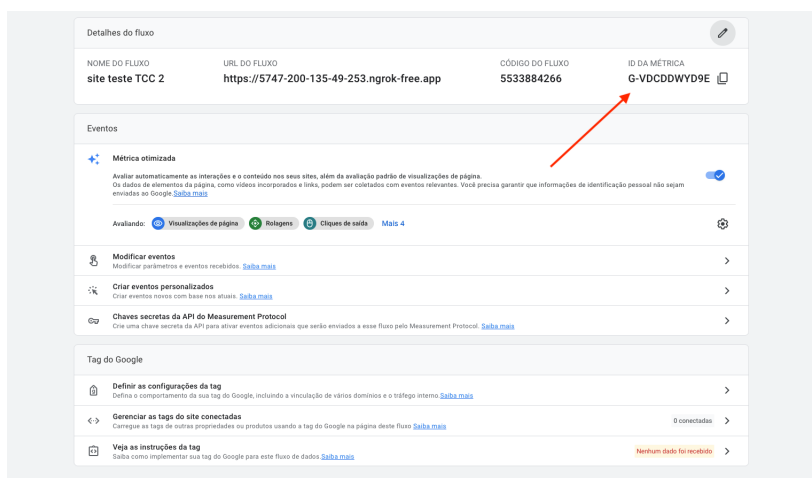
- E. No arquivo

App.tsx

onde anteriormente foram criados as estruturas de tipagem e o componente de botão, importe a biblioteca React-ga4. É importante iniciar o Google Analytics nesse arquivo, dado que todas as vezes que a aplicação for acessada pelos usuários o arquivo de *cookies* ga.js será baixado no navegador do usuário e iniciará a coleta de informações.

- F. Após a importação deve-se repassar a chave de métricas fornecidas pelo Google Analytics mostrado na Figura 14 a seguir.

Figura 13 – Chave de métricas que será conectado no código



Fonte: Desenvolvida pelo autor

Figura 14 – Chave de métricas fornecido pelo Google analytics na ferramenta

```
import { Router } from './routes/Routes'
import ReactGA from "react-ga4";
ReactGA.initialize("G-VCDDWYD9E");

function App() {
  return (
    <div>
      <Router />
    </div>
  );
}

export default App;
```

Fonte: Desenvolvida pelo autor.

Após finalizar esses passos, no projeto, é possível criar diversas tags personalizadas conforme a necessidade da aplicação que foram anteriormente mapeadas na etapa de planejamento. Na seção 4.2.1 descrevemos alguns comportamentos de interesse que serão implementados a seguir.

4.2.5 Tag de rastreo de cliques

Primeiramente na função chamada *'handleClickButton'* será criada a primeira tag. Que deverá ser acionada quando o usuário clicar no botão de "adicionar carrinho".

Para ter melhores resultados, deve-se adicionar tags em outros componentes, sendo uma boa prática que gera um maior conjunto de dados para o Google Analytics gerar relatórios mais detalhados.

Na segunda parte do tagueamento por clique, deve-se criar uma nova rota intitulada de

Figura 15 – Função de disparo de clique com a tag que será enviada ao Google analytics

```
const handlerClickButton = (e: MouseEvent<HTMLButtonElement, globalThis.MouseEvent>) => {
  alert('Produto adicionado ao carrinho')
  e.preventDefault()

  ReactGA.event({
    category: 'Carrinho',
    action: 'Adicionado ao carrinho',
    label: 'Clique no botão de adicionar ao carrinho',
  })
}
```

Fonte: Desenvolvida pelo autor

contact, geralmente usadas em comércios eletrônicos para entrar em contato com a administração da loja.

Para criar uma nova rota, deve-se adicionar um novo componente React intitulado de */contact* no arquivo de *Router.js* como no exemplo a baixo:

```
<Route path='/NomeDesejado' element=<ComponenteGenerico /> />
```

A figura 18 exemplifica como deve ficar o arquivo de rotas

Figura 16 – Arquivos de rotas da aplicação

```
import { BrowserRouter, Routes, Route } from 'react-router-dom'
import { Home } from '../pages/home';
import { ContactUs } from '../pages/main';

export const Router = () => {
  return (
    <BrowserRouter>
      <Routes>
        <Route index element={<Home />} />
        <Route path='/contact' element={<ContactUs />} />
      </Routes>
    </BrowserRouter>
  )
}
```

Fonte: Desenvolvida pelo autor

Após criar o arquivo de rotas, adicione ao componente de contato os botões de *Call* e *Email* que receberá as tags a ser acionadas, quando os eventos associados aos respectivos botões serem clicados pelo usuário.

A figura 17 exibe o componente de contato com as tags que dispara o evento *onClick*, enviando diretamente para o Google analytics as ações captadas no clique do usuário.

Figura 17 – Arquivos de rotas da aplicação

```
return (
  <div>
    <h1>Entre em contato</h1>
    <div>
      <Button variant="contained" onClick={()=>gaEventTracker('call')} className="btnRedirect">
        Ligue
      </Button>
      <Button variant="contained" onClick={()=>gaEventTracker('Email')} className="btnRedirect">
        Escreva
      </Button>
    </div>
  </div>
)
```

Fonte: Desenvolvida pelo autor

De acordo com (1) rastrear cliques em seu site resulta em saber que os usuários do site estão sendo redirecionados por URLs para o fluxo correto. Desta forma pode-se descobrir fontes de interesses dos usuários para futuras aplicações de anúncios e sugestão de conteúdos.

4.2.6 Tag de visualizações da página

A segunda tag aplicada, é utilizada para quantificar as vezes que a página foi visualizada por determinado visitante. Além disso a possibilidade de coletar dados como localização geográfica, idioma e dispositivo usado para acesso, sendo capturados de forma automática pelo próprio Google analytics.

A figura 16 mostra como inserir a tag na função *useEffect*, um *hook* (funções prontas do React), nativo do React que é usado para carregar ou recarregar os efeitos e ações realizadas pelo usuário em todo o ciclo de vida do componente(19).

Figura 18 – Função de disparo de clique com a tag que será enviada ao Google analytics

```
useEffect(() => {
  ReactGA._gaCommandSendPageview(window.location.pathname, window.location.search)
})
```

Fonte: Desenvolvida pelo autor

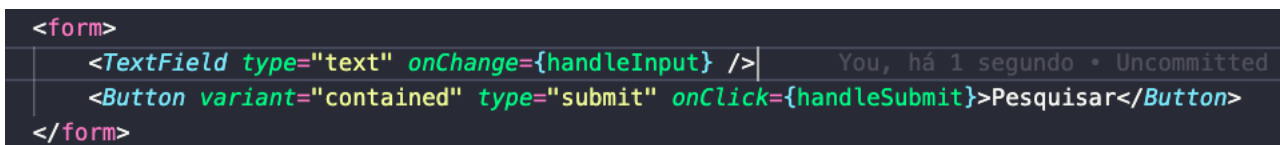
A tag de visualizações de páginas é importante para os relatórios de tagueamento pois possibilita segmentar o público por diversos pontos, como localização, número de interação por páginas e plataforma de acesso abrindo uma gama de *insights* que podem ser extraídos a partir do engajamento do usuário com a página.

4.2.7 Tag de input de pesquisa

A terceira tag aplicada, é utilizada para capturar textos digitados pelos usuários no campo de pesquisa. Além disso podemos segmentar os tipos de ações enviadas pela tag, adicionando atributos e parâmetros para complementar as informações que serão enviadas ao Google Analytics.

A figura 19 exemplifica a criação de um componente de formulário utilizando 2 outros componentes, um de campo de texto e outro de botão que efetua a submissão do formulário para o Google Analytics.

Figura 19 – Formulário com componentes de input e botão



```
<form>
  <TextField type="text" onChange={handleInput} />
  <Button variant="contained" type="submit" onClick={handleSubmit}>Pesquisar</Button>
</form>
```

Fonte: Desenvolvida pelo autor

Após desenvolver esses componentes deve-se adicionar as funções que serão disparadas quando for digitado textos de entrada via teclado no campo de pesquisa. No início do arquivo adicione as variáveis de estado que deve receber o evento quando for digitado a pesquisa, esse estado é acessível globalmente dentro da função que desejar incluir.

Figura 20 – Variável de estado que recebe texto de pesquisa



```
const [pequisa, setPesquisa] = useState('')
const history = useNavigate();
```

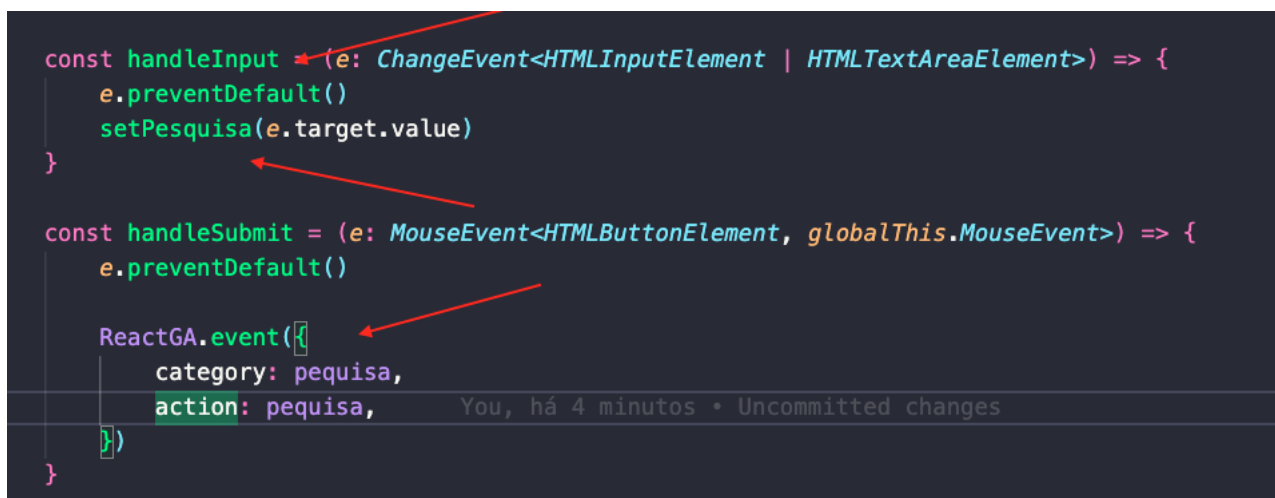
Fonte: Desenvolvida pelo autor

Após criar a variável de estado podemos conecta-la a função de input, que vai 'ouvir' todas as alterações recebidas via teclado quando o usuário digitar no campo de pesquisa, adicionando o texto recebido a variável de estado *setPesquisa()*. Quando o texto for submetido após o usuário clicar no botão de 'Pesquisar', ocorreram os eventos de disparo da tag e das informações digitada ao Google Analytics.

A imagem 21 exemplifica como se deve desenvolver as funções chamadas pelos componentes.

Finalizando essas etapas pode-se disponibilizar a aplicação para testes. O React disponibiliza um grande número de *hooks* e *features* para utilização durante o desenvolvimento do projeto conforme a necessidade do desenvolvedor, por exemplo: bibliotecas internas de CSS para interfaces amigáveis e aplicação de UX/UI designer; bibliotecas internas de tratamento de dados e submissão de requisições ao Back-end e uso de APIS; funções para o melhoramento de performance e carregamento de conteúdo dinâmico, entre outras (16).

Figura 21 – Funções de input e envio de formulário



```
const handleInput = (e: ChangeEvent<HTMLInputElement | HTMLTextAreaElement>) => {
  e.preventDefault()
  setPesquisa(e.target.value)
}

const handleSubmit = (e: MouseEvent<HTMLButtonElement, globalThis.MouseEvent>) => {
  e.preventDefault()

  ReactGA.event({
    category: pesquisa,
    action: pesquisa,
  })
}
```

Fonte: Desenvolvida pelo autor

4.3 ANÁLISE DOS DADOS COLETADOS

Após disponibilizar a página teste utilizando o serviço Ngrok, foram identificadas 3 comportamentos chaves para fornecer *insights* sobre a aplicação. Esses comportamentos foram relacionados as tags utilizadas no desenvolvimento porém, a comportamentos capturados de forma automática ao se utilizar o serviço de tagueamento do Google. Nesse contexto, será possível elaborar melhorias nas campanhas (5).

A seguir é descrito os dados comportamentais dos usuários, a quantificação de cliques e visitas da página, a geolocalização dos usuários e a quantidade de acessos que recebemos conforme a disponibilização do URL da página teste. Esses dados foram coletados com o intuito de testar a aplicação apresentada.

Contagem de eventos:

Nome do evento	Contagem	Usuários	Contagem por usuário	Receita
Add to Cart	160	9	17.78	0
page_view	92	10	9.2	0
Call	53	6	8.83	0
scroll	35	10	3.5	0
Email	28	6	4.67	0
first_visit	11	10	1.1	0

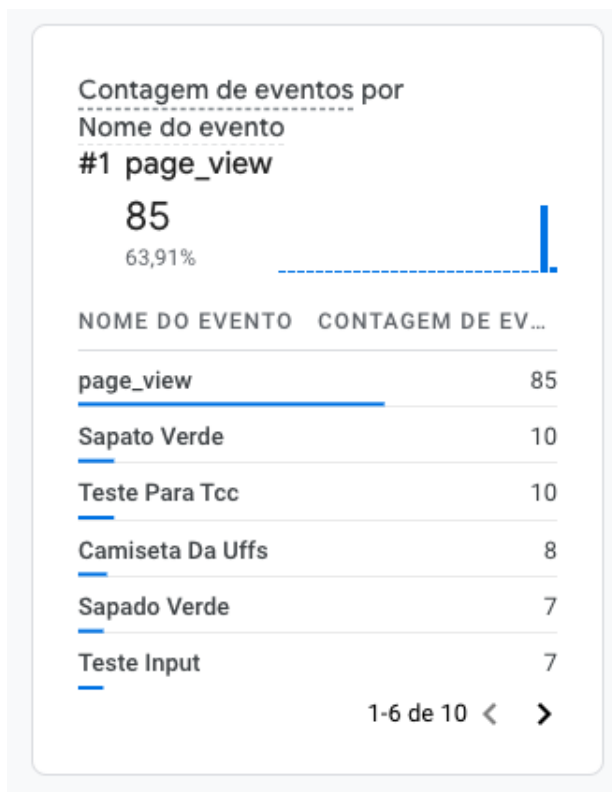
Tabela 1 – Eventos coletados pelo Google analytics

A tabela 1 apresenta a contagem de eventos registra os comportamentos A, B e D, conforme detalhado na seção 4.2.1, relacionados à análise de visitas de usuários em uma página, quantidade de cliques em botões, e o número de usuários por página. Ao analisarmos a tabela, podemos observar que o evento de 'Add to Cart' recebeu o maior número de cliques.

Inputs dos usuários:

A figura 22 exibe os dados digitados no campo de input que foram pesquisados pelo usuário dentro da plataforma. Os textos exibidos na figura tiveram finalidade de teste.

Figura 22 – Textos digitados no campo de pesquisa



Fonte: Desenvolvida pelo autor

O Google Analytics fornece várias informações que são coletadas automaticamente e auxiliam na análise dos dados. Entre eles pode-se destacar: geolocalização, visitantes por página, número de usuários novos e de revisitantes na página. A seguir é apresentado esses dados que foram coletados automaticamente.

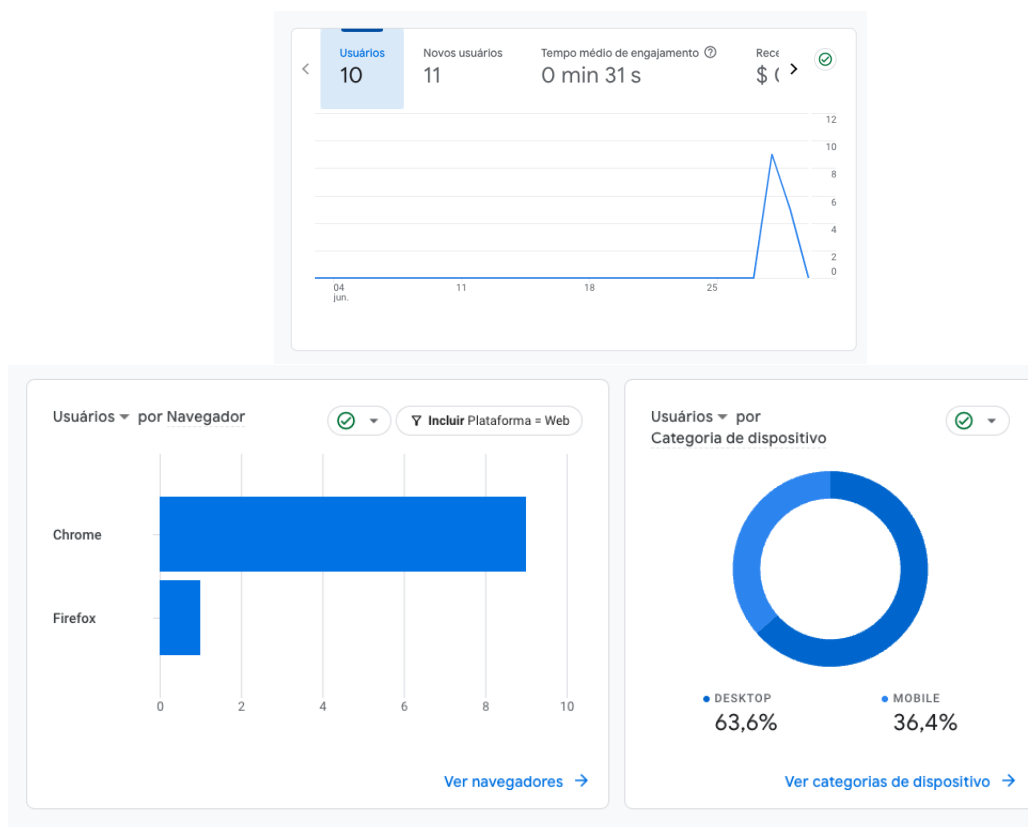
Geolocalização:

Na geolocalização fornecida pelo Google Analytics dos usuários que acessaram a plataforma, foi observado que todos eles acessaram do Brasil. De acordo com (1) relatórios que utilizam geolocalização são extremamente úteis para o direcionamento e a segmentação dos usuários. Assim pode-se efetuar a criação de campanhas de marketings baseadas na localização dos usuários e também a personalização de anúncios e conteúdos.

Total de usuários:

A Figura 23 exibe 3 gráficos que mostram informações para segmentar os visitantes. 63,6%, dos novos usuários que acessaram a plataforma entraram via web e 36,4% acessaram via mobile. No total 11 usuários acessaram a plataforma e 10 deles retornaram. Grande parte dos visitantes usaram o navegador Google Chrome para acessar a plataforma.

Figura 23 – Gráfico com total de usuários e tempo médio de engajamento



Fonte:

Desenvolvida pelo autor

Todas as métricas obtidas foram essenciais para entender o comportamento do usuário em nossa plataforma, assim futuramente pode-se aprofundar estudos e ações para melhoria de nossa plataforma e também a criação de campanhas e estratégias para a personalização e direcionamento de conteúdo dentro de nossa aplicação(9).

5 CONCLUSÃO

Este trabalho trouxe uma visão sobre a importância de coletar dados sobre o comportamento dos usuários durante o uso de uma aplicações web.

Para personalizar o monitoramento dos comportamentos e otimizar a captura desses dados é recomendado implementar tagueamento, trazendo assim flexibilidade e otimização das análises desses dados.

Nesse contexto, este trabalho teve como objetivo principal a descrição da implementação do tagueamento em aplicações web.

Para alcançar este objetivo foi apresentado um plano de ação composto por fases e atividades para implementar tagueamento em aplicações web possibilitando capturar dados do comportamento específicos no Front-end da aplicação, e integrar com o Google Analytics. O plano é composto por 4 etapas: planejamento da aplicação, desenvolvimento do front-end, configuração da ferramenta de coleta das ações de tagueamento e análise dos dados coletados. Também exemplificou passo a passo a implementação do plano utilizando o framework React, e demonstrou os dados coletados pelo Google Analytics das TAGs implementadas.

A fim de dar continuidade na pesquisa, recomenda-se os seguintes trabalhos futuros:

- Demonstrar a implementação do plano proposto utilizando outras tecnologias de desenvolvimento, entre elas: Vue, Angular, Wordpress, entre outras;
- Expandir a visão da proposta, coletar e aplicar os dados em projetos de data analytics.

REFERÊNCIAS

- 1 CLIFTON, Brian. **Advanced web metrics with Google Analytics**. [S.l.]: John Wiley & Sons, 2012.
- 2 CONTENT, Rock. **Google Analytics: O que é, como funciona e como usar**. Data de acesso: 2023. Disponível em:
<<https://rockcontent.com/br/blog/google-analytics/>>.
- 3 FACHINELLI, Ana Cristina. BIG DATA: o novo desafio para gestão. **Revista Inteligência Competitiva**, v. 4, n. 1, p. 18–38, 2014.
- 4 FILATRO, Andreia C. **Data science da educação**. first. [S.l.]: Saraiva, 2020. (9786587958446).
- 5 GOOGLE. **Google Analytics - Suporte: Configurar o Google Analytics 4 em sites e apps**. [S.l.: s.n.], 20XX.
https://support.google.com/analytics/answer/12329599?hl=pt-BR&ref_topic=2919631&sjid=9409215938431453349-SA. Acessado em 01 de julho de 2023.
- 6 _____. **Google Tag Manager - Tags de Conversão**. [S.l.: s.n.].
<https://support.google.com/tagmanager/answer/6105160?hl=pt-BR>. Acessado em: 30 de junho de 2023.
- 7 GUIA de início rápido. Jan. 2021. Disponível em:
<<https://developers.google.com/tag-manager/quickstart>>.
- 8 GUIMARÃES, Willian Pereira; BITAR, Alan Barros. GOOGLE ANALYTICS COMO FERRAMENTA ESTRATÉGICA NO MARKETING DIGITAL. **Revista Ibero-Americana de Humanidades, Ciências e Educação**, v. 9, n. 5, p. 601–616, 2023.
- 9 HASAN, Layla; MORRIS, Anne; PROBETS, Steve. Using Google Analytics to evaluate the usability of e-commerce sites. In: SPRINGER. INTERNATIONAL conference on human centered design. [S.l.: s.n.], 2009. p. 697–706.
- 10 LUIS, Rodolfo. **Coleta de dados multiplataforma: Principais desafios para aplicativos e web**. Ago. 2018. Disponível em:
<<https://blog.dp6.com.br/coleta-de-dados-multiplataforma-principais-desafios-para-aplicativos-e-web-4bfd70c57115>>.
- 11 MAGALHÃES, Jorge et al. Big data em saúde ea ciência aberta: um contributo para a gestão do conhecimento em covid-19. **Revista Prevenção de Infecção e Saúde**, v. 6, n. 10.26694, 2020.
- 12 MINISTÉRIO PÚBLICO FEDERAL. **O que é a LGPD**. [S.l.: s.n.], Data de acesso: 2023. MPF - Ministério Público Federal.
<https://www.mpf.mp.br/servicos/lgpd/o-que-e-a-lgpd>.

- 13 NGROK. [S.l.: s.n.]. <https://ngrok.com/>. Accessed on July 17, 2023.
- 14 NPM - REACT-GA. [S.l.: s.n.]. <https://www.npmjs.com/package/react-ga>. Acesso em 29 de junho de 2023.
- 15 RAFAEL G C FERREIRA, Leandro B. A. D; PINTO; Rafael A.; **Preparação e Análise Exploratória de Dados**. [S.l.]: Grupo A, 2021. (9786556902890). Disponível em: <https://app.minhabiblioteca.com.br/books/9786556902890/>. Acesso em: 18 mar. 2022.
[ca.com.br/reader/books/9786556902890/pageid/12](https://app.minhabiblioteca.com.br/books/9786556902890/pageid/12).
- 16 REACT - LEARN THE BASICS. [S.l.: s.n.]. <https://react.dev/learn>. Acesso em 29 de junho de 2023.
- 17 ROCK CONTENT. **NPM - Guia completo para iniciantes**. [S.l.: s.n.]. <https://rockcontent.com/br/blog/npm/>. Acesso em 29 de junho de 2023.
- 18 TAURION, Cezar. **Big data**. [S.l.]: Brasport, 2013.
- 19 TREINAWEB. **React: Conheça o Poder dos Hooks**. [S.l.: s.n.], Data de Acesso: 30 de Junho de 2023.
<https://www.treinaweb.com.br/blog/react-conheca-o-poder-dos-hooks>.
- 20 TYPESCRIPT - JAVASCRIPT THAT SCALES. [S.l.: s.n.]. <https://www.typescriptlang.org/>. Acesso em 29 de junho de 2023.
- 21 VASCONCELLOS, Paulo. **Como aplicamos Data Science e Machine Learning na Hotmart**. Fev. 2020. Disponível em: <<https://paulovasconcellos.com.br/como-aplicamos-data-science-e-machine-learning-na-hotmart-51ee3525f64>>.