

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS CORNÉLIO PROCÓPIO
BACHARELADO EM ENGENHARIA DE SOFTWARE

MAGNO CARVALHO DOS SANTOS

**DESENVOLVIMENTO DE UMA PLATAFORMA *MEAN STACK* DE
VEICULAÇÃO PUBLICITÁRIA POR GEOLOCALIZAÇÃO**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO
2019

MAGNO CARVALHO DOS SANTOS

**DESENVOLVIMENTO DE UMA PLATAFORMA *MEAN STACK* DE
VEICULAÇÃO PUBLICITÁRIA POR GEOLOCALIZAÇÃO**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina Trabalho de Conclusão de Curso 2, do curso de Engenharia de Software da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito para a obtenção do título de Bacharel.

Orientadora: Prof. Ma. Adriane Carla Anastácio Da Silva

CORNÉLIO PROCÓPIO
2019

DEDICATÓRIA

Dedico este trabalho à minha família,
em especial meus pais, pelo amor,
incentivo e apoio incondicional.

RESUMO

SANTOS, M. C. dos. **DESENVOLVIMENTO DE UMA PLATAFORMA MEAN STACK DE VEICULAÇÃO PUBLICITÁRIA POR GEOLOCALIZAÇÃO**. Trabalho de Conclusão de Curso (Graduação) - Engenharia de Software. Universidade Tecnológica Federal do Paraná. Cornélio Procopio, 2019.

Atualmente, existem diversas plataformas destinadas à publicidade ao setor de varejo. Contudo, observa-se que as redes sociais e/ou as plataformas de publicidade apresentam algumas particularidades para as pequenas e médias empresas do varejo, referindo-se à eficiência das mídias de publicidade e a relação entre custo e benefício destes instrumentos. Este trabalho apresenta uma plataforma de veiculação publicitária por geolocalização, uma alternativa de publicidade que permite publicar promoções, via *web*, utilizando o GPS (*Global Positioning System*), para relacionar as publicações com localização dos usuários e empresas. Para o desenvolvimento foram utilizadas as tecnologias relacionadas desenvolvimento web com o conceito *MEAN Stack*, Angular, ExpressJS, NodeJS e MongoDB.

Palavras-chave: *MEAN Stack*, Angular, ExpressJS, NodeJS e MongoDB, Geolocalização.

ABSTRACT

SANTOS, M. C. dos. **DEVELOPMENT OF A MEAN STACK PLATFORM FOR GEOLOCALIZED ADVERTISING VEHICLE.** Trabalho de Conclusão de Curso (Graduação) - Engenharia de Software. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2019.

There are currently several platforms for advertising to the retail sector. However, it is observed that social networks and / or advertising platforms present some particularities for small and medium retail companies, referring to the efficiency of advertising media and the cost-benefit ratio of these instruments. This paper presents a geolocation advertising platform, an advertising alternative that allows the publication of promotions, via the web, using the GPS (Global Positioning System), to relate the publications with the location of users and companies. For the development were used web development technologies related to the concept MEAN Stack, Angular, ExpressJS, NodeJS and MongoDB.

Keywords: MEAN *Stack*, Angular, Express.JS, Node.JS e MongoDB, Geolocation.

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
CAPTCHA	<i>Completely Automated Public Turing test to tell Computers and Humans Apart</i>
CNPJ	Cadastro Nacional da Pessoa Jurídica
CSS	<i>Cascading Style Sheet</i>
GPS	<i>Global Position System</i>
HTML	<i>Hyper Text Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IOS	Sistema operacional móvel da Apple
JSON	<i>JavaScript Object Notation</i>
JWT	<i>JSON Web Token</i>
MEAN	MongoDB, Express, Angular, NodeJS
NoSQL	<i>Not Only SQL</i>
NPM	<i>Node Package Manager</i>
UID	Firestore ID <i>Unique</i>
UTFPR	Universidade Tecnológica Federal do Paraná

LISTAGEM DE FIGURAS

Quadro 1 - Exemplo de aplicações que utilizam geolocalização	18
Quadro 2 - Aplicações similar a plataforma proposta	18
Quadro 3 - Principais características.	19
Quadro 4 - Requisitos funcionais do domínio empresa	22
Quadro 5 - Requisitos funcionais do domínio cliente	23
Quadro 6 - Requisitos não-funcionais de ambiente	23
Quadro 7 - relação das permissões das collections com as respectivas aplicações	27
Quadro 8 - Listagem do Product Backlog da aplicação desenvolvida	28
Quadro 9 - Listagem de itens do schema do objeto Empresa	37
Quadro 10 - Campos do formulário de cadastro de promoção.	42
Quadro 11 – navegadores compatíveis com HTML Geolocation API: getCurrentPosition	47
Quadro 12 – Suporte às tecnologias usadas pelos principais navegadores	55

LISTA DE TABELAS

FIGURA 1 - Relação arquitetura MEAN STACK	14
FIGURA 2 - Visão geral ilustrada do processo de desenvolvimento do Scrum Solo	24
FIGURA 3 - Interação das tecnologias utilizadas	25
FIGURA 4 - Listagem de coleções e suas relações	26
FIGURA 5 - Relação das tecnologias com aplicação do lado servidor	30
FIGURA 6 - Página inicial Firebase	30
FIGURA 7 - Terceira dimensão Material Design	31
FIGURA 8 - Comando para verificar Angular Version	32
FIGURA 9 - Página inicial domínio empresa resolução 960x600	33
FIGURA 10 - Página inicial domínio empresa resolução iPhone 5/SE 320x568	34
FIGURA 11 - Página de cadastro de nova empresa	35
FIGURA 12 - Página de confirmação de email da empresa nova empresa	36
FIGURA 13 - Estrutura do método de autenticação	38
FIGURA 14 - Página de cadastro de nova empresa	39
FIGURA 15 - Diagrama de sequência da autenticação da aplicação	40
FIGURA 16 - Menu lateral dimensões 960x600	41
FIGURA 17 - Formulário de cadastro de promoções	42
FIGURA 18 - Formulário de cadastro de promoções	43
FIGURA 19 - Tela de editar perfil	44
FIGURA 20 - Listagem de relatórios de favoritos	45
FIGURA 21 - Tela de autenticação domínio cliente em diferentes resoluções.	46
FIGURA 22 - Permissão de usar a localização do usuário	48
FIGURA 23 - Organização do layout do domínio cliente	49
FIGURA 24 - Card de listagem de promoções	50
FIGURA 25 - Listagem de promoções	51
FIGURA 26 - Tela de informações da empresa	52
FIGURA 27 - Listagem de promoções favoritas	53

SUMÁRIO

1	INTRODUÇÃO	11
1.1	CONTEXTUALIZAÇÃO	11
1.2	ESCOPO E MOTIVAÇÃO	12
1.3	OBJETIVO PRINCIPAL	13
1.4	OBJETIVOS ESPECÍFICOS	13
1.5	ORGANIZAÇÃO DO TEXTO	13
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	MEAN STACK	15
2.1.1	MONGODB	16
2.1.2	EXPRESSJS	16
2.1.3	ANGULAR	17
2.1.4	NODE.JS	17
2.2	GEOLOCALIZAÇÃO	18
2.2.1	APLICAÇÕES QUE UTILIZAM GEOLOCALIZAÇÃO	18
2.3	SIMILARES	19
3	TECNOLOGIAS DO DESENVOLVIMENTO	21
3.1	HTML E CSS	21
3.2	MATERIAL DESIGN	21
3.3	ANGULAR MATERIAL 2	21
3.4	JAVASCRIPT	21
3.5	TYPESCRIPT	22
3.6	FIREBASE	22
3.7	LEVANTAMENTO DE REQUISITOS	22
3.7.1	REQUISITOS FUNCIONAIS DO DOMÍNIO EMPRESA	23
3.7.2	REQUISITOS FUNCIONAIS DO DOMÍNIO CLIENTE	24
3.7.3	REQUISITOS NÃO-FUNCIONAIS DE AMBIENTE	24
3.8	PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	25
3.9	ARQUITETURA DO SOFTWARE	26
3.10	BANCO DE DADOS	26
4	DESENVOLVIMENTO DO SOFTWARE	29
4.1	PRODUCT BACKLOG	29
4.1.1	SPRINT 1	30

4.1.2	SPRINT 2	33
4.1.3	SPRINT 3	40
4.1.4	SPRINT 4	46
4.1.5	SPRINT 5	49
5	CONSIDERAÇÕES FINAIS	55
5.1	DIFICULDADES E LIMITAÇÕES	55
5.2	TRABALHOS FUTUROS	56

1 INTRODUÇÃO

Este capítulo apresenta informações sobre o contexto e justificativa no qual se baseia o desenvolvimento do projeto, bem como, o objetivo geral, objetivos específicos e organização do texto.

1.1 CONTEXTUALIZAÇÃO

Segundo o estudo de Rocha e Alves (2010) publicidade em mídias *online*, está relacionada ao ambiente virtual da Internet, pode ser localizada em *blogs*, *e-commerces* e redes sociais como o Facebook e LinkedIn, redes que transformaram o consumo de seus respectivos usuários, ao despertar novas experiências e gerar um ambiente de criação de mídias especializadas nestes públicos.

Nos últimos anos, tem ocorrido um crescente interesse na publicidade em mídias *online* por parte das empresas em todo mundo. Este fato pode ser ilustrado por dados da 19ª Pesquisa Global de Entretenimento e Mídia 2018-2022, da empresa PwC, realizada em 53 países, que analisou gastos dos consumidores e anunciantes (EEDEN; CHOW, 2018).

Outro dado relevante é apresentado pela EMARKETER, uma empresa especializada em pesquisas na área de marketing digital, que prevê para 2019 um investimento do mercado americano de publicidade em mídias *online* que ultrapassará o valor gasto nas mídias tradicionais pela 1ª vez na história (WAGNER, 2019).

Para Nonnenmacher (2012), os *smartphones* atuais aumentam a capacidade do usuário de criar conteúdo e interagir com outros usuários em tempo real, por meio das funções agregadas a estes dispositivos, como o GPS, a conexão com Internet e a câmera, entre outras tecnologias emergentes.

O lançamento do *smartphone* iPhone da Apple® em 2007 iniciou uma profunda e transformadora inovação econômica, que tornou os celulares a porta de entrada de pessoas no mundo da Internet, conseqüentemente, criou uma forte demanda por aplicativos para *smartphones* e a maneira como as informações são divulgadas na Internet (MANDEL; LONG, 2017). A última, com maior destaque, reflete na indústria do marketing *online* ou publicidade digital. O desafio é acompanhar a grande e rápida expansão do crescimento da própria Internet. Sendo que a publicidade digital é a principal fonte de receita de grandes empresas, Google

e Facebook com participação de 20% do total da publicidade global (RIBEIRO, 2017).

1.2 ESCOPO E MOTIVAÇÃO

Para Rocha e Alves (2010) a publicidade digital no Brasil tem muito espaço para se desenvolver, quando associada às tecnologias e previsões. Deste modo, o cenário evidencia possibilidades para as empresas desenvolverem ações publicitárias que alcancem o cidadão comum, que passa cada vez mais tempo acessando a Internet.

De acordo com Amaral (2018):

O aumento da concorrência e da migração do comércio via Internet, possibilitou ao marketing também atuar no mundo digital. As organizações que não souberem se ajustar a essa mecânica perderão espaço para as empresas que souberem tornar o consumo casual em uma relação mais estimada.

A previsão de expansão de acesso à Internet com o advento da rede 5G em todo mundo (JONSSON *et al*, 2018) apresenta um cenário atrativo para a publicidade *online*, com a popularização da Internet, as empresas buscam novos meios de divulgação das informações, diferente de antes, onde se usava meios tradicionais de comunicação para veiculação de publicidade, como as redes de televisão, rádio, jornais e revistas ao grande público (ROCHA; ALVES, 2010).

O avanço da tecnologia trouxe a mudança do comportamento do consumidor (AMARAL, 2018). Para Figueredo (2017) a publicidade tradicional pode não ter a mesma eficiência de antes, como meio de informação, atração e persuasão sendo a hora dos profissionais de publicidade ir à busca de outras soluções e mídias, revendo as técnicas de produção publicitária e reinventando a publicidade.

A construção de uma plataforma de veiculação de publicidade por geolocalização ao público regional, teve como objetivo incluir empresas de todos os portes no meio virtual. Outro fator considerado é o atual cenário da economia brasileira, propício ao incremento da publicidade *online*, considerando que 72% da população declaram que mudaram a rotina financeira na busca por menores preços (SPC BRASIL e CNDL, 2018).

1.3 OBJETIVO PRINCIPAL

Desenvolver uma aplicação *web* onde todos os módulos de interface de acesso até a aplicação do lado servidor sejam escritos em linguagem *TypeScript* para ser modelo para futuros estudos no desenvolvimento de software utilizando as tecnologias relacionadas ao acrônimo *MEAN Stack*, sendo elas respectivamente MongoDB, Express, Angular e NodeJS, junto a palavra *Stack* do inglês “pilha”.

O escopo da aplicação seguiu uma proposta de produto que surgiu durante uma competição de empreendedorismo, o *Startup Weekend* 2016¹, realizada em Londrina, estado do Paraná.

A plataforma permitirá obter dados da localização dos usuários, via geolocalização e manipula esses dados no lado servidor, sendo na sua maioria desenvolvida na Linguagem de programação já mencionada *TypeScript* e suas bibliotecas.

A aplicação desenvolvida disponibiliza um conjunto de informações relacionadas a produtos em promoções que considera a distância entre da localização dos clientes e empresas.

1.4 OBJETIVOS ESPECÍFICOS

Os objetivos específicos permitiram a criação de duas *interfaces* de acesso via navegador web e uma aplicação no lado servidor, ambas baseada na arquitetura de desenvolvimento de uma aplicação *MEAN Stack*, são elas:

- O desenvolvimento de um domínio *web* que permitir ao usuário consultar via navegador no seu dispositivo, utiliza como parâmetro sua localização as promoções próximas vinculadas à plataforma.
- O desenvolvimento de uma aplicação *web* para divulgação das promoções e publicidades pelas empresas, onde as mesmas adicionam sua localização e publicam os produtos em ofertas.
- O desenvolvimento de uma API para a integração e manipulação dos dados gerados pelas plataformas web.

1.5 ORGANIZAÇÃO DO TEXTO

A estrutura do presente trabalho está dividida em 5 Capítulos. No capítulo 2 é apresentado a fundamentação teórica, produtos similares, contextualização das

¹<http://blogdoaluno.utfpr.edu.br/?p=9529>

tecnologias relacionadas ao conceito *MEAN Stack* e geolocalização. O capítulo 3 aborda as demais tecnologias aplicadas no desenvolvimento, os requisitos, a arquitetura da plataforma, modelo de desenvolvimento e banco de dados. O Capítulo 4 refere-se à o processo de desenvolvimento e das *sprints* com suas descrições. O Capítulo 5, são apresentadas as limitações do trabalho e as considerações acerca do desenvolvimento.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta informações do trabalho resultantes do levantamento bibliográfico e embasam o desenvolvimento deste projeto, apresentando produtos similares, o uso de geolocalização, o desenvolvimento aplicações utilizando o conceito do acrônimo *MEAN Stack* e suas respectivas tecnologias.

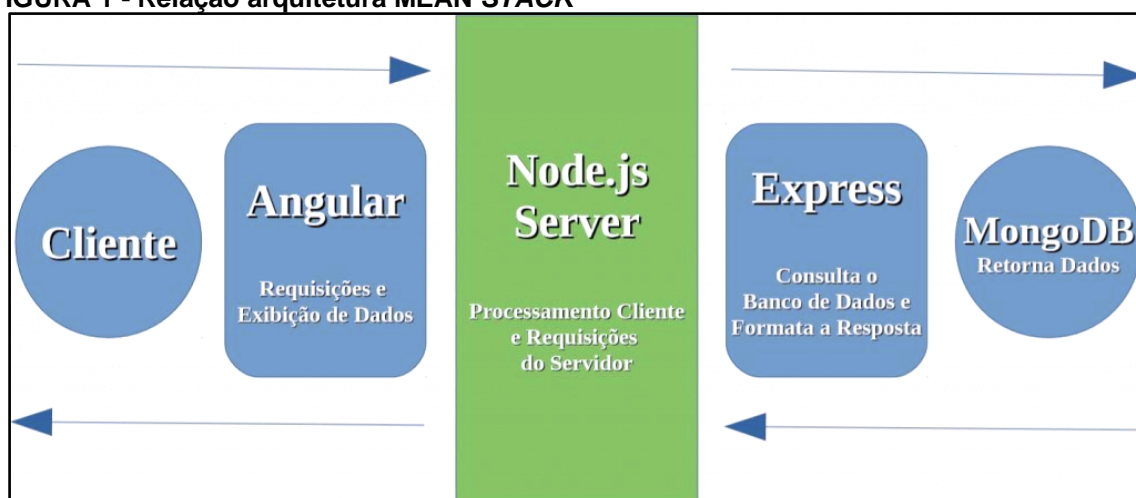
2.1 MEAN STACK

Segundo Bera, Mine e Lopes (2015), o *MEAN Stack* é o conjunto de tecnologias que possibilita o desenvolvimento de aplicações web escaláveis, utiliza a sintaxe de escrita do *JavaScript* desde a interface do usuário até a aplicação do lado servidor, de acordo com a mesma pesquisa.

Os benefícios em utilizar uma aplicação baseada em *JavaScript* abrange dois fatores principais: (i) Escalabilidade; que se define pela disponibilidade de serviços requisitados a proporções crescentes de novos usuários; e (ii) Performance; o desempenho obtido levando em conta a escalabilidade e recursos consumidos.

A Figura 1 apresenta a arquitetura MEAN e a relação com as tecnologias.

FIGURA 1 - Relação arquitetura MEAN STACK



Fonte: Adaptada de (OCTALMIND, 2018)

Segundo Rodrigues e Zuchi (2017) o desenvolvimento web em geral temos duas áreas distintas que juntas se complementam *front-end* e *back-end*. A vantagem do uso do *MEAN Stack* como pilha de desenvolvimento, permite a integração entre as áreas citadas a sintaxe da linguagem de programação *JavaScript*. O uso dessa linguagem permite um ganho de produtividade e curva de aprendizado.

O estudo de Bonfim e Liang (2014), demonstrou que uso do conjunto de tecnologias do Acrônimo *MEAN Stack*, se mostra uma opção no desenvolvimento escalável eficiente em comparação a outras tecnologias.

A descrição das tecnologias relacionadas ao acrônimo MEAN e apresentado a seguir:

2.1.1 MONGODB

De acordo com Hows, Membrey E Plugge (2015), o MongoDB é um banco de dados NoSQL, usa um modelo de dados flexível orientado a documentos JSON (*JavaScript Object Notation*) estes documentos podem mudar facilmente, conforme a necessidade e evolução da aplicação a ser desenvolvida.

O modelo de dados de documentos do MongoDB facilita o armazenamento e a combinação de dados de qualquer estrutura, sem abrir mão de regras sofisticadas de validação, acesso flexível a dados e rica funcionalidade de indexação. Você pode modificar dinamicamente o esquema sem tempo de inatividade - essencial para aplicativos em rápida evolução (MORGAN, 2017, tradução nossa).

No banco de dados MongoDB temos coleções de documentos, onde cada documento é independente, este contém todos os dados que precisam sendo diferentes dos bancos de dados relacionais, suas consultas e operações são realizadas de forma assíncrona, assim como o NodeJS e segundo Politowski e Mara (2014).

O fato de que os bancos NoSQL terem nascido para suprir a demanda por performance, deixando outros detalhes, como atomicidade, por exemplo, em segundo plano(..) Bancos de dados NoSQL e Relacionais utilizam paradigmas diferentes e, por sua vez, possuem finalidades diferentes, mas com o mesmo propósito, persistir dados.

2.1.2 EXPRESSJS

O ExpressJS é um framework e componente *middleware* que facilita o desenvolvimento de aplicações *web* com o NodeJS, permite instanciar servidores *web* e receber requisições HTTP, organizando e garantido a segurança e a entrega das informações trafegadas (MARDAN, 2019).

2.1.3 ANGULAR

O Angular é um *framework JavaScript* desenvolvido e mantido pela empresa Google em parceria com a Microsoft. criado para permitir a fácil reutilização de código e manutenção. Ele possui um conjunto de ferramentas e componentes desenvolvidos pela comunidade que permitem ao desenvolvedor agilizar o desenvolvimento, conhecidos como *packages*, disponíveis no repositório NPM (*Node Package Manager*) que pode ser acrescentada a aplicação (FONSECA JR; FONSECA; DE LIMA RANGEL, 2018).

2.1.4 NODE.JS

O Node.JS é uma plataforma para o desenvolvimento de aplicações do lado servidor que utiliza como a linguagem padrão o *JavaScript*, podendo ser escrita em *TypeScript*, ideal para criação de aplicações *web* no lado servidor de com foco em escalabilidade (MORAES, 2016).

Segundo Handy (2011) a plataforma é popular entre os desenvolvedores *web* e utilizada por grandes empresas como LinkedIn, Microsoft, GitHub e MySpace.

A arquitetura do NodeJS é composta em sua maior parte por componentes desenvolvidos em C++ e em *JavaScript* (MORAES, 2016). Sendo criado para ser estendido por meio de módulos, sendo instalados por meio do gerenciador de pacotes conhecido como NPM (*Node Package Manager*), com a possibilidade de implementar diversos componentes *middleware* que facilitem o desenvolvimento de aplicações *web* (WITTERN; SUTER; RAJAGOPALAN; 2015).

Os módulos disponíveis no NPM servem para facilitar diversos aspectos relacionados à aplicação, como instalação de dependências, a compilação e a atualização de pacotes (MORAES, 2019).

Segundo Pereira (2016) as principais características do NodeJS, estão relacionados a sua arquitetura não-bloqueante sendo um *Single Thread*, com a programação feita de forma assíncrona com seus recursos compartilhados, com todas as operações de entrada e saída com suas execuções paralelas e sendo orientado a eventos.

2.2 GEOLOCALIZAÇÃO

As aplicações modernas que usam o geolocalização de dados dos seus usuários só foram possíveis com o advento do Sistema de Posicionamento Global – GPS (*Global Positioning System*). As tecnologias capazes de lidar com dados espaciais deixaram de ser de alto custo e restritas às instituições formais e passaram a ser mais acessíveis ao cidadão comum (GOODCHILD, 2007).

Para Machado (2015) a necessidade do ser humano de saber “onde se está”, levou o desenvolvimento de diversas tecnologias, entre elas, a bússola e o GPS.

Na sociedade atual o GPS é utilizado na navegação para carros, navios, aviões até no monitoramento dos movimentos de placas tectônicas (BERNARDI; LANDIM, 2002). Ao considerar o potencial de um GPS muitas aplicações foram e estão sendo desenvolvidas para auxiliar as pessoas em suas tarefas.

2.2.1 APLICAÇÕES QUE UTILIZAM GEOLOCALIZAÇÃO

No mercado existem diversas aplicações e jogos que usam o recurso de geolocalização com GPS, alguns deles apresentados a seguir no Quadro 1.

Quadro 1 - Exemplo de aplicações que utilizam geolocalização

Aplicação	Descrição
Google Maps	Um software de mapas disponibilizado pela Google com informações de mapas, tráfego, rotas e imagens de satélites, com API de integração com diversos software e linguagem de programação (ZAMIR 2010).
Waze	Um software de informações de tráfego e rotas desenvolvido pelo startup Waze Mobile, lançado em 2008 (ADORNES, 2016).
Moovit	Aplicativo voltado para experiência do transporte público, presente em vários países pelo mundo (CAMPOS et al., 2016).
PokemonGO	Um jogo de realidade virtual que usa o recurso de geolocalização e câmera dos dispositivos móveis para interação entre os usuários (DE CARLI; DE ARAÚJO GASTAL; GOMES, 2016).

Fonte: Autoria Própria

2.3 SIMILARES

Os produtos apresentados a seguir, no Quadro 2 utilizam as mesmas premissas que serão aplicadas no desenvolvimento da plataforma desenvolvida e tem no seu modelo de negócio algo relacionado a cupom de desconto ou listagem de promoções.

Quadro 2- Aplicações similar a plataforma proposta

Aplicação	Descrição
Cuponeria	Um serviço que disponibiliza aos seus clientes ofertas e cupons de desconto de diversos tipos de produtos e lojas de forma gratuita, com filtros por localização e produtos, possui versões para Android e IOS (CUPONERIA, 2019).
Pelando	Uma comunidade onde os usuários postam as ofertas e promoções encontradas pela Internet, permite a interação por meio de votações, comentários e fotos das compras, possui versões para Android e IOS (PELANDO, 2019).
Groupon	Um <i>site</i> agregador de cupons de desconto, com filtro por localização, possui versões para Android e IOS (GROUPON, 2019).
Promobit	Uma comunidade de compartilhamento de cupons de desconto, com membros por todo Brasil, nas versões Android e IOS (PROMOBIT, 2019).

Fonte: Autoria Própria

Cada aplicação citada no Quadro 2 uma ou mais de suas características similares ao software desenvolvido.

O Quadro 3 apresenta um resumo das características em comum a plataforma desenvolvida com os produtos citado do Quadro 2, relacionado funções e características.

Quadro 3 - Principais características.

Funções x aplicação	Pelando	Groupon	Cuponeria	Promobit	Gastando pouco*
GPS	não	sim	sim	não	sim
Login com Redes Sociais	sim	sim	sim	sim	sim
Produtos da Região	não	sim	não	não	sim
Ofertas baseadas no perfil do usuário	não	não	sim	não	não
Qualificação da Promoção	sim	não	sim	não	não
Aplicativo Android e IOS	sim	sim	sim	sim	não

***software desenvolvido**

Fonte: Autoria Própria

3 TECNOLOGIAS DO DESENVOLVIMENTO

Listagem das principais tecnologias e ferramentas utilizadas no desenvolvimento da plataforma.

3.1 HTML E CSS

HTML é a acrônimo em inglês para *Hyper Text Markup Language*, a linguagem de marcação e estruturação de conteúdo, utilizada para desenvolver *websites* e aceito por diversos navegadores e dispositivos, sendo desenvolvida por Tim Berners-Lee em 1990 (TORRES, 2018).

O CSS é a abreviação do termo em inglês *Cascading Style Sheet*, em português, Folhas de Estilo em Cascata, responsável pela apresentação e estilização dos componentes, como cores, posição, bordas, fontes, tamanhos de letras e imagens, entre outras customizações (SILVA, 2018).

3.2 MATERIAL DESIGN

Material Design foi anunciada em 2014 pela empresa Google e mantido pela mesma desde então, sendo um *framework* para padrão de desenvolvimento do visual de aplicações multiplataforma, com a premissa de trazer uma experiência tridimensional ao espaço de duas dimensões da tela do dispositivo que está sendo utilizado para acessar aplicação ou *site*, sintetizando os princípios de suas características para o visual intuitivo, fluido e de simples compreensão que permite no desenvolvimento uma experiência unificada entre as plataformas e tamanhos de dispositivos (MEW, 2015).

3.3 ANGULAR MATERIAL 2

Angular Material 2 é a biblioteca para Angular que implementa o padrão Material design, contém um conjunto de componentes de interface que são manipulados por meio de suas *tags* próprias com atributos de nomes autoexplicativos, O Angular Material 2 possui componentes como animações, tabelas, temas e segundo a documentação do oficial suporta os principais navegadores: Chrome, Firefox, Safari, Opera e Edge (KASAGONI, 2017).

3.4 JAVASCRIPT

A linguagem ECMAScript, popularmente conhecida por *JavaScript*, possibilita a revolução na maneira como a *web* trabalhava, desde sua criação em 1995, por meio da execução do *script* com as instruções no lado do navegador do

cliente, sendo implementado pelos principais navegadores, ampliando sua popularização (POWERS, 2010).

De acordo com Bera, Mine e Lopes (2015), *JavaScript* é uma linguagem orientada a eventos, ou seja, suas operações são executadas somente quando as mesmas são requisitadas em alguma parte da aplicação.

3.5 TYPESCRIPT

O *TypeScript* é a evolução da linguagem *JavaScript* desenvolvido pela Microsoft. Projetada para agilizar o desenvolvimento das aplicações, possui recursos comuns das linguagens orientadas a objeto, sendo fortemente marcado, com tipos de dados, pode ser definido interfaces e serviços. Angular utiliza *TypeScript* como linguagem padrão de desenvolvimento (ALVES, 2019).

O *TypeScript* foi usado no desenvolvimento da aplicação do lado servidor, sendo compilado em *JavaScript* utilizando *framework* Gulp.

3.6 FIREBASE

Firebase é uma coletânea de serviços disponibilizados pela empresa Google para auxiliar os desenvolvedores de aplicações, com funções de banco de dados em *real time*, serviços de análise de dados, inteligência artificial, armazenamento de arquivos dos usuários, monitoramento de desempenho, hospedagem e autenticação (MORONEY, Laurence; MORONEY; ANGLIN., 2017).

Para o desenvolvimento do software foi utilizado o serviço *Firebase Auth* como gerenciador de autenticação dos usuários utilizando redes sociais ou e-mail garantido a segurança dos dados trafegados, com uso de criptografia assimétrica, para implementar o protocolo OAuth.

O serviço *Firebase Storage* de *upload* de arquivos foi utilizado para hospedagem das imagens publicadas na aplicação.

3.7 LEVANTAMENTO DE REQUISITOS

Segundo Pressman (2011), o levantamento dos requisitos é baseado em tarefas, levantado por todos os envolvidos diretamente e indiretamente, o processo de criação de um conjunto de requisitos que compõem o escopo do projeto. Durante o levantamento de requisitos, existe a necessidade da classificação do tipo de requisito levantado, sendo assim organizados em funcionais e não-funcionais (SOMMERVILLE, 2011).

Segundo Pressman e Maxim (2016), os requisitos funcionais formam um conjunto das definições que o software deve fazer, entre armazenamento de dados, detalhes técnicos manipulação e os requisitos não-funcionais formam um conjunto de requisitos que contém itens relacionados à qualidade e desempenho do software.

O levantamento de requisitos da plataforma desenvolvida está dividido em dois módulos, domínio cliente e domínio empresa.

3.7.1 REQUISITOS FUNCIONAIS DO DOMÍNIO EMPRESA

O Domínio empresa permite publicar as promoções do estabelecimento e visualizar quantidade de vezes que a promoção foi favoritada pelos clientes na aplicação.

O Quadro 4 apresenta os requisitos funcionais do domínio empresa.

Quadro 4 - Requisitos funcionais do domínio empresa

RFPE-001	O domínio empresa contém autenticação para o usuário com e-mail e senha.
RFPE-002	O domínio empresa usa a localização do navegador do usuário para preenchimento automático do endereço aproximado da empresa.
RFPE-003	O domínio empresa permite o cadastro de novas promoções pelo usuário da empresa.
RFPE-004	O domínio empresa permite o cadastro e edição das informações da empresa.
RFPE-005	O domínio empresa permite que o usuário da empresa visualize a quantidade de favoritos de suas promoções cadastradas.
RFPE-006	O domínio empresa permite a edição das promoções já cadastradas.

Fonte: Autoria Própria

3.7.2 REQUISITOS FUNCIONAIS DO DOMÍNIO CLIENTE

No domínio cliente, o usuário pode acessar as promoções publicadas na plataforma, salvar as promoções favoritas e visualizar as informações das empresas e suas respectivas publicações. O quadro 5 apresenta os requisitos funcionais do domínio cliente.

Quadro 5 - Requisitos funcionais do domínio cliente

RFC-001	O domínio cliente permite ao usuário efetuar autenticação na aplicação utilizando perfil da rede social Facebook ou Google.
RFC-002	O domínio cliente requisita ao usuário a permissão de acessar sua localidade.
RFC-003	O domínio cliente capturar a localização do usuário e enviá-la ao servidor como parâmetro da busca.
RFC-004	O domínio cliente calcular a distância entre o usuário e a localização das empresas com promoções ativas próximas.
RFC-005	O domínio cliente permite que o usuário acesse informações sobre as empresas ativas com promoções.
RFC-006	O domínio cliente filtrar a exibição de promoções pelas categorias informadas.
RFC-007	O domínio cliente permite que o usuário salve suas promoções preferidas.
RFC-008	O domínio cliente permite ao usuário liste suas promoções favoritas.
RFC-009	O domínio cliente na visualização das informações da empresa mostra a localização da mesma em mapa.

Fonte: Autoria Própria

3.7.3 REQUISITOS NÃO-FUNCIONAIS DE AMBIENTE

Os requisitos não-funcionais de ambiente são apresentados no Quadro 6.

Quadro 6 - Requisitos não-funcionais de ambiente

RNFA-001	O Código escrito em linguagem <i>TypeScript</i> .
RNFA-002	O Sistema utiliza IDE de desenvolvimento Visual Studio Code.
RNFA-003	O Sistema tem conectividade com a Internet para conexão com aplicação do lado servidor.
RNFA-004	O Sistema tem acesso à localização do usuário enquanto utiliza a aplicação.
RNFA-005	O Sistema utiliza chave assimétrica na implementação de segurança das informações dos usuários utilizando protocolo OAuth.
RNFA-007	O Sistema tem o seu código disponibilizado em repositório público Github.
RNFA-008	O Sistema utiliza reCAPTCHA nos formulários
RNFA-009	O Sistema faz o upload de imagens no serviço Firebase Storage

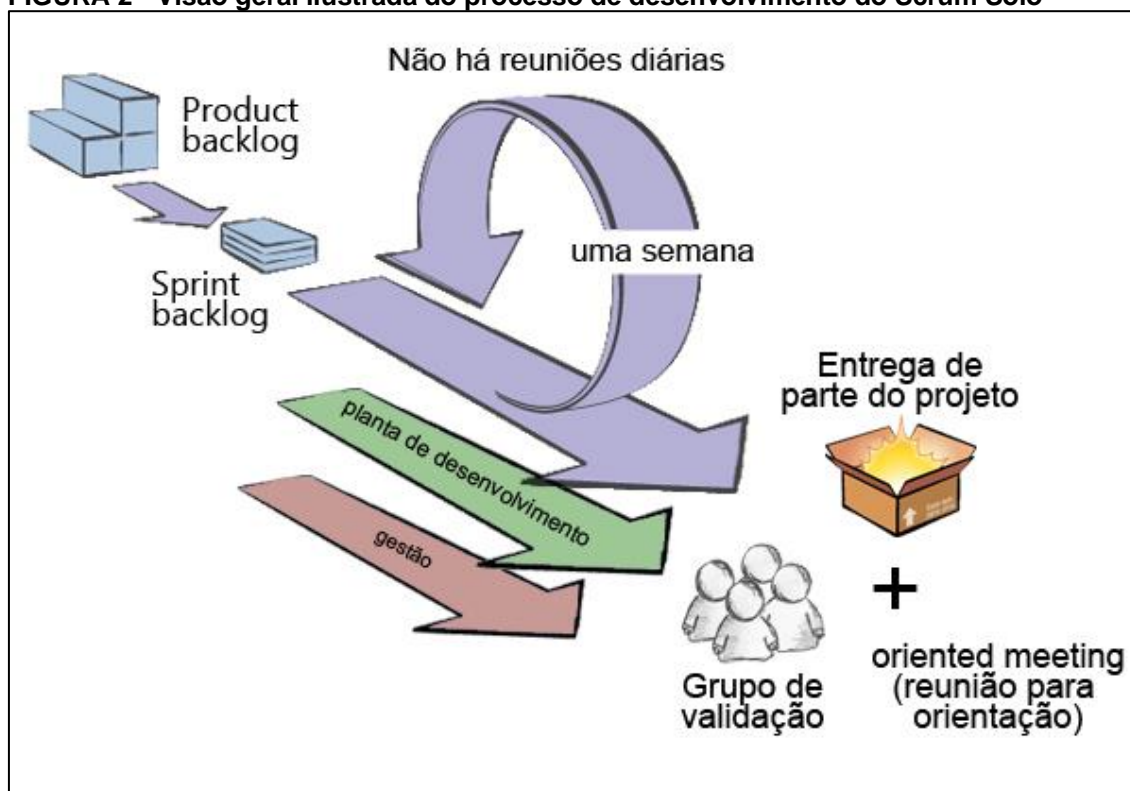
Fonte: Autoria Própria

3.8 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

O processo utilizado no desenvolvimento do software foi a adaptação da metodologia de desenvolvimento ágil *Scrum*. O processo *Scrum Solo* é iterativo e incremental, usa boas práticas delineadas pelo *Personal Software Process* (PSP) e pelo *Scrum*.

O *Scrum Solo* tem uma estrutura similar ao *Scrum* tradicional proposto por Schwaber e Sutherland. O formato *Solo* foi aplicado por alunos do curso de Engenharia da Computação e Análise de Desenvolvimento de Sistemas da UTFPR nos anos de 2012 a 2014 com sucesso no desenvolvimento, e contempla as necessidades de gestão de projetos (PAGOTTO *et al*, 2016), A visão geral do *Scrum Solo* é apresentada na Figura 2.

FIGURA 2 - Visão geral ilustrada do processo de desenvolvimento do Scrum Solo



Fonte: Adaptado de PAGOTTO *et al*, (2016).

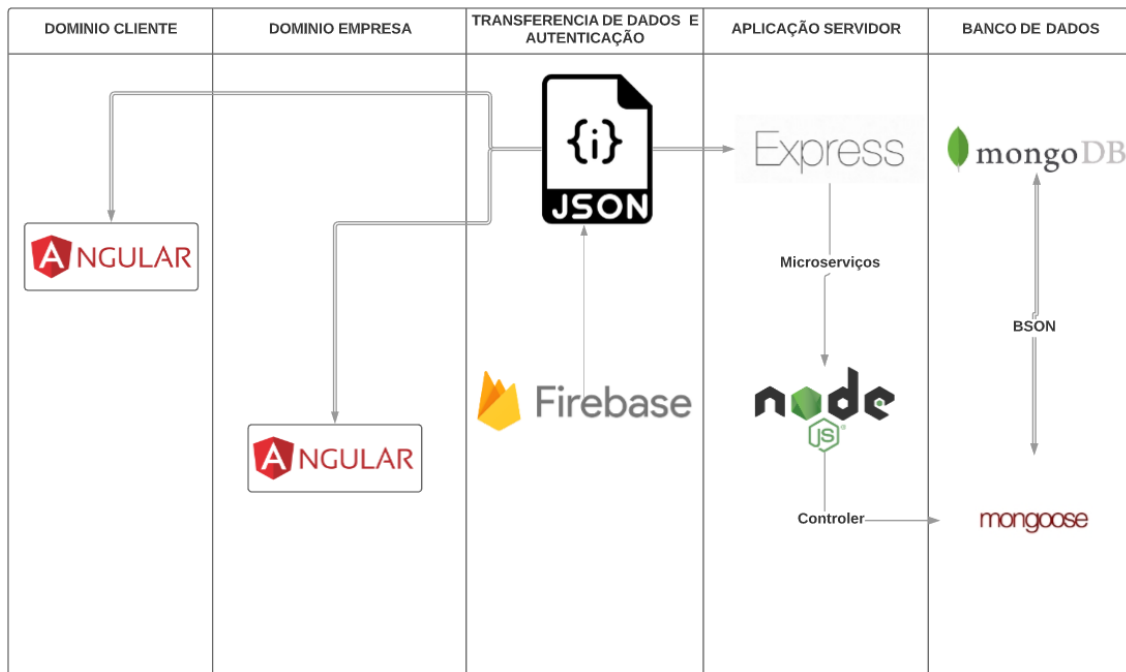
No desenvolvimento do projeto foi utilizando o adaptação do *Scrum Solo* utilizando parcialmente os objetivos de cada etapas do processo, durante a etapa do *Requirements*, foi realizada a elaboração do escopo do software, protótipos iniciais e *product backlog*, as *sprints* foram organizadas com duração de 7 dias, com a etapa de *Deployment* sendo realizada às segundas-feiras. Os itens relacionados a etapa de *Management* não foram realizados.

3.9 ARQUITETURA DO SOFTWARE

A arquitetura do software desenvolvido com suas interações entre tecnologias para aplicação servidor, serviços de autenticação e plataforma web, utilizando JSON, como comunicação entre seus módulos.

Na Figura 3 é apresentada a relação de ambiente e interação entre as principais tecnologias utilizadas na elaboração do projeto.

FIGURA 3 – Interação das tecnologias utilizadas



Fonte: Autoria Própria.

A aplicação do lado servidor é responsável por armazenar e/ou processar e fornecer dados para que outras aplicações e/ou sistemas. Essa aplicação contém as regras de negócio uma vez que dispõe de maior recurso computacional quando comparada com aplicações cliente, como páginas *web* requisitadas de notebooks de uso pessoal ou aplicativos móveis rodando em *smartphones* comuns.

3.10 BANCO DE DADOS

O banco de dados escolhido para o desenvolvimento do *software* foi o MongoDB como solução NoSQL. As Tecnologias NoSQL podem ser utilizadas para banco de dados que precisam de escalabilidade, desempenho e possuir tolerância a falhas, sendo recomendado para aplicações *web* (TOTH, 2011).

FIGURA 4 - Listagem de coleções e suas relações



Fonte: Autoria Própria.

O Quadro 7 apresenta a relação dos respectivos acessos e suas permissões na interação com o banco de dados entre inserir, ler, editar e atualizar, estas funções a serem implementadas e validadas pela aplicação do lado servidor,

utilizando de *token* de autenticação do protocolo OAuth para interação entre domínio empresa ou clientes com a aplicação servidor.

Quadro 7 - relação das permissões das *collections* com as respectivas aplicações

<i>collections</i>	domínio cliente	domínio empresa
categoria	ler	ler
empresa	ler	inserir, ler, editar
promocoos	ler	inserir, ler, editar
favoritos	ler, inserir, deletar	ler

Fonte: Autoria Própria.

4 DESENVOLVIMENTO DO SOFTWARE

O capítulo aborda como foi realizado o desenvolvimento da plataforma MEAN *Stack* Gastando Pouco, os aspectos técnicos e a implementação, seguindo as práticas descritas na etapa de *Sprints* do processo *Scrum Solo*.

4.1 **PRODUCT BACKLOG**

O *Product Backlog* contém uma lista de item, que possa ser associado com valor de negócio para a finalização do projeto (PEREIRA; TORREÃO; MARÇAL, 2007). O Quadro 8 apresenta a listagem dos *Product Backlog* da aplicação.

Quadro 8 - Listagem do *Product Backlog* da aplicação desenvolvida

Sprints	Funcionalidades
1ª	Ambiente de hospedagem e versionamento
	Estruturação da aplicação de lado servidor
	Estruturação do projeto na plataforma Firebase
	Layouts
	Início do projeto Angular <i>front-end</i> empresa
2ª	Início da estrutura da autenticação Firebase
	Desenvolvimento da tela inicial
	Tela de cadastro de usuário
	Tela de confirmação de e-mail
	Tela de preenchimento de informações da empresa
	Tela de login de usuário
3ª	Estrutura de rotas e segurança
	Implementação de rotas do menu
	Logout
	Tela de cadastro de promoção
	Tela de listagem de promoções
	Tela de editar perfil
4ª	Início desenvolvimento aplicação do domínio cliente
	Autenticação Firebase
	Tela de login redes sociais
	Autenticação Facebook
	Autenticação Google
	Permissão de localização do usuário

Sprints	Funcionalidades
5ª	Arquitetura do layout no domínio cliente
	Parâmetros da listagem
	Card de promoções da empresa
	Tela listagem de promoções
	Tela de listagem das informações da empresa
	Tela de listagem de promoções favoritas

Fonte: Autoria Própria

4.1.1 SPRINT 1

O Período de trabalho foi de 7 dias, entre os dias 03/08/2019 e 09/08/2019 com o desenvolvimento das atividades listadas a seguir:

4.1.1.1 AMBIENTE DE HOSPEDAGEM E VERSIONAMENTO

Para a hospedagem da aplicação do lado servidor foi escolhido o serviço gratuito do conhecido como Heroku, conhecido pelos serviços de nuvem PaaS (*Plataform as a service*), oportunidade de estudantes e pesquisadores usarem de forma gratuita (MIDDLETON; SCHNEEMAN, 2013).

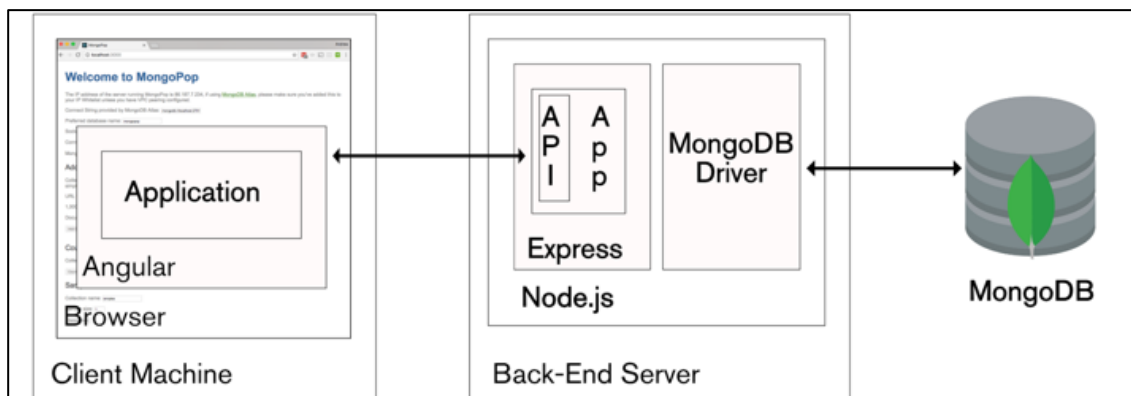
Para o versionamento da aplicação foi utilizado o serviço GITHUB, para colaboração de projetos de código aberto ou privados (DABBISH *et al*, 2012).

4.1.1.2 ESTRUTURAÇÃO DO APLICAÇÃO DE LADO SERVIDOR

A Aplicação do lado servidor foi desenvolvida utilizando as tecnologias NodeJS, ExpressJS. O *TypeScript* foi utilizado como linguagem de escrita do código e as ferramentas, Gulp e Mongoose para agilizar o desenvolvimento.

Na Figura 5 apresentamos a estrutura da relação orgânica entre as tecnologias com a aplicação do lado servidor.

FIGURA 5 – Relação das tecnologias com aplicação do lado servidor



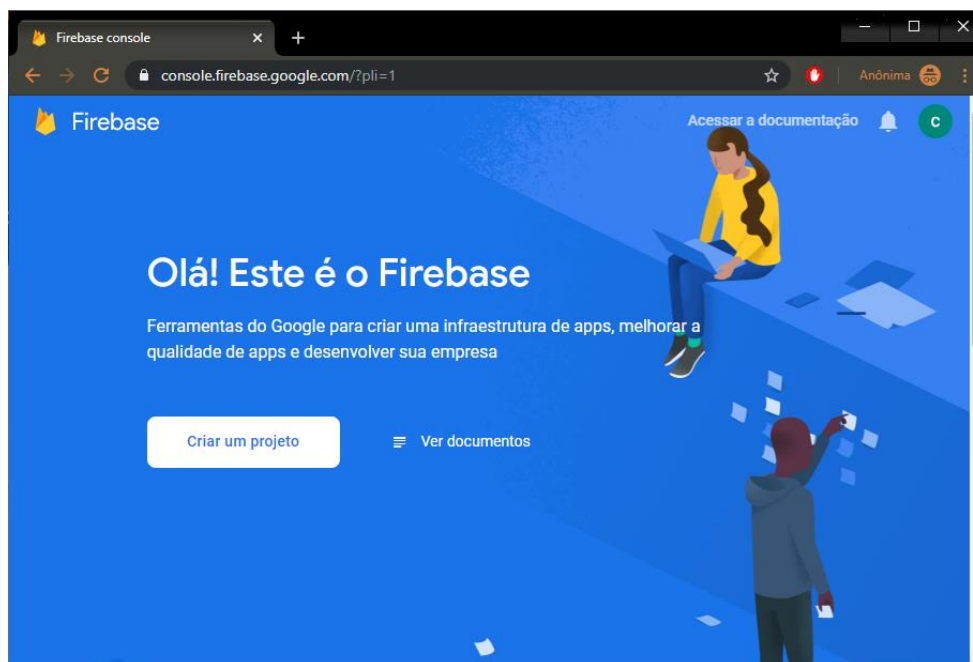
Fonte: Adaptado de (MORGAN, 2017)

4.1.1.3 ESTRUTURAÇÃO DO PROJETO NA PLATAFORMA FIREBASE

A criação do projeto do Firebase e estruturação serviram para gerar as credenciais, para o acesso aos serviços de autenticação com redes sociais, e-mail e senha, hospedagem de páginas estáticas e *upload* de arquivos.

Segundo (MORONEY, Laurence; MORONEY; ANGLIN, 2017) o Firebase disponibiliza um conjunto de ferramentas ideais para o desenvolvimento ágil de grande performance de aplicações multiplataforma, com serviços relacionados a teste, monetização, notificação, autenticação e banco de dados, entre outros.

FIGURA 6 - Página inicial Firebase



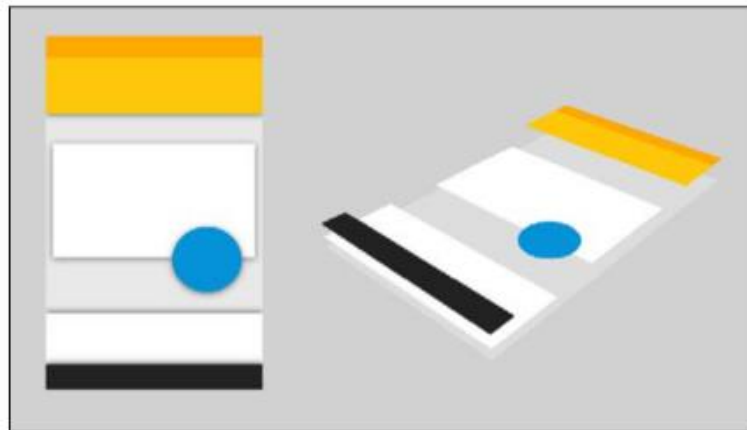
Fonte: Autoria Própria

4.1.1.4 LAYOUTS

Para a criação dos *layouts* foi utilizado o software Adobe XD para o *design* de experiência do usuário seguindo a filosofia do Material Design.

Segundo Mew (2015) O Material Design permite uma ilusão de uma terceira dimensão a interação do usuário com a aplicação, às ações do usuário retornam uma interação visual, animações ou troca de cores.

FIGURA 7 - Terceira dimensão Material Design

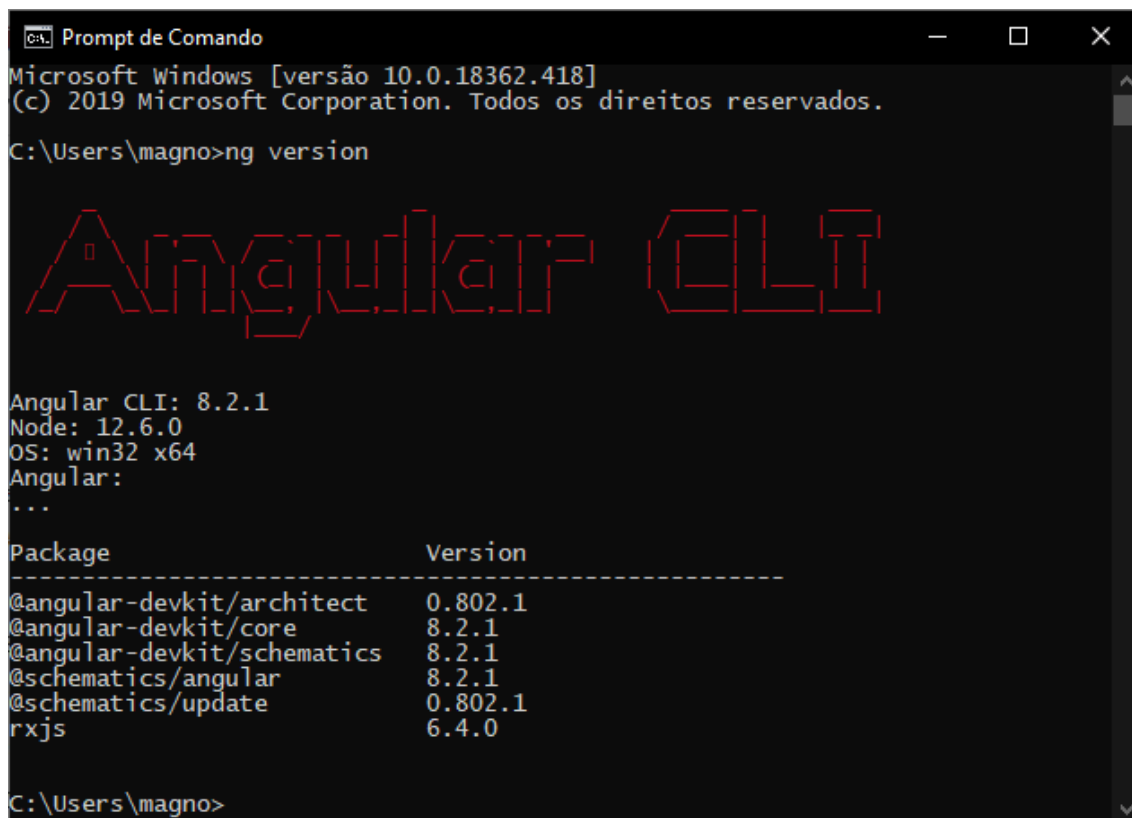


Fonte: Adaptado de (MEW, 2015)

4.1.1.5 INÍCIO DO PROJETO ANGULAR FRONT-END EMPRESA

A criação do projeto Angular do domínio empresa exige a instalação no ambiente de desenvolvimento o NodeJS e Angular CLI. A instalação pode ser observada conforme a Figura 8, com a execução da verificação de versão instalada, com o comando via terminal *“ng version”*.

FIGURA 8- Comando para verificar Angular Version



```
C:\Users\magno>ng version

Angular CLI: 8.2.1
Node: 12.6.0
OS: win32 x64
Angular:
...
Package                                Version
-----
@angular-devkit/architect              0.802.1
@angular-devkit/core                   8.2.1
@angular-devkit/schematics             8.2.1
@schematics/angular                   8.2.1
@schematics/update                     0.802.1
rxjs                                   6.4.0
```

Fonte: Autoria Própria

4.1.2 SPRINT 2

O período trabalho foi de 7 dias, entre os dias 10/08/2019 a 16/08/2019 com o desenvolvimento das atividades listadas, a seguir:

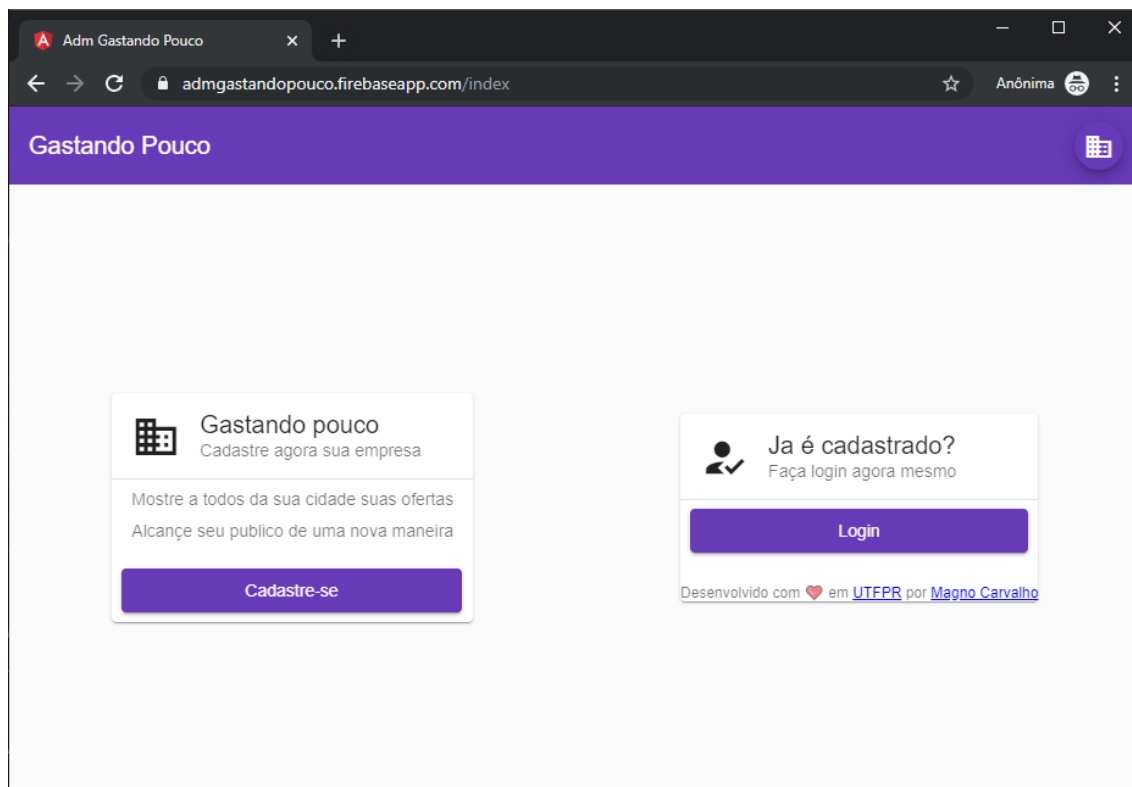
4.1.2.1 INÍCIO DA ESTRUTURA DA AUTENTICAÇÃO FIREBASE

A estruturação da interação entre autenticação realizado pelo serviço do Firebase e a domínio empresa, implementando a biblioteca do Firebase.

4.1.2.2 DESENVOLVIMENTO DA TELA INICIAL

A tela inicial da aplicação foi a primeira a ser criada, contém 2 rotas distintas, implementada de forma responsiva. A Figura 9 e Figura 10, ao serem comparadas permite visualizar a página adaptada para diferentes resoluções e dispositivos.

FIGURA 9 - Página inicial domínio empresa resolução 960x600



Fonte: Autoria Própria

Na Figura 13 e Figura 14 é possível observar os mesmos componentes organizados de formas a se adaptarem a suas respectivas resoluções.

FIGURA 10 - Página inicial domínio empresa resolução iPhone 5/SE 320x568



Fonte: Autoria Própria

Na tela inicial da aplicação o usuário pode seguir dois fluxos distintos, ao clicar nos botões relacionado ao *login*, direcionado para a tela de realizar o *login* da aplicação ao clicar nos botões relacionados ao cadastro direcionado para a página responsável pelo cadastro de novos usuários.

4.1.2.3 TELA DE CADASTRO DE USUÁRIO

A tela para o início do cadastro das informações relacionadas a autenticação do usuário utiliza o serviço de autenticação do Firebase. O usuário obrigatoriamente deve informar um e-mail válido, senha, foto de perfil e nome da empresa.

Uma vez que as informações fornecidas são salvas junto ao serviço do Firebase, utiliza uma requisição do tipo post, no caso de sucesso na criação do usuário, a biblioteca do Firebase, válida e autêntica o novo usuário no *cache* do navegador (AGARWAL, 2018).

FIGURA 11 - Página de cadastro de nova empresa

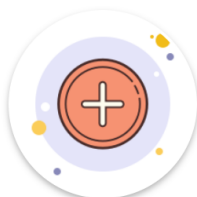


Cadastro empresa

Crie seu usuário para o acesso

Mostre a todos da sua cidade suas ofertas

Alcance seu publico de uma nova maneira



Adicionar Logo da Empresa

Empresa *



Email *



Senha *



 Registrar

ou

[Ja possui uma conta ?](#)

Desenvolvido com  em [UTEPR](#) por [Magno Carvalho](#)



Privacidade - Termos

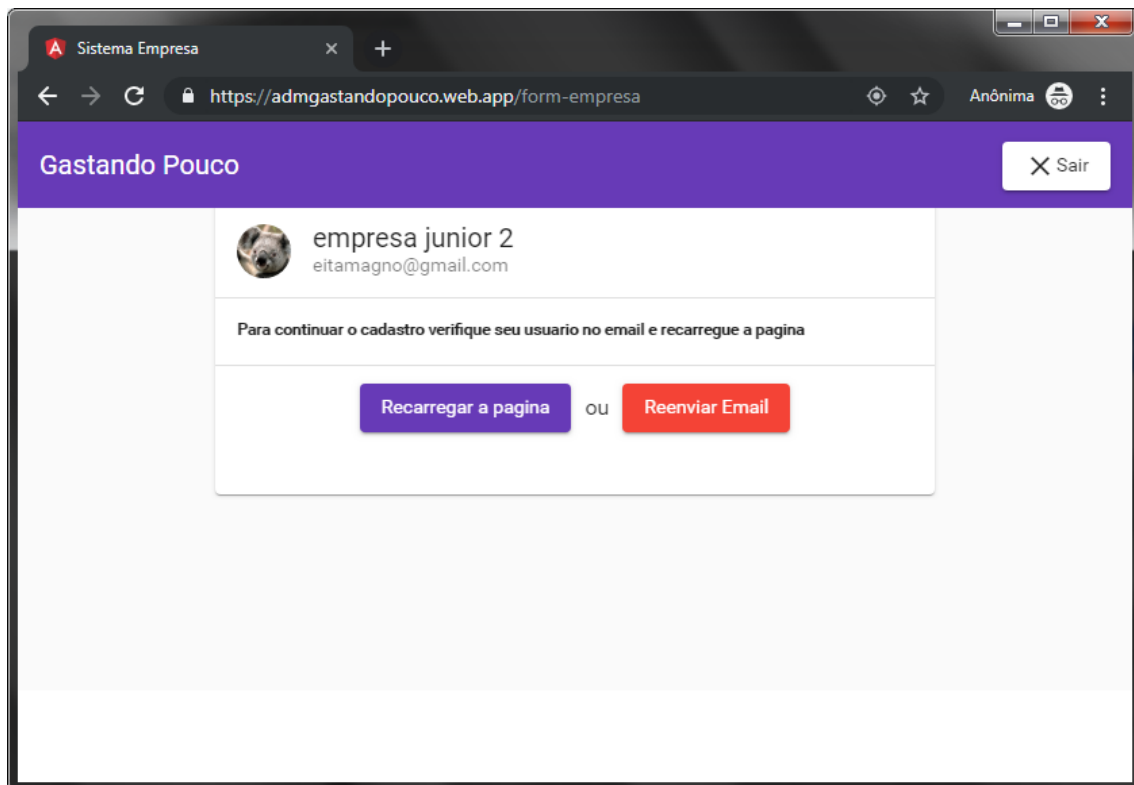
Fonte: Autoria Própria

Na tela de cadastro foi implementado um formulário com validação adicional que utiliza a tecnologia de reCAPTCHA v2 do Google, segundo Beede (2010), o CAPTCHA serve para distinguir se a entrada dos dados no formulário é proveniente de um ser humano ou de um computador. Seu principal objetivo é ajudar a impedir ou reduzir a ação de um computador automatizado.

4.1.2.4 TELA DE CONFIRMAÇÃO DE EMAIL

Na tela de cadastro anterior, quando um usuário se inscreve usando o endereço de e-mail, envia uma mensagem de confirmação para verificar o endereço de e-mail fornecido, na tela de confirmação de e-mail. Assim, cria uma autenticação de dois fatores com o serviço de envio de confirmação de e-mail do Firebase. A continuação do cadastro só é permitida após a confirmação do e-mail.

FIGURA 12 - Página de confirmação de email da empresa nova empresa



Fonte: Autoria Própria

4.1.2.5 TELA DE PREENCHIMENTO DE INFORMAÇÕES DA EMPRESA

Após a validação do e-mail informado no cadastro da empresa, o usuário é encaminhado para a tela onde o mesmo informa o endereço, telefone, descrição, informações sobre a categoria do estabelecimento e número de inscrição do CNPJ da empresa.

No lado servidor foi criado todos arquivos e métodos relacionados com o cadastro empresa, No Quadro 9 temos a representação do campo, tipo de dado salvo, obrigatoriedade e descrição objetiva de cada dado salvo.

Quadro 9 - Listagem de itens do *schema* do objeto Empresa

Campo	Tipo	Obrigatório	Descrição
_id	<i>ObjectID</i>	sim	Identificação única em formato de <i>hexadecimal String</i>
uid	<i>String</i>	sim	Identificação única gerada pelo Firebase
bairro	<i>String</i>	sim	Bairro da empresa
cep	<i>String</i>	sim	Cep da empresa
cidade	<i>String</i>	sim	Cidade da empresa
complemento	<i>String</i>	não	Complemento do endereço da empresa
estado	<i>String</i>	sim	Estado do endereço da empresa
googlePlace	<i>String</i>	sim	String com o endereço concatenado da empresa
número	<i>String</i>	sim	Número do endereço da empresa
pais	<i>String</i>	sim	País onde está localizado a empresa
rua	<i>String</i>	sim	Logradouro da Localização da empresa
location	<i>GeoJSON</i>	sim	Conjunto de informações da localização das coordenadas da empresa para calcular a geometria sobre uma esfera parecida com a Terra.
tipo	<i>ObjectID</i>	sim	<i>Hexadecimal String</i> referência da categoria da empresa
cnpj	<i>String</i>	sim	CNPJ da empresa com o número que identifica a empresa.
telefone	<i>String</i>	sim	Telefone a ser informado pela empresa
description	<i>String</i>	não	Descrição das atividades e história da empresa

Fonte: Autoria Própria

4.1.2.6 TELA DE LOGIN DE USUÁRIO

A tela de *login* de usuário empresa estão a entrada de dados para e-mail e senha, botão de interação para ação de fazer a autenticação na plataforma e botão para navegar a página de criar usuário.

O e-mail é verificado por um conjunto de métodos da biblioteca de validação de formulários do “@angular/form”. A senha foi definida para ter o mínimo 6 caracteres.

Segundo Agarwal (2018) ao efetuar a requisição de autenticação o usuário enviar a informação de e-mail e senha ao Firebase, ele utiliza uma conexão segura de chaves assimétricas, fornecidas durante o processo de cadastro do projeto no Firebase, com o sucesso da autenticação temos as informações do usuário, entre elas, UID e o Token. A Figura 13 apresenta a utilização da biblioteca do Firebase na autenticação de usuário.

FIGURA 13 - Estrutura do método de autenticação

```
firebase.auth()
  .signInWithEmailAndPassword(email: String, password: String)
  .then((res: firebase.auth.UserCredential) => {
    let uid = res.user.uid;
    let token = res.user.getIdToken();
    // UID e Token são usados como os parâmetros para
    //trazer as informações da empresa cadastradas
  }).catch(function (error) {
    //Implementa o retorno do erro ao usuário
  });
```

Fonte: Autoria Própria

O *Token* gerado pela autenticação é adicionado ao *header* de todas as requisições (GET, POST, PUT e DELETE) da plataforma, durante o tempo que o usuário estiver autenticado, uma chave pública da requisição aplicação do lado servidor (MORONEY, Laurence; MORONEY; ANGLIN, 2017). A Figura 18 apresenta a tela de *login*.

FIGURA 14 - Página de cadastro de nova empresa

Adm Gastando Pouco

admgastandopouco.web.app/login

Anônima

Gastando Pouco

Ja é cadastrado?
Faça login agora mesmo

Email *

Senha *

Login

ou

Não possui uma conta ?

Desenvolvido com ❤ em UTFPR por [Maígo Carvalho](#)

Fonte: Autoria Própria

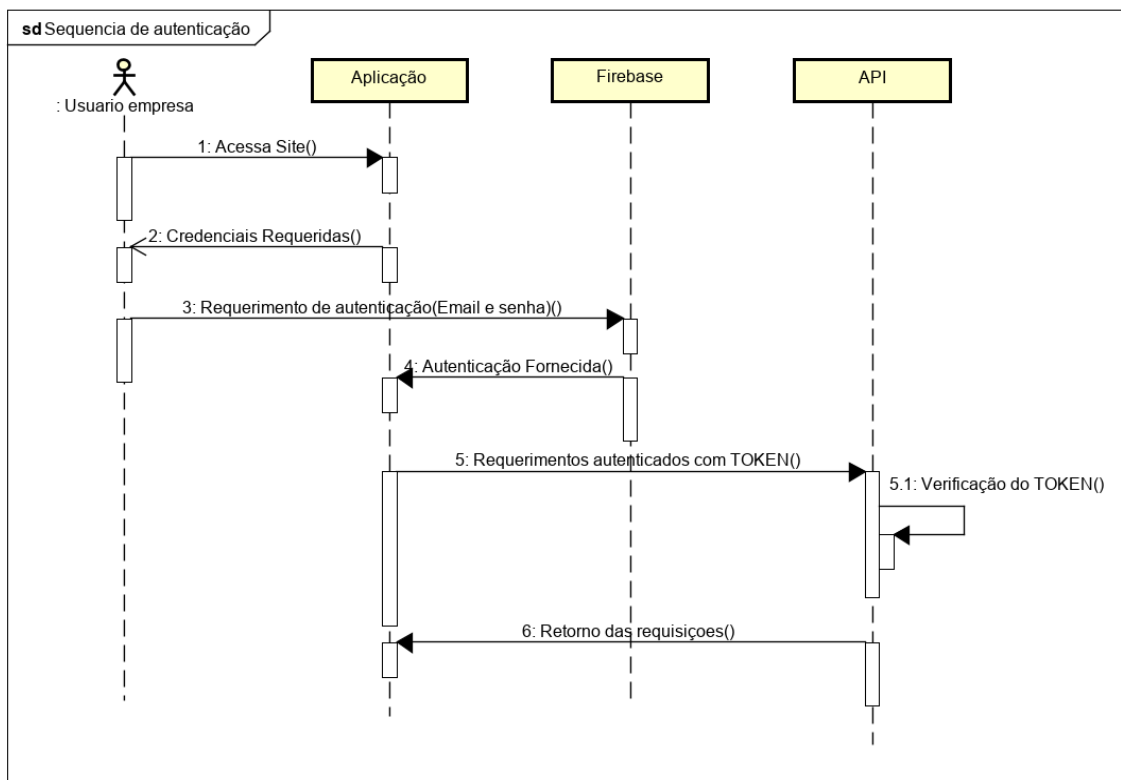
4.1.3 SPRINT 3

O período de trabalho foi de 7 dias, entre os dias 17/08/2019 a 23/08/2019 com o desenvolvimento das atividades listadas a seguir:

4.1.3.1 ESTRUTURA DE ROTAS E SEGURANÇA

A aplicação usa autenticação com o *token* gerado pelo Firebase, no formato JWT, acrônimo de JSON Web Token. O JWT define o que é enviado pelo cabeçalho HTTP contém as informações do usuário criptografadas na transmissão de informações entre cliente e servidor de forma segura, usando um objeto JSON, salvo no *cache* do dispositivo do usuário, com par de chaves públicas e privadas, garante a confiabilidade (JONES *et al*, 2012). A Figura 15 apresenta o diagrama com a sequência da autenticação.

FIGURA 15 - Diagrama de sequência da autenticação da aplicação



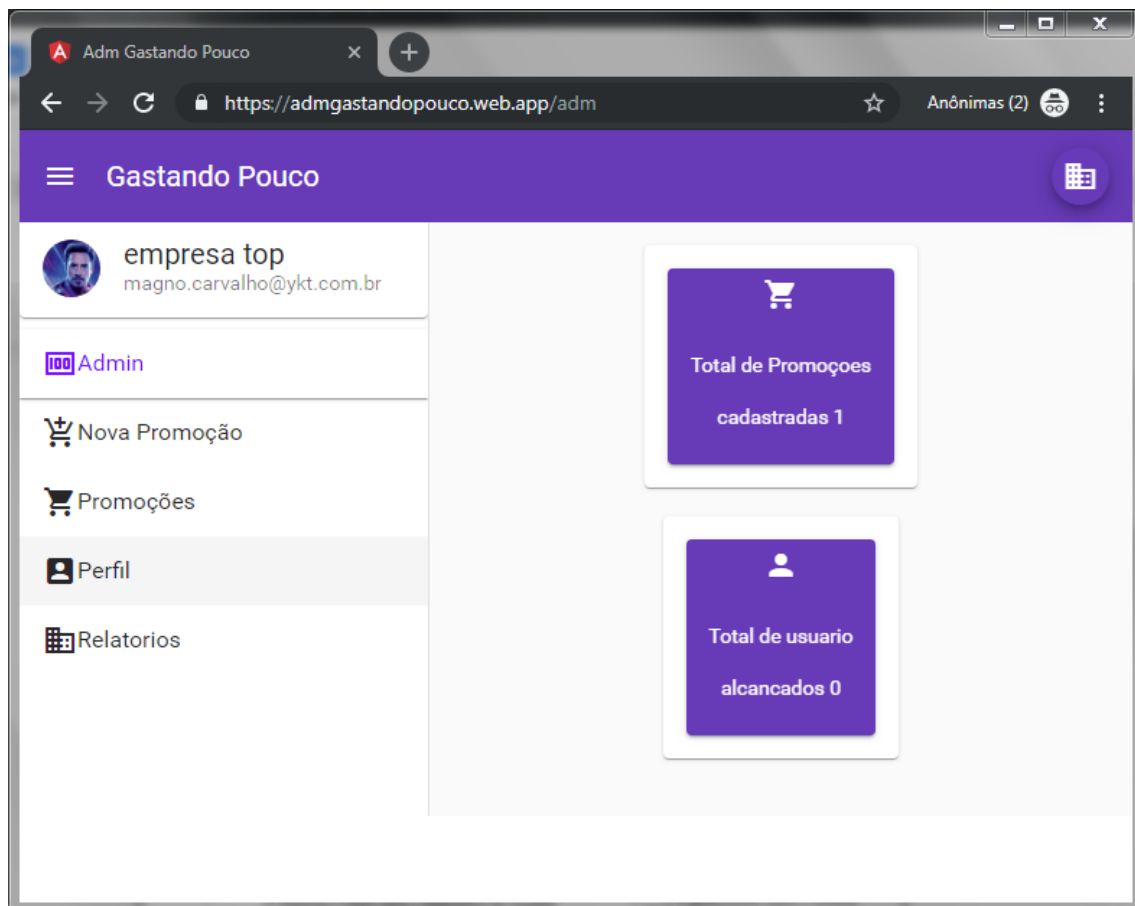
Fonte: Autoria Própria

Na aplicação *web* é verificado se usuário está autenticado a cada navegação ou troca de tela dentro da aplicação.

4.1.3.2 IMPLEMENTAÇÃO DE ROTAS DO MENU

Foi implementado um componente na lateral esquerda como menu de com opções da aplicação apresentada na Figura 16.

FIGURA 16 - Menu lateral dimensões 960x600



Fonte: Autoria Própria

Temos a informação do nome da empresa e e-mail, sempre listado no menu lateral e as seguintes opções:

- Admin contém um relatório simples de promoções cadastradas e clientes alcançados.
- Nova promoção contém um formulário para adição de nova promoção
- Promoções contém a listagem de promoções cadastradas pela empresa, ativas e desativadas.
- Relatórios contém relatórios de acesso a promoção e visualização.

4.1.3.3 LOGOUT

A opção de *logout* da aplicação é disposta no *Topbar* da aplicação, acessada a qualquer momento pelo usuário, chamando o método que limpa o *cache* no dispositivo do usuário todas as informações relacionadas, ao final redireciona para a página inicial da aplicação.

4.1.3.4 TELA DE CADASTRO DE PROMOÇÕES

O formulário de cadastro de promoções é utilizado para adicionar novas promoções da empresa ao banco de dados.

A Figura 17 apresenta o formulário de cadastro de promoções.

FIGURA 17 - Formulário de cadastro de promoções

A imagem mostra a interface de um navegador com o endereço <https://admgastandopouco.web.app/new-promo>. O cabeçalho da página é roxo e contém o nome 'Gastando Pouco' e um ícone de menu. À esquerda, há um menu lateral com opções: 'Admin', 'Nova Promoção' (destacada), 'Promoções', 'Perfil' e 'Relatorios'. O formulário principal, em branco, contém os seguintes campos: 'Titulo *' (campo de texto), 'Descrição *' (campo de texto com contador '0 / 300'), 'Data inicial *' e 'Data Final *' (campos de data), 'Valor anterior *' e 'Valor Com desconto *' (campos de valor monetário com ícone de dólar e 'R\$ 0,00'). Abaixo desses campos, há um botão 'Salvar' e uma mensagem de erro 'Porcentagem Invalida'. Um placeholder para uma imagem de 500x500 pixels é exibido no lado esquerdo do formulário, com a legenda 'Thumbnail obrigatório*'.

Fonte: Autoria Própria

O cadastro da promoção pela empresa, permite preencher um formulário com as informações apresentadas no Quadro 10.

Quadro 10 - Campos do formulário de cadastro de promoção.

Campo	Tipo	Obrigatório	Descrição
Título	<i>String</i>	sim	Título da promoção, com máximo de 100 caracteres
Descrição	<i>String</i>	sim	Descrição detalhada da Promoção com o máximo de 300 caracteres
Data Inicial	<i>Date</i>	sim	Data inicial da vigência da promoção
Data Final	<i>Date</i>	sim	Data final da vigência da promoção

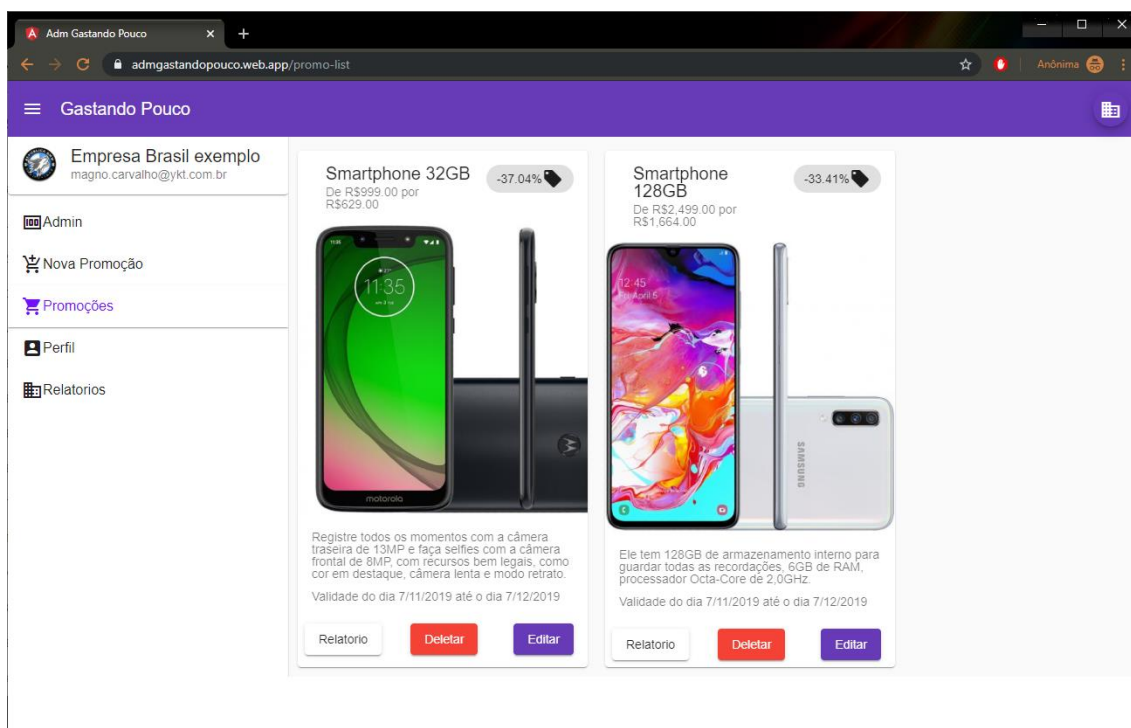
Campo	Tipo	Obrigatório	Descrição
Valor anterior	Double	sim	Valor anterior ao desconto
Valor com desconto	Double	sim	Valor com desconto
Thumbnail	String	sim	Foto da promoção
Porcentagem	Number	sim	Porcentagem de desconto calculada a partir dos valores informados

Fonte: Autoria Própria

4.1.3.5 TELA DE LISTAGEM DE PROMOÇÕES

A tela de listagem de promoções, Figura 18, lista os produtos cadastrados pela empresa, com o período de validade da promoção, estado ativo, ordenados pela data da criação do produto, com as opções de editar e deletar as ofertas já cadastradas.

FIGURA 18 - Formulário de cadastro de promoções



Fonte: Autoria Própria

4.1.3.6 TELA DE EDITAR PERFIL

A tela de editar perfil, possui a listagem das informações cadastradas pela empresa e a permite editar as informações exceto o e-mail do cadastro da empresa, apresentado na Figura 19.

FIGURA 19 – Tela de editar perfil

A interface da tela de edição de perfil é dividida em três seções principais:

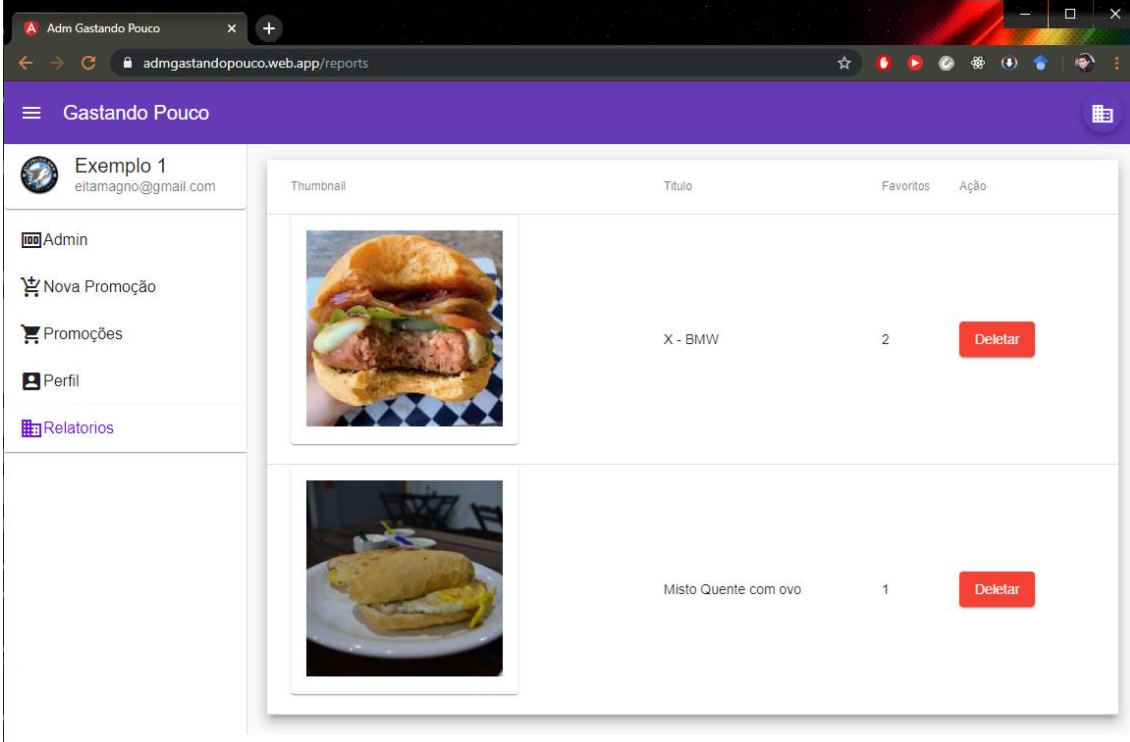
- Barra Superior:** Exibe o nome do usuário "Gastando Pouco" e o ícone de uma calculadora.
- Menu Lateral (Esquerda):** Contém links para "Admin", "Nova Promoção", "Promoções", "Perfil" (destacado em roxo) e "Relatorios".
- Área Principal:**
 - Cabeçalho:** Mostra o e-mail "magno.carvalho@ykt.com.br" e o nome "Exemplo 2" com o e-mail "magno.carvalho@ykt.com.br".
 - Imagem de Perfil:** Um círculo contendo uma ilustração de pessoas dançando.
 - Formulário de Dados:**
 - Empresa:** Campo com o valor "Exemplo 2".
 - Mapa:** Uma visualização do Google Maps com um marcador vermelho no centro de Cornélio Procyópio, PR. Abaixo do mapa, o endereço completo é exibido: "R. Tiradentes, 501 - Centro, Cornélio Procyópio - PR, 86300-000, Brasil".
 - Rua:** Campo com o valor "Rua Tiradentes".
 - Numero:** Campo com o valor "501".
 - Complemento:** Campo com o valor "centro".
 - CEP:** Campo com o valor "86300-000".
 - Bairro:** Campo com o valor "Centro".
 - Cidade:** Campo vazio.
 - Estado:** Campo com o valor "Paraná".
 - Pais:** Campo com o valor "Brasil".
 - CNPJ:** Campo com o valor "43.212.544/2342-45".
 - Telefone:** Campo com o valor "(43) 99999-9999".
 - Descrição:** Campo com o valor "Empresa de entretenimento noturno".
 - Tipo de estabelecimento:** Menu suspenso com o valor selecionado "Vida Noturna/ Bar / Pub".
 - Botão:** Um botão roxo na base do formulário com o texto "Atualizar Perfil".



Fonte: Autoria Própria

4.1.3.7 TELA DE LISTAGEM DE RELATÓRIOS

A tela de listagem de relatórios, Figura 20, lista todas promoções já cadastradas pelo cliente em tabela com a informações de nome, imagem e quantidade de clientes tornaram a promoção favorita.

FIGURA 20 – Listagem de relatórios de favoritos



Thumbnail	Título	Favoritos	Ação
	X - BMW	2	<button>Deletar</button>
	Misto Quente com ovo	1	<button>Deletar</button>

Fonte: Autoria Própria

4.1.4 SPRINT 4

O período de trabalho foi de 7 dias, entre os dias 24/08/2019 a 30/08/2019 com o desenvolvimento das atividades listadas a seguir:

4.1.4.1 INÍCIO DESENVOLVIMENTO DO DOMÍNIO CLIENTE

A nova aplicação Angular foi criada para o domínio de acesso dos clientes, utiliza base já desenvolvida para o projeto empresa durante a *Sprint 1*.

4.1.4.2 AUTENTICAÇÃO FIREBASE

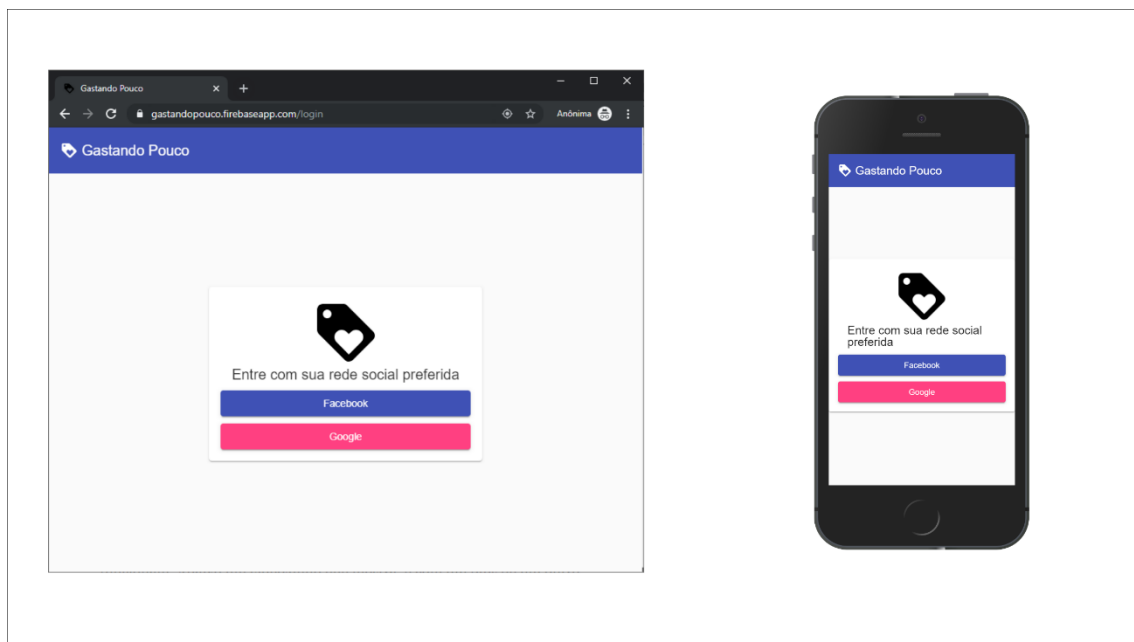
Para facilitar a experiência do cliente na listagem de promoções é obrigatório a autenticação do cliente. A seguir foi implementada autenticação com rede sociais Facebook e Google, utilizando o Firebase, a exemplo da aplicação empresa.

4.1.4.3 TELA DE *LOGIN* REDES SOCIAIS

A tela de *login* consiste em *mat-card* centralizado com os botões para início da autenticação com as redes sociais Google e Facebook, Figura 21 apresenta a tela de *login*, utiliza o padrão OAuth ou *Open Authorization* segundo Xavier e Carvalho (2011).

Sua especificação consiste em duas partes. A primeira parte define um processo no navegador, para redirecionamento do usuário final para autorização aos seus recursos, pelo cliente. A segunda parte define um método para realização de requisições HTTP autenticadas usando dois conjuntos de credenciais. Um conjunto destinado à identificação do cliente e outro à identificação do dono do recurso a ser requisitado.

FIGURA 21 – Tela de autenticação domínio cliente em diferentes resoluções.



Fonte: Autoria Própria

A exemplo do domínio empresa, todas as telas do domínio cliente foram criadas para se adaptarem a todos os tamanhos de dispositivos.

4.1.4.4 PERMISSÃO DE LOCALIZAÇÃO DO USUÁRIO

O método para solicitar a localização utiliza HTML *Geolocation* API fornecendo as coordenadas do usuário. A localização é mais precisa para dispositivos com GPS, como smartphones.

O HTML *Geolocation* API é limitado a alguns navegadores que implementa essa função sendo necessário que usuário conceda a permissão aprovando o uso

da localização geográfica do seu dispositivo (PEJIĆ, Bojan; PEJIĆ; ČOVIĆ,2010), O Quadro 11 temos a relação de navegadores compatíveis com HTML *Geolocation* API.

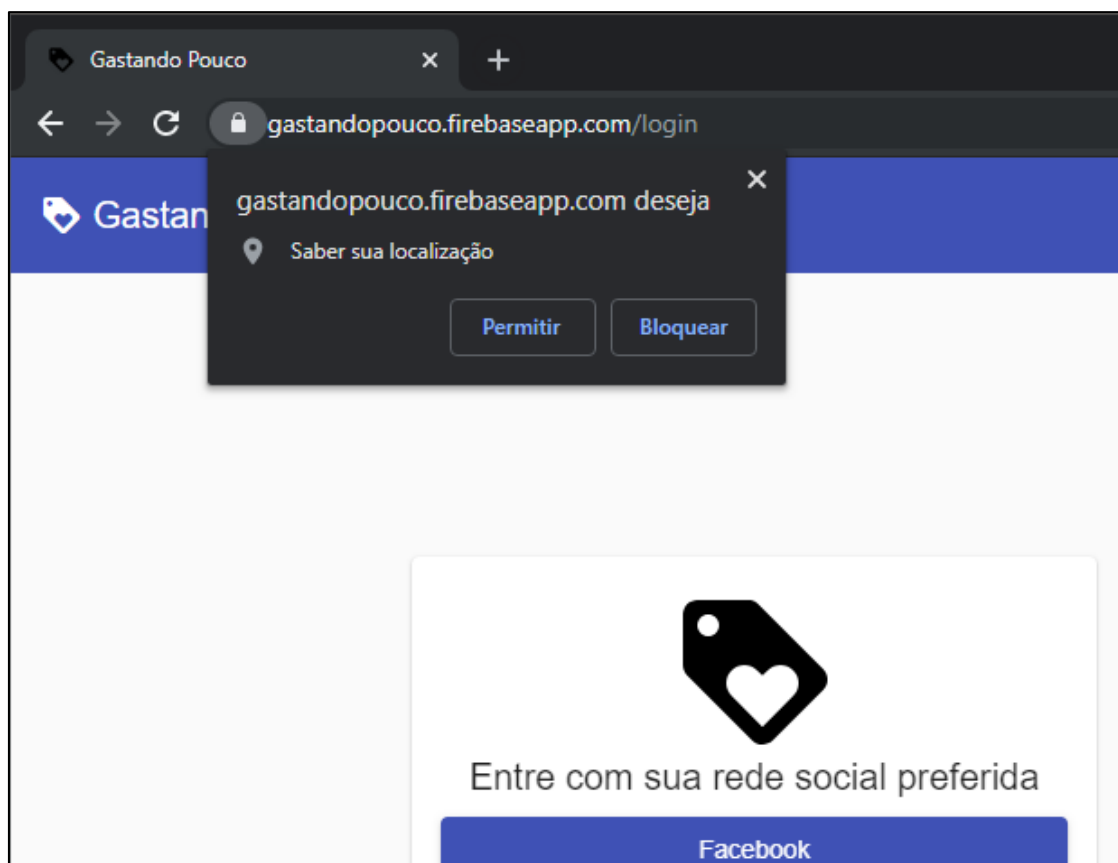
Quadro 11 – navegadores compatíveis com HTML *Geolocation* API: *getCurrentPosition*

Navegadores
Android Browser*
Chrome
Chrome for Android
Edge*
Firefox
Firefox for Android
IE
iOS Safari*
KaiOS Browser
Opera
Opera Mobile*
QQ Browser
Safari
Samsung Internet
UC Browser for Android

Fonte: Adaptado de (CANIUSE, 2019)

O método faz a pergunta ao usuário se pode fornecer as informações da sua localização que serão usadas na busca de promoções a partir da localização do mesmo apresentada na Figura 22.

FIGURA 22 – Permissão de usar a localização do usuário



Fonte: Autoria Própria

O usuário deve fornecer a localização para ter acesso a plataforma.

4.1.4.5 AUTENTICAÇÃO FACEBOOK

O botão de autenticação com Facebook abre uma conexão com o serviço do Facebook conhecido pelo nome *Graph API*, permite requisitar a autenticação do usuário, cria as credenciais e configurando junto ao serviço do Firebase.

4.1.4.6 AUTENTICAÇÃO GOOGLE

O botão de autenticação com Google faz autenticação de usuário com o serviço do Google de Auth dentro do Firebase.

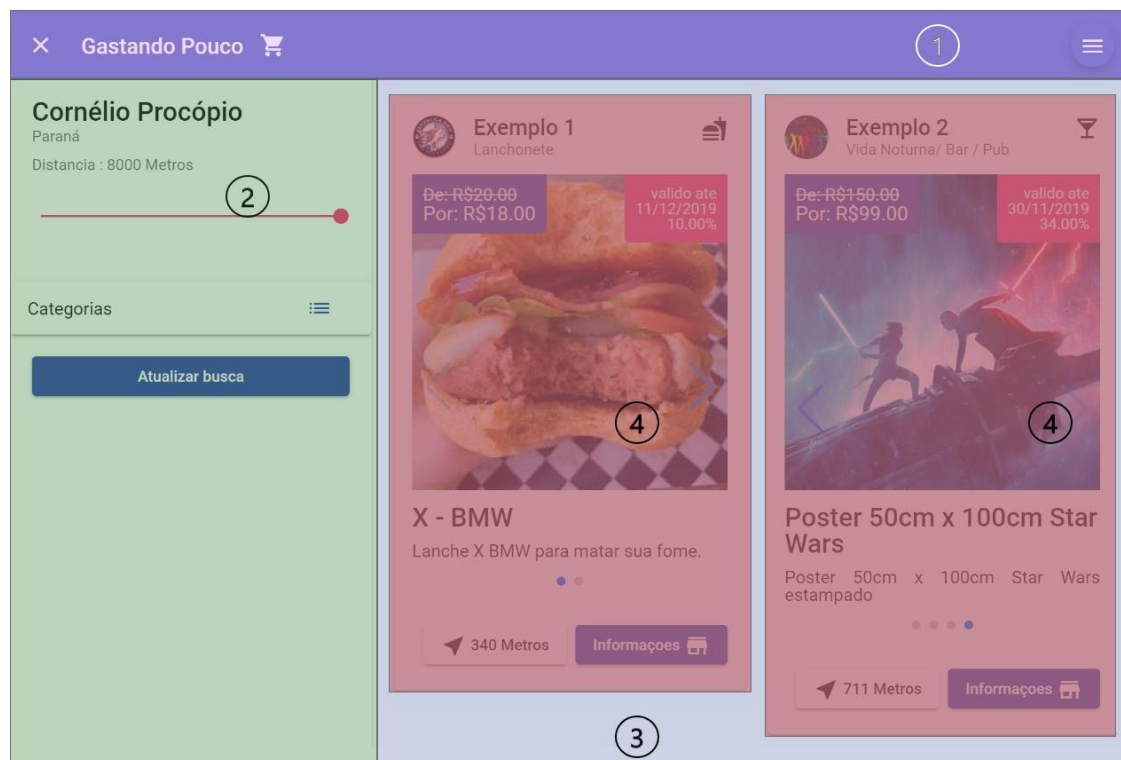
4.1.5 SPRINT 5

O período de trabalho foi de 7 dias, entre os dias 31/08/2019 a 06/09/2019, com o desenvolvimento das atividades listadas a seguir:

4.1.5.1 ARQUITETURA DO LAYOUT NO DOMÍNIO CLIENTE

A organização do *layout* no domínio cliente é apresentada na Figura 23.

FIGURA 23 – Organização do layout do domínio cliente



Fonte: Autoria Própria

O domínio cliente e empresa foram implementados utilizando a mesma base de *layout*, a estrutura é baseada no *layout* Material *Sidenav Container*, toda a navegação ocorre no componente do Angular “*router-outlet*”, os demais componentes, não atualizam quando acontece a navegação dentro da aplicação, a listagem de referência a os números na Figura 23 correspondem a os seguintes componentes do Angular Material e Angular Router:

1. Mat-Toolbar
2. Mat-Sidenav-Container
3. Router-Outlet
4. Mat-Card

4.1.5.2 PARÂMETROS DA LISTAGEM

Os parâmetros da listagem são implementados dentro do componente de *Sidenav* com as opções para listagem de promoções, *slide* com a possibilidade de editar a distância como parâmetro de busca e listagem de categorias de empresas,

indica a cidade onde o usuário está localizado utiliza o método de *reverse geocode* do serviço do Google Maps.

4.1.5.3 CARD DE PROMOÇÕES DA EMPRESA

A implementação do *Card* com resumo das informações da empresa com *slide* contém a imagem e descrição das promoções cadastradas.

FIGURA 24 – Card de listagem de promoções



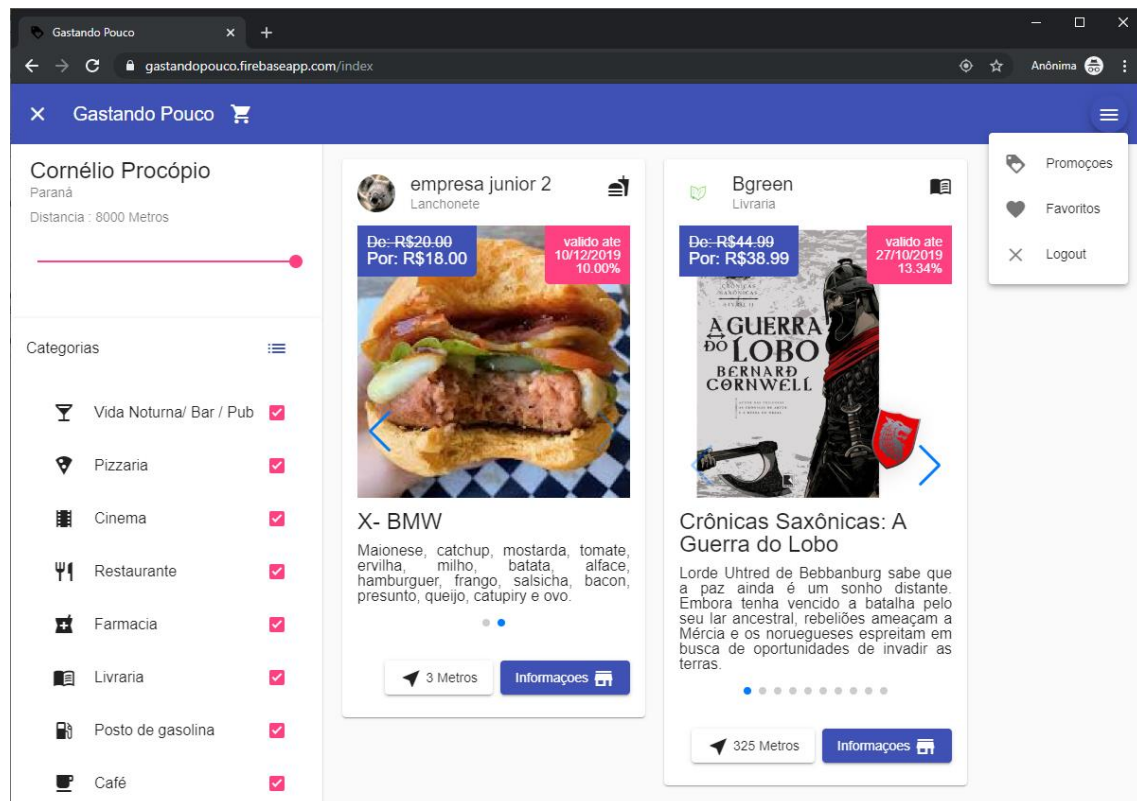
Fonte: Autoria Própria

A Figura 24 contém o botão com a distância entre o usuário e a empresa, a opção de abrir a localização no mapa do Google, com o botão de informações abre a tela com as informações da empresa e listas as promoções apenas da empresa.

4.1.5.4 TELA LISTAGEM DE PROMOÇÕES

A listagem de promoções, Figura 25, utilizando os parâmetros de localização e categorias ordenados pela proximidade da localização do usuário e organizados de forma a se adaptarem em diferentes resoluções de dispositivos.

FIGURA 25 – Listagem de promoções

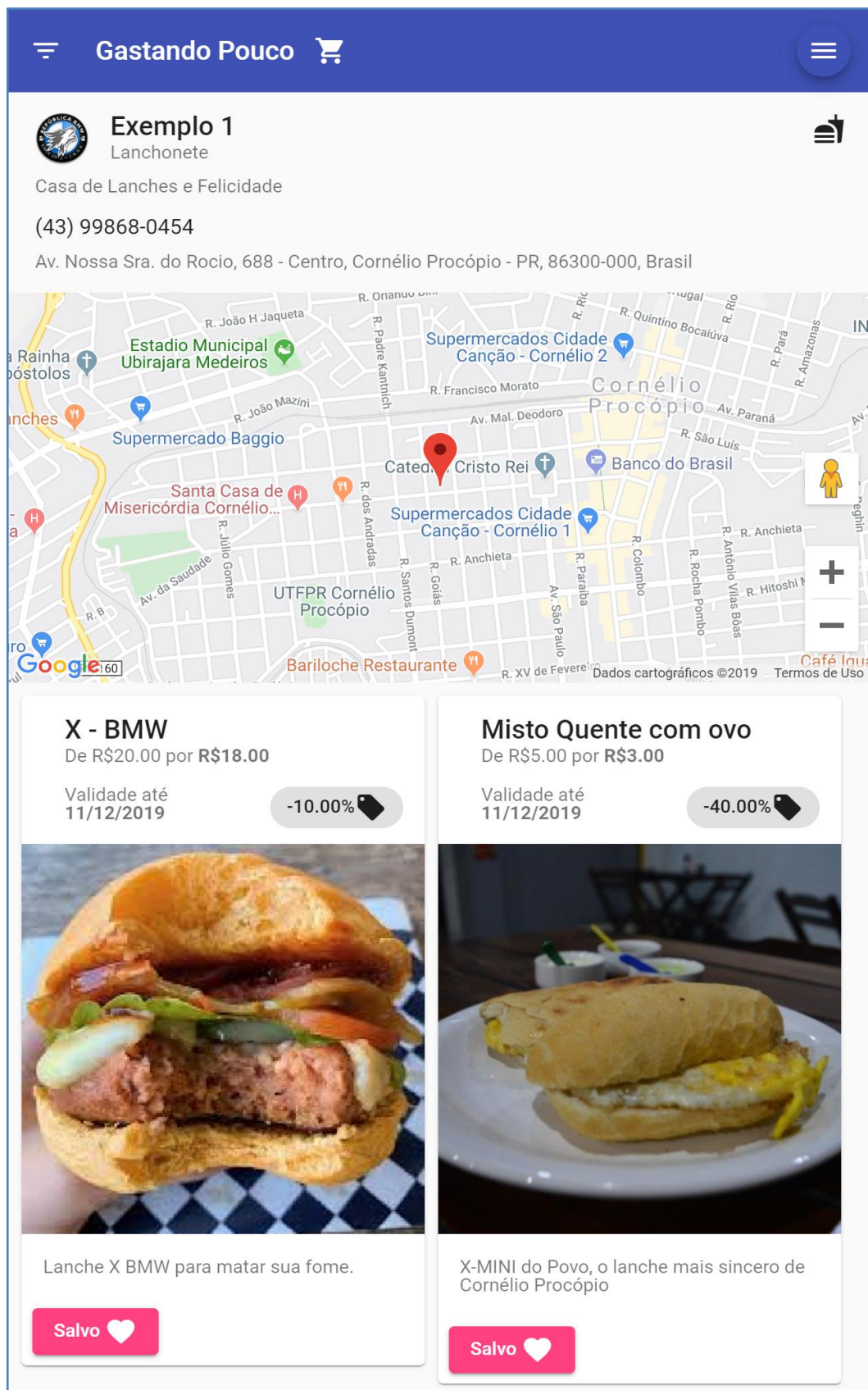


Fonte: Autoria Própria

4.1.5.5 TELA DE INFORMAÇÕES DA EMPRESA

A tela de listagem de promoções da empresa, contém as informações da empresa e suas respectivas promoções ativas, com opção de adicionar promoções e especificar como favoritas, apresentada na Figura 26.

FIGURA 26 - Tela de informações da empresa

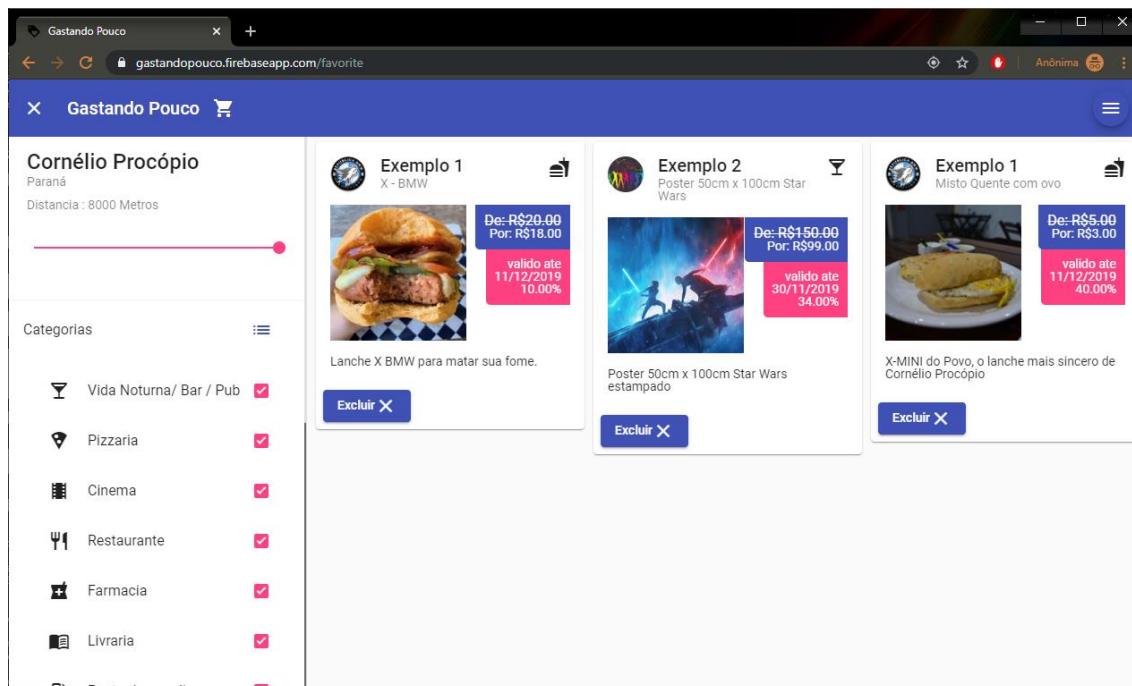


Fonte: Autoria Própria

4.1.5.6 TELA DE LISTAGEM DE PROMOÇÕES FAVORITAS

A tela de listagem de promoções favoritas, Figura 27, consiste em listagem de promoções salvas como favorita pelo cliente, com a possibilidade de excluir a promoção da listagem.

FIGURA 27 – Listagem de promoções favoritas



Fonte: Autoria Própria

As promoções marcadas como favoritas são adicionadas a contagem do relatório de promoções do domínio empresa.

5 CONSIDERAÇÕES FINAIS

O objetivo principal do trabalho foi desenvolver uma plataforma de publicação de ofertas orientados aos usuários da região, com ferramentas de código livre, com uma curva de aprendizagem e desempenho.

O desenvolvimento da plataforma *Gastando Pouco*, está disponível para acesso via navegador *web*, nos respectivos *links*:

- Domínio cliente <https://gastandopouco.web.app/>
- Domínio empresa <https://admgastandopouco.web.app/>

O software desenvolvido foi uma experiência de colocar em prática os conhecimentos adquiridos durante a graduação em Engenharia de Software, permitiu implementar um MVP (mínimo produto viável) que agregue a sociedade a listagem de produtos em ofertas na região. Para isso, foi utilizado o conjunto de tecnologias do acrônimo *MEAN Stack* e a implementação do protocolo de segurança *OAuth*, os dados de localização, aliados a metodologia de desenvolvimento ágil *Scrum Solo*, permitiu alcançar os requisitos funcionais e não funcionais definidos na fase de projeto.

A aplicação desenvolvida e relatada neste trabalho é um exemplo de MVP de software *MEAN Stack* e usa o processo de software ágil *Scrum Solo*. O código fonte, pode ser utilizado para novos escopos com as implementações de tecnologias e ferramentas relacionadas com o desenvolvimento ágil de software e segurança. Na implementação foi utilizado um conjunto de tecnologias, como protocolo *OAuth*, *Firebase*, *Mongoose*, *MongoDB*, *Flex Layout*, *Material Design* e *Angular*.

Toda documentação e o código fonte estão disponíveis de forma aberta na plataforma *GitHub* no link <https://github.com/magnocarvalho/tcc>.

5.1 DIFICULDADES E LIMITAÇÕES

As dificuldades inerentes a utilização de novas tecnologias. As tecnologias utilizadas no desenvolvimento, tem algumas limitações sendo necessário que o navegador tenha a implementação de API de *Geolocation* HTML5, recomenda-se o uso de uma versão atualizada dos navegadores mais populares entre eles, *Chrome*, *Firefox* e *Edge*. O Quadro 12 apresenta a relação de suporte entre as tecnologias e a versão do navegador que a tecnologia foi implementada.

Quadro 12 – Suporte às tecnologias usadas pelos principais navegadores

Tecnologia	Navegador				
	Chrome	Firefox	Edge	Safari	Opera
ES6 <i>ECMAScript</i> 2015 ²	58	54	14	10	55
<i>Flex Layout</i> ³	29	28	11	10	48
CSS <i>transform Property</i> (2D) ⁴	36	16	10	9	23
CSS <i>transform Property</i> (3D) ⁵	36	16	10	9	23
CSS <i>@keyframes</i> ⁶	43	16	10	9	30
<i>Api Geolocation</i> HTML5 ⁷	50*	3.5	9	5	16

*A partir do Chrome 50, a API de *Geolocation* funcionará apenas com HTTPS.

Fonte: Autoria Própria

A plataforma é limitada a função de organizar as promoções publicadas, ordenadas, utiliza como parâmetro a localização do usuário.

Não acontecendo nenhuma transação financeira ou interação entre os usuários empresa e cliente dentro da plataforma.

5.2 TRABALHOS FUTUROS

Plataforma Gastando Pouco, admite o desenvolvimento de um aplicativo para as plataformas Android e IOS e o desenvolvimento de novas funções, tais como: o compartilhamento das promoções em outras redes sociais, gerar código de desconto dentro da aplicação, adicionar notificações *push* para o usuário da região da empresa com as novas promoções cadastradas e ampliar as áreas de interesses dos usuários.

² https://www.w3schools.com/js/js_es6.asp

³ https://www.w3schools.com/cssref/css3_pr_flex.asp

⁴ https://www.w3schools.com/css/css3_2dtransforms.asp

⁵ https://www.w3schools.com/css/css3_3dtransforms.asp

⁶ https://www.w3schools.com/cssref/css3_pr_animation-keyframes.asp

⁷ https://www.w3schools.com/html/html5_geolocation.asp

REFERÊNCIAS

ADORNES, Gustavo Scheffel. **Motivações ao uso de tecnologia colaborativa: o caso do Waze**. 2016.

AGARWAL, Uttam. **Hands-On Full Stack Development with Angular 5 and Firebase: Build real-time, serverless, and progressive web applications with angular and Firebase**. Packt Publishing Ltd, 2018.

ALVES, William Pereira. **Desenvolvimento de Aplicações Web com Angular 6**. Alta Books Editora, 2019.

AMARAL, Pedro Henrique Silvério. **Marketing Digital E Sua Evolução**. 2018.

BEEDE, Rodney. **Analysis of reCAPTCHA effectiveness**. University of Colorado at BoulderDec, 2010.

BERA, Marcio H. Gimenes; MINE, Anderson Fernandes; LOPES, Luiz Fernando B. **MEAN Stack: Desenvolvendo Aplicações Web Utilizando Tecnologias Baseadas em JavaScript**. 2015.

BERNARDI, JOSÉ VICENTE ELIAS; LANDIM, PAULO M. BARBOSA. **Aplicação do Sistema de Posicionamento Global (GPS) na coleta de dados**. DGA, IGCE, UNESP/Rio Claro, Lab. Geomatemática, Texto Didático, v. 10, n. 31, p. 2002, 2002.

BONFIM, FILIPE LEUCH; LIANG, M. **Aplicações Escaláveis com Mean Stack**. Monografia Graduação, 2014.

CANIUSE. **Can I use... Support tables for HTML5, CSS3, etc.** Disponível em: <<https://caniuse.com/#feat=geolocation>> Acesso em: 20 out 2019.

CAMPOS, André; SOARES, Evelin; MARTINS, Gabriela; YOSHIDA, Ligia; OLIVEIRA, Marcos; ZAINA, Luciana; **Avaliação de Comunicabilidade, Usabilidade e Retorno Emocional no Transporte Público: Um Estudo do Moovit**. In: Extended proceedings of XV Symposium on Human Factors in Computing Systems. 2016.

CUPONERIA, **Cupons de desconto exclusivos**. Disponível em: <<https://www.cuponeria.com.br/>> Acesso em: 27 mar. 2019.

DABBISH, Laura et al. **Social coding in GitHub: transparency and collaboration in an open software repository**. In: Proceedings of the ACM 2012 conference on computer supported cooperative work. ACM, 2012. p. 1277-1286.

DE CARLI, Iraci Cristina da Silveira; DE ARAÚJO GASTAL, Susana; GOMES, Micael Nozari. **Pokémon Go, Realidade Aumentada e Geolocalização: A gamificação nas suas possibilidades para o Turismo**. Revista Hospitalidade, v. 13, p. 01-17, 2016.

EEDEN, Ennèl van; CHOW, Wilson. **Global Entertainment & Média Outlook 2018 – 2022**. PwC. EUA. 2018. Disponível: <<https://www.pwc.com/gx/en/entertainment-media/outlook/perspectives-from-the-global-entertainment-and-media-outlook-2018-2022.pdf>> Acesso em: 24 out. 2019.

FIGUEREDO, Cristiane. **Publicidade na era do consumidor digital: como o crescimento das mídias sociais vem interferindo no modo de fazer publicidade.** 2017.

FONSECA JR, Luiz Carlos; FONSECA, Marcio Roberto; DE LIMA RANGEL, Hélio Augusto. **An applied study on Angular framework 2.** Unisanta Science and Technology, v. 7, n. 1, p. 18-25, 2018.

GOODCHILD, Michael F. **Citizens as sensors: the world of volunteered geography.** GeoJournal, v. 69, n. 4, p. 211-221, 2007.

GROUPON, Ofertas e Cupons de Desconto. Disponível em: <<https://www.groupon.com.br/>> Acesso em: 27 mar. 2019.

HANDY, Alex. **Node.JS pushes JavaScript to the server-side** – SD Times. 2011. Disponível em: <<https://sdtimes.com/JavaScript/node-JS-pushes-JavaScript-to-the-server-side/>>. Acesso em: 29 mar. 2019.

HOWS, David; MEMBREY, Peter; PLUGGE, Eelco. **Introdução ao MongoDB.** Novatec Editora, 2015.

JONES, Michael et al. **JSon web token (jwt).** 2012.

JONSSON, Peter et al. **Ericsson mobility report.** Ericsson: Stockholm, Sweden, 2018. Disponível em: <<https://www.ericsson.com/assets/local/mobility-report/documents/2018/ericsson-mobility-report-november-2018.pdf>> . Acesso em: 25 mar. 2019.

KASAGONI, Shravan Kumar. **Building Modern Web Applications Using Angular.** Packt Publishing Ltd, 2017.

MACHADO, Everto Fabio da Silva. **Sistema de geolocalização e rastreamento para a plataforma Android-Compass.** 2015.

MANDEL, Michael; LONG, Elliott. **Brazil's App Economy.** Progressive Policy Institute, 2017. Disponível em: <<https://www.progressivepolicy.org/publications/policy-memo/brazils-app-economy/>>. Acesso em: 20 mar. 2019.

MARDAN, Azat. **Express. JS Guide: The Comprehensive Book on Express.** JS. Azat Mardan, 2014.

MEW, Kyle. **Learning Material Design.** Packt Publishing Ltd, 2015.

MIDDLETON, Neil; SCHNEEMAN, Richard. **Heroku: Up and Running: Effortless Application Deployment and Scaling.** " O'Reilly Media, Inc.", 2013.

MORGAN, Andrew. **"The Modern Application Stack – Part 1: Introducing The MEAN Stack."** MongoDB, 26 Jan. 2017. Disponível em: <www.mongodb.com/blog/post/the-modern-application-Stack-part-1-introducing-the-mean-Stack>. Acesso em: 20 set. 2019.

MORAES, William Bruno. **Construindo aplicações com NodeJS.** Novatec Editora, 2018.

MORONEY, Laurence; MORONEY; ANGLIN. **Definitive Guide to Firebase**. Apress, 2017.

NONNENMACHER, Renata Favretto. **Estudo do comportamento do consumidor de aplicativos móveis**. 2012.

PAGOTTO, Tiago et al. **Scrum solo: software process for individual development**. In: Information Systems and Technologies (CISTI), 2016 11th Iberian Conference on. IEEE, 2016. p. 1-6.

PEJIĆ, Bojan; PEJIĆ, Aleksandar; ČOVIĆ, Zlatko. **Uses of W3C's Geolocation API**. In: 2010 11th International Symposium on Computational Intelligence and Informatics (CINTI). IEEE, 2010. p. 319-322.

PELANDO. **Ofertas, Cupons de Descontos e Promoções**. Disponível em: <<https://www.pelando.com.br/>> Acesso em: 27 mar. 2019.

PEREIRA, C. R. **Construindo APIs Rest com Node.JS**. 1. ed. São Paulo: Casa do Código, 2016.

PEREIRA, Paulo; TORREÃO, Paula; MARÇAL, Ana Sofia. **Entendendo Scrum para gerenciar projetos de forma ágil**. Mundo PM, v. 1, p. 3-11, 2007.

POLITOWSKI, Cristiano; MARAN, Vinicius. **Comparação de Performance entre PostgreSQL e MongoDB. X Escola Regional de Banco de Dados**. SBC, p. 1-10, 2014.

POWERS, S. **Aprendendo JavaScript**. São Paulo: Novatec, 2010.

PRESSMAN, Roger; MAXIM, Bruce. **Engenharia de Software. 8ª Edição**. McGraw Hill Brasil, 2016.

PRESSMAN, S. Roger. **Engenharia de Software: Uma Abordagem Profissional. 7ª. Edição**. Rio de Janeiro–RJ. Editora McGraw-Hill, 2011.

RIBEIRO, Sara. **Google e Facebook dominam publicidade online**. *Jornal de Negócios*, v. 23, 2017.

ROCHA, Eudson; ALVES, Lara Moreira. **Publicidade Online: o poder das mídias e redes sociais**. *Revista Fragmentos de Cultura-Revista Interdisciplinar de Ciências Humanas*, v. 20, n. 2, p. 221-230, 2010.

Rodrigues Gomes de Oliveira, A. e Zuchi, J. (2017) **MEAN STACK**, *Revista Interface Tecnológica*, 14(1), p. 84-95. Disponível em: <https://revista.fatectq.edu.br/index.php/interfacetecnologica/article/view/141>
Acesso em: 21 out 2019.

SILVA, Maurício Samy. **Fundamentos de HTML5 e CSS3**. Novatec Editora, 2018.

SOMMERVILLE, Ian. **Engenharia de Software, 8 edição**. Pearson, Addison Wesley, v. 8, n. 9, p. 10, 2007.

SPC Brasil e CNDL. **72% dos brasileiros mudaram seus hábitos financeiros por causa da crise econômica**. Disponível

em:<<https://www.spcbrasil.org.br/pesquisas/pesquisa/4281>>. Acesso em: 20 mar. 2019.

OCTALMIND. **Arquitetura MEAN Stack: Angular, Node, Express e MongoDB**. Disponível em: <<https://blog.octalmind.com/arquitetura-mean-Stack-angular-node-express-e-mongodb/>>. Acesso em: 20 jun. 2019.

TORRES, Victor Monteiro. **HTML e seus Componentes**. Revista Ada Lovelace, v. 2, p. 99-101, 2018.

TOTH, Renato Molina. **Abordagem NoSQL-Uma real Alternativa**. Sorocaba, São Paulo, Brasil: Abril, v. 13, 2011.

WAGNER, K. **Digital advertising in the US is finally bigger than print and television**. Disponível em: <<https://www.vox.com/2019/2/20/18232433/digital-advertising-facebook-google-growth-tv-print-emarketer-2019>> Acesso em: 16 mar. 2019.

WITTERN, Erik; SUTER, Philippe; RAJAGOPALAN, Shriram. **A look at the dynamics of the JavaScript package ecosystem**. In: 2016 IEEE/ACM 13th Working Conference on Mining Software Repositories (MSR). IEEE, 2016. p. 351-361.

XAVIER, Otávio C.; CARVALHO, C. L. D. **Desenvolvimento de Aplicações Sociais A Partir de APIs em Redes Sociais Online**. Relatório Técnico, UFG, Goiânia, 2011.

ZAMIR, Amir Roshan; SHAH, Mubarak. **Accurate image localization based on google maps street view**. In: European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2010. p. 255-268.