**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**CÂMPUS CORNÉLIO PROCÓPIO**

*Bacharelado em Engenharia de Software*

**DISCIPLINA:** *Oficina de Integração 2*

**PROFESSOR:** *Andrey Cabral Meira*

**PLATAFORMA DE VEICULAÇÃO**

**PUBLICITÁRIA POR GEOLOCALIZAÇÃO**

Magno Santos

Massallys Silva

Ykaro Andrade

Cornélio Procópio

2019

1. **Introdução**

Este capítulo apresenta informações sobre o contexto e justificativa no qual se baseia o desenvolvimento da proposta, bem como, a descrição da proposta e a organização do documento.

* **Contexto**

Segundo o estudo de Rocha e Alves (2010) publicidade em mídias onlines, sendo elas redes sociais como o Facebook e Linkedin, transformou o consumo de seus respectivos usuários, criando novas experiências e gerando 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 19° Pesquisa Global de Entretenimento e Mídia 2018-2022, da empresa PwC, realizada em 53 países, que analisou gastos dos consumidores e anunciantes (PRICEWATERHOUSECOOPERS, 2019).

Atualmente, o mundo possui aproximadamente 7.9 bilhões de celulares ativos, com uma previsão de acréscimo, a partir da disponibilização da rede 5G em vários países, chegando a 2024 na casa dos 8.9 bilhões de celulares, smartphones e celulares convencionais (ERICSSON, 2018).

O Brasil, segundo Meirelles (2018), em maio de 2018 possui aproximadamente 220 milhões de celulares avançados ou *smartphones*, somado a *tablets* e *notebooks*, que atinge o número de mais de 306 milhões de dispositivos portáteis, ou seja, aproximadamente 1,5 dispositivos portáteis por habitante brasileiro.

A demanda por aplicativos para smartphones e a maneira como as informações são divulgadas na Internet, a última, com maior destaque, refletem na indústria da publicidade digital ou publicidade online. O desafio é acompanhar a grande e rápida expansão do crescimento da própria internet.

A publicidade digital é a principal fonte de receita de grandes empresas de Internet, Google e Facebook, com participação de 20% do total da publicidade global (PAULO, 2017).

O faturamento total das empresas com publicidade digital em 2017 chegou a 27% e em 2022 os investimentos em publicidade chegarão a 33%, atingindo a taxa de crescimento de 8% ao ano em todo o mundo (PRICEWATERHOUSECOOPERS, 2019).

Para Agrela (2017) 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 com o celular.

* **Justificativa**

O crescente número de acessos à Internet por dispositivos móveis em 2018 apresenta um cenário atrativo para a publicidade online (MOLLA, 2018). A popularização da Internet permite às empresas utilizar novos meios de divulgação da informação, antes, usavam meios tradicionais de comunicação para veiculação de publicidade, como as redes de televisão, rádio, jornais e revistas ao grande público.

O avanço da tecnologia trouxe a mudança do comportamento do consumidor. Deste modo, a publicidade tradicional pode não ter a mesma eficiência de antes, como meio de informação, atração e persuasão.

Para Figueredo (2017) é a hora dos profissionais de publicidade ir à busca de outras soluções e mídias, revendo as técnicas de produção publicitária e reinventando a publicidade.

Ao se propor o desenvolvimento de uma plataforma de veiculação de publicidade por georreferenciamento ao público regional, pretende-se 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, são propícias a alternativas e buscam menores preços (SPCBRASIL e CNDL, 2018).

* **Proposta**

Desenvolver uma aplicação WEB que deste a interface de acesso até a aplicação do lado servidor, escrita em linguagem Typescript para ser modelo para futuros estudos no desenvolvimento de software utilizando as tecnologias citadas nesta proposta em conjunto. Utilizando como escopo do software a ser desenvolvido, uma proposta de produto que surgiu durante uma competição de empreendedorismo, o Startup Weekend 2016 , realizada em Londrina, estado do Paraná.

* **Organização do Documento**

A estrutura do presente trabalho está dividida em 8 Capítulos. No qual o 1 capítulo se refere a introdução do documento, capítulo 2 é apresentado a descrição geral do sistema. O Capítulo 3 refere-se ao desenvolvimento do sistema, nele são descritas as tecnologias a serem utilizadas, a metodologia e o cronograma previsto. O Capítulo 4 contém os requisitos funcionais e não-funcionais junto aos diagramas de caso de uso, no Capítulo 5 são apresentadas na análise do sistema o modelo do banco de dados do projeto e contendo os diagramas de classe e atividades. No capítulo 6 é apresentado os protótipos de telas e a descrição do código e por fim nos capítulos 7 e 8 respectivamente, são apresentadas as considerações finais e a bibliografia.

1. **Descrição Geral do Sistema**

* **Objetivos (Gerais e Específicos)**

1. Objetivo Geral

O objetivo principal da proposta é aplicar conceitos de Engenharia de Software para o desenvolvimento de um aplicativo web, que permitirá obter dados da localização dos dispositivos do usuário, via georreferenciamento, desenvolvida em Linguagem de programação TypeScript. O aplicativo proposto pretende disponibilizar as promoções publicadas na plataforma que estão próximas do usuário, utilizando como parâmetro a quantidade de quilômetros que o usuário fornecer, assim permitir a pesquisa por produtos em promoções em sua região. A plataforma online oferece às empresas uma alternativa para publicar as promoções de seu estabelecimento e acompanhar a interação com os clientes.

1. Objetivos Específicos

Os objetivos específicos a serem alcançados para sucesso do desenvolvimento desta proposta é a criação de uma interface de acesso e uma aplicação no modo servidor, baseada na tecnologia MEAN Stack, segundo Dickey (2014), são elas:

1. O desenvolvimento de aplicação multiplataforma que permitirá ao usuário consultar via web, utilizando de parâmetro sua localização, as promoções mais próximas vinculadas à plataforma.
2. O desenvolvimento de uma API para a integração e manipulação dos dados gerados pela plataforma proposta.

* **Limites e Restrições**

O projeto a ser desenvolvido terá restrição ao ambiente web com a linguagem typescript para o seu desenvolvimento, também será o usado o MongoDB como banco de dados.

* **Descrição dos Usuários do Sistema**

O sistema poderá ser utilizado por usuários que buscam perto de sua localização um preço mais acessível e/ou alguma promoção de algum produto que esteja procura. No caso de uma empresa, seria interessante pois, quanto mais clientes a empresa puder captar perto de onde esteja será melhorar para os seus interesses além, da empresa oferecer promoções ou produtos que estão por um preço mais acessível assim, com o intuito de atingir o seu objetivo que é atrair mais clientes.

1. **Desenvolvimento do Projeto**

* **Tecnologias e ferramentas**

Listagem das principais tecnologias e ferramentas a serem utilizadas no desenvolvimento da proposta.

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 (SILVA, 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). A escolha dessas tecnologias considera a aceitação dos principais navegadores utilizados atualmente.

1. ANGULAR

Angular é um framework JavaScript desenvolvido e mantido pela empresa Google. Ele foi criado para permitir a fácil reutilização de código tanto para aplicação web móvel, nativa móvel ou desktop (GOOGLE, 2019a). O Angular foi escolhido em declínio de outros frameworks de SPA (single-page application) por conter um conjunto de ferramentas que permitem o desenvolvimento mais ágil, com vários packages disponíveis no repositório NPM (Node Package Manager) e pode ser acrescentada a aplicação.

1. ANGULAR MATERIAL 2

Angular Material 2 implementa as especificações do Material Design da Google, com objetivo de criar uma linguagem visual que sintetize os princípios de boas práticas de design, permite o desenvolvimento que proporcione uma experiência unificada entre as plataformas e tamanhos de dispositivos (GOOGLE, 2019b).

O Angular não determina utilizar o framework CSS e não oferece componentes prontos. A versão Angular Material 2 auxilia o desenvolvimento, os componentes são facilmente manipulados por meio de suas tags próprias com atributos de nomes auto-explicativos, possui componentes como animações, tabelas, temas e suporta as duas versões mais recentes de todos os principais navegadores: Chrome (incluindo Android), Firefox, Safari (incluindo iOS) e IE11 / Edge (GOOGLE, 2019b).

O Angular Material 2 foi escolhido para proporcionar uma experiência unificada entre os usuários do IOS, Android e WEB, e por suas características design intuitivo, fluido e de simples compreensão.

1. JAVASCRIPT

A linguagem ECMA Script, popularmente conhecida por JavaScript, foi desenvolvida por Brendan Eich, possibilitou a revolução na maneira como a web trabalhava, 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).

1. TYPESCRIPT

O TypeScript é um conjunto da linguagem Java Script desenvolvido pela Microsoft que adiciona tipagem e outros recursos à linguagem. Segundo Foley (2012) foi projetada para agilizar o desenvolvimento das aplicações. O Angular utiliza TypeScript como linguagem padrão de desenvolvimento e possui suporte a JavaScript e Dart. O TypeScript será usado no desenvolvimento da aplicação do lado servidor, posteriormente, no teste da aplicação será convertida em JavaScript utilizando framework Gulp. A escolha do TypeScript considerou a quantidade de bibliotecas disponíveis e sua comunidade.

1. NODE.JS

O Node.js é uma plataforma para o desenvolvimento de aplicações do lado servidor que utiliza com a linguagem padrão o JavaScript, para criação de aplicações e páginas web de alta escalabilidade (NODEBR, 2016). Para Handy (2011) a plataforma é popular entre os desenvolvedores web e utilizada por grandes empresas como LinkedIn, Microsoft, GitHub e MySpace. A arquitetura do Node.js é composta em sua maior parte por componentes desenvolvidos em C++ e em JavaScript (NODEBR, 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. Estes módulos disponíveis no NPM servem para facilitar diversos aspectos relacionados à aplicação, como instalação de dependências, a compilação e a atualização de pacotes (NODE.JS, 2019). A escolha pelo Node.js se deve a sua arquitetura orientada a eventos, a sua comunidade ativa, sua alta escalabilidade e performance.

1. EXPRESS.JS

O Express.js é um framework e componente middleware que facilita o desenvolvimento de aplicações web com o Node.js é um dos pacotes mais populares entre os desenvolvedores (NPM, 2019). O Express.js permite instanciar servidores web e receber requisições HTTP (EXPRESS.JS, 2019). Foi escolhido por possuir um conjunto robusto de recursos, sua organização de rotas e views.

1. VISUAL STUDIO CODE

IDE Visual Studio Code é um editor de código de programação, desenvolvido pela empresa Microsoft, possui suporte embutido para as tecnologias a serem utilizadas no desenvolvimento do software proposto entre elas, JavaScript, TypeScript e Node.js, e a extensão de suas funções com adição de plug-ins desenvolvidos pela comunidade (MICROSOFT, 2019).

1. FIREBASE

Firebase e 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 (GOOGLE, 2019c). Para o desenvolvimento do software proposto será utilizado o Firebase Auth como gerenciador de autenticação dos usuários utilizando redes sociais ou e-mail, em todos os módulos da proposta e armazenamento das imagens das publicações.

1. MONGODB

O MongoDB é um banco de dados NoSQL, segundo o site DB-Engines (2019) usa um modelo de dados flexível e podem mudar facilmente, conforme a necessidade e evolução da aplicação a ser desenvolvida, mesmo sendo NoSQL. O MongoDB possui funcionalidades comuns aos bancos de dados SQL, por exemplo, operações de agregação e consistência rigorosa (MONGODB, 2019). O MongoDB foi o banco de dados escolhido entre os vários bancos NoSQL disponíveis, devido ser de código aberto e permitir a interação entre o framework mongoose.

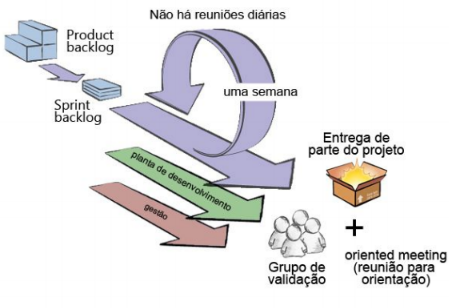
1. TRELLO

Software de gerenciamento de projetos que utiliza o kanban (em Japonês, Kan = “visual” e ban = “cartão”) de forma online e com vários colaboradores (TRELLO, 2019). Foi escolhido para manter a organização do desenvolvimento da fase de concepção deste documento até a aplicação.

* **Metodologia de desenvolvimento**

O processo de desenvolvimento seguirá a metodologia de desenvolvimento ágil Scrum Solo. O processo 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, 2016). A visão geral do Scrum Solo é apresentada na Figura 1.

FIGURA 1 - Visão geral ilustrada do processo de desenvolvimento do Scrum Solo.



Fonte: Adaptado de (PAGOTTO, 2016).

No desenvolvimento do projeto proposto utilizando o *Scrum Solo* será gerado o cronograma de processos, o *Requeriment* do projeto para definir os *product backlogs*, o levantamento e a validação de requisitos, como resultado espera-se obter um repositório online com todas as informações do projeto. Posteriormente, serão definidas as *Sprints* de desenvolvimento, uma vez que todos os *product backlogs* estarão definidos. Ao final de cada *Sprint* de uma ou duas semanas será gerado um *Deployment* para disponibilizar uma versão com as modificações realizada em cada *Sprint*.

O processo de desenvolvimento será registrado por completo, considerando que na fase atual, foi estimado o tal de 7 *sprints* para realização do projeto, descritos no item Cronograma previsto.

* **Cronograma previsto**

**Quadro 1 - cronograma do projeto**

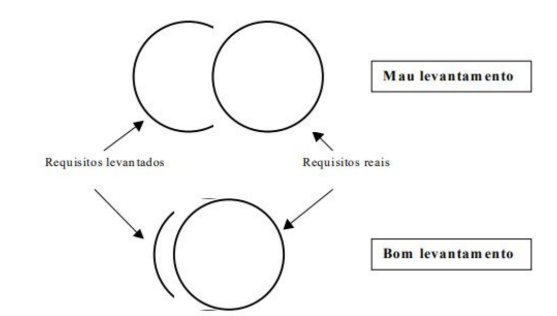
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sprints** | **Funcionalidades** | **Tempo Estimado** | **Início** | **Término** |
| **1°** | **Estruturação do desenvolvimento** | **4 dias** | **05/09/19** | **09/09/19** |
|  | Estruturação do ambiente de hospedagem e versionamento | 10 horas |  |  |
|  | Estruturação da aplicação do lado servidor | 1 dia |  |  |
|  | Estrutura Firebase | 5 horas |  |  |
|  | Estruturação de pipeline de entrega contínua **(Travis-ci)** | 1 dia |  |  |
| **2°** | **Início desenvolvimento aplicação empresa** | **8 dias** | **11/09/2019** | **19/09/19** |
|  | Layout e material *design* base da *dashboard* | 2 dia |  |  |
|  | Autenticação Firebase | 10 horas |  |  |
|  | Comunicação aplicação lado servidor **(HTTPCLIENT)** | 2 dia |  |  |
|  | Elaboração de componentes base de formulário e listagem | 2 dia |  |  |
| **3°** | **Continuação do desenvolvimento aplicação empresa** | **8 dias** | **21/09/19** | **29/09/19** |
|  | Telas de autenticação login, “esqueci senha” e cadastre-se | **2 dias** |  |  |
|  | Formulário de cadastrar dados da empresa | **2 dias** |  |  |
|  | Implementação de Rotas, *canactive* e *guards* | **2 dias** |  |  |
| **4°** | **Finalização do desenvolvimento da aplicação da empresa** | **6 dias** | **01/09/19** | **07/10/19** |
|  | Formulário de cadastro de ofertas | **2 dias** |  |  |
|  | Componente de listagem de ofertas cadastradas | **1 dia** |  |  |
|  | Desativar e atualizar ofertas cadastradas | **2 dias** |  |  |
| **5°** | **Início desenvolvimento aplicação híbrida** | **08 dias** | **09/09/19** | **17/10/19** |
|  | Pacotes e dependências iniciais | **1 dia** |  |  |
|  | *Layout* e material *design*  Telas de autenticação redes sociais (Firebase auth) | **1 dia** |  |  |
|  | Comunicação aplicação lado servidor **(HTTPCLIENT)** | **1 dia** |  |  |
|  | Elaboração de componentes base listagem de ofertas | **6 horas** |  |  |
|  | Dependências cordova | **4 horas** |  |  |
|  | Georreferenciamento dispositivo | **1 dia** |  |  |
| **6°** | **Continuação do desenvolvimento aplicação híbrida** | **10 dias** | **20/10/19** | **30/10/19** |
|  | Implementação de Rotas, *canactive* e *guards* | **2 dias** |  |  |
|  | Listagem de ofertas da região | **3 dias** |  |  |
|  | Listagem de informações da empresa | **3 dias** |  |  |
| **7°** | **Finalização do desenvolvimento da aplicação híbrida** | **10 dias** | **31/10/19** | **10/11/19** |
|  | Selecionar oferta específica | **2 dia** |  |  |
|  | Realizar check-in obter desconto | **2 dia** |  |  |
|  | Publicar aplicação Web | **4 dias** |  |  |

1. **Requisitos do Sistema**

Segundo Pressman (2011), o levantamento dos requisitos é baseado em tarefas, levantado por todos os envolvidos diretamente e indiretamente, num 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 é um conjunto das definições que o software deve fazer, entre armazenamento de dados, detalhes técnicos manipulação, e os requisitos não-funcionais é um conjunto de requisitos que contém itens relacionados à qualidade e desempenho do software.

No levantamento dos requisitos deve acontecer a captura das necessidades dos usuários em relação ao produto, expressas na linguagem do usuário, sendo que quando o documento é bem feito, os requisitos têm maiores chances de serem desenvolvidos atendendo a expectativa do cliente (DE PÁDUA PAULA FILHO, 2003). A Figura 2 ilustra a afirmação dos autores.

FIGURA 2 - Relação de eficiência levantamento de requisitos Fonte: Adaptado de (DE PÁDUA FILHO, 2003)

* **Requisitos Funcionais**

**Quadro 2 - requisitos funcionais do projeto**

|  |  |  |
| --- | --- | --- |
| **ID** | **Funcionalidade** | **Prioridade** |
| RFA-001 | O Sistema deverá capturar a localização do usuário e enviá-la ao servidor. | Essencial |
| RFA-002 | O Sistema deverá requisitar ao usuário a permissão de acessar sua localidade. | Essencial |
| RFA-003 | O Sistema deverá calcular a distância entre o usuário e a  localização das empresas com promoções ativas próximas, pela localização do mesmo. | Essencial |
| RFA-004 | O Sistema deverá permitir que o usuário acesse informações sobre as empresas ativas com promoções. | Desejável |
| RFA-005 | O Sistema deverá filtrar a exibição de promoções pelo perfil do usuário. | Desejável |
| RFA-006 | O Sistema deverá permitir que o usuário faça o “CHECK-IN” na promoção que ele vai interagir. | Desejável |
| RFA-007 | O Sistema deverá usar os parâmetros de busca do usuário para configuração ao fazer a busca de promoções na região. | Importante |
| RFA-008 | O Sistema deverá usar como parâmetro inicial as 10 empresas localizadas na proximidade do cliente. | Desejável |
| RFA-009 | O Sistema deverá usar com padrão no seu *dashboard* carregamento dinâmico. | Importante |
| RFA-010 | O Sistema deverá permitir que o usuário liste suas promoções marcadas como favoritas. | Desejável |
| RFA-011 | O sistema da empresa deverá informar o alcance de visualização da empresa. | Importante |
| RFA-012 | O Sistema da empresa deverá permitir o cadastro de novas promoções pelo administrador da empresa. | Essencial |
| RFA-013 | O sistema da empresa deverá permitir o cadastro e edição das informações da empresa. | Essencial |
| RFA-014 | Sistema da empresa deve permitir que o administrador da empresa visualize a quantidade de ‘CHECK-IN’ e favoritos de suas promoções cadastradas. | Desejável |
| RFA-015 | O sistema empresa deverá permitir a edição das promoções já cadastradas. | Desejável |

* **Requisitos Não-funcionais**

**Quadro 3 - requisitos não-funcionais do projeto**

|  |  |  |
| --- | --- | --- |
| **ID** | **Requisito** | **Categoria** |
| RNFA-001 | O Sistema deverá ser escrito em linguagem TypeScript. | Compatibilidade |
| RNFA-002 | O Sistema deverá ser construído utilizando a ferramenta de desenvolvimento Visual Studio Code. | Compatibilidade |
| RNFA-003 | O Sistema deverá ter conectividade com a Internet para conexão com aplicação do lado servidor. | Disponibilidade |
| RNFA-004 | O Sistema deverá ter acesso à localização do usuário enquanto utiliza a aplicação. | Interoperabilidade |
| RNFA-005 | O Sistema deverá garantir a segurança das informações dos  usuários por meio de conexão autenticada com chave assimétrica. | Segurança |
| RNFA-006 | O Sistema deverá ter seu código disponibilizado em repositório público Github. | Confiabilidade |

* **Diagramas de Casos de Uso**

1. Caso de Uso - Requisitos Funcionais (RFA)

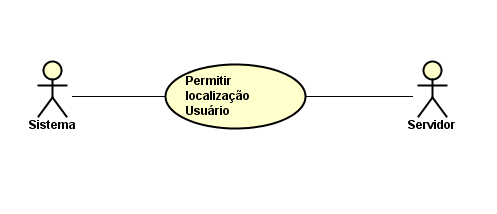


FIGURA 3 - ID: UC-001

FIGURA 4 - ID: UC-002

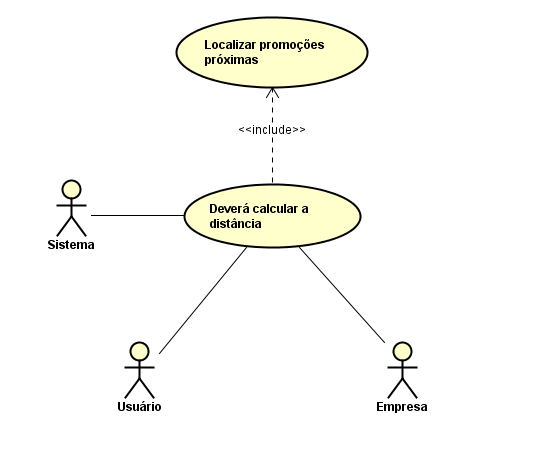


FIGURA 5 - ID: UC-003



FIGURA 6 - ID: UC-004

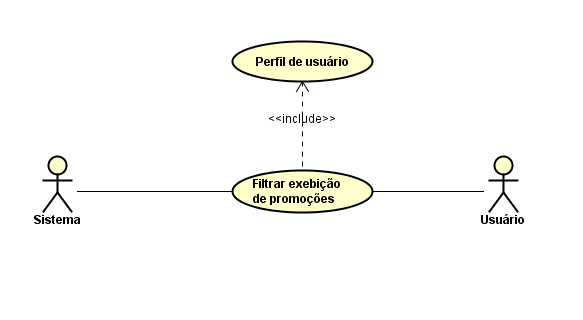


FIGURA 7 - ID: UC-005

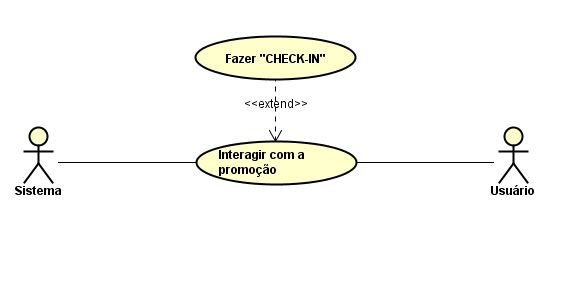


FIGURA 8 - ID: UC-006

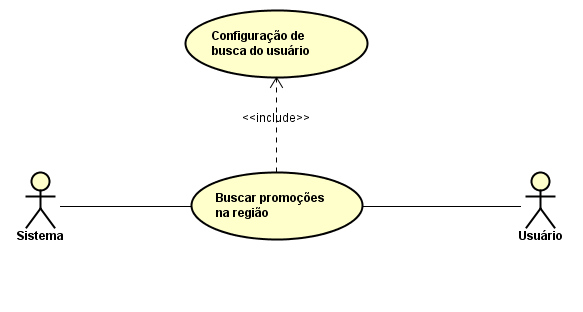


FIGURA 9 - ID: UC-007

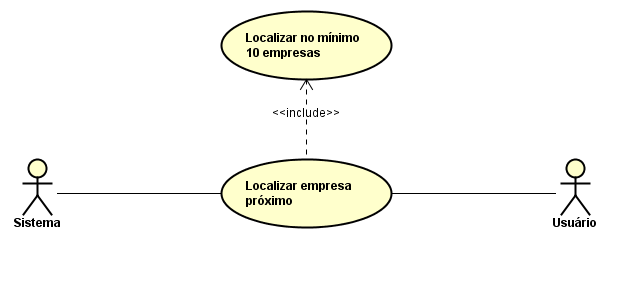


FIGURA 10 - ID: UC-008

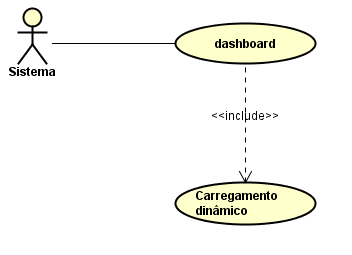


FIGURA 11 - ID: UC-009

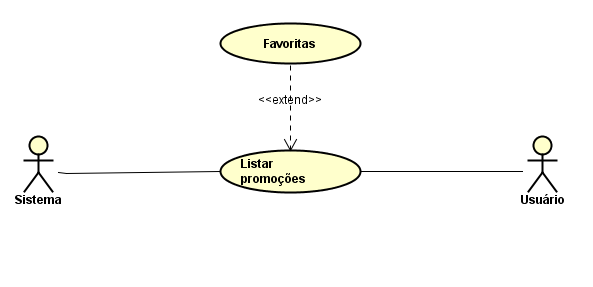


FIGURA 12 - ID: UC-010

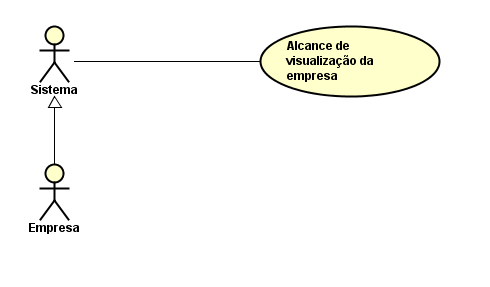


FIGURA 13 - ID: UC-011

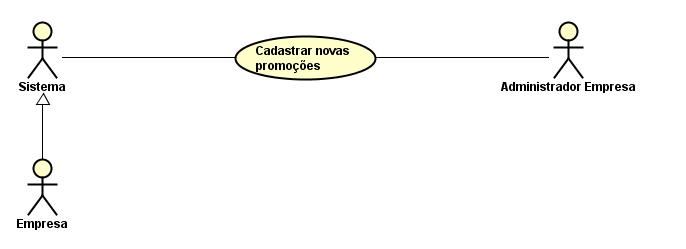


FIGURA 14 - ID: UC-012

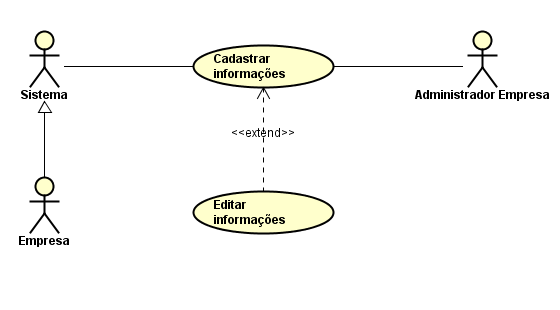


FIGURA 15 - ID: UC-013

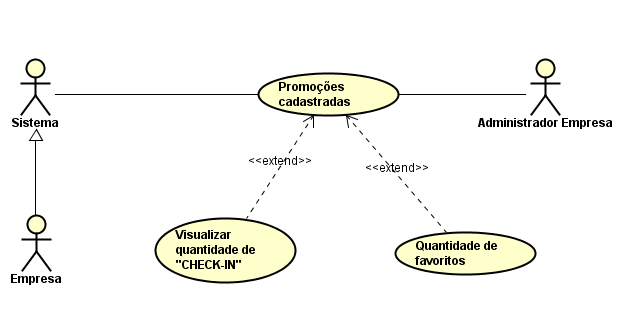


FIGURA 16 - ID: UC-014

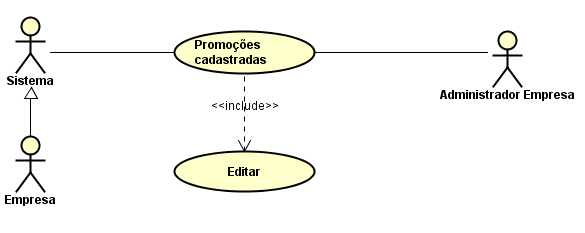


FIGURA 17 - ID: UC-015

1. Caso de Uso - Requisitos Não-Funcionais (RNFA)

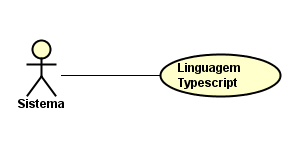


FIGURA 18 - ID: UC-016

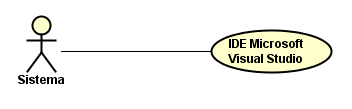


FIGURA 19 - ID: UC-017

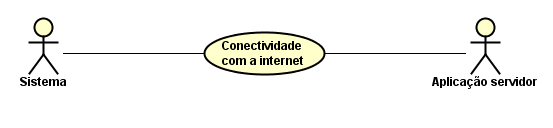


FIGURA 20 - ID: UC-018

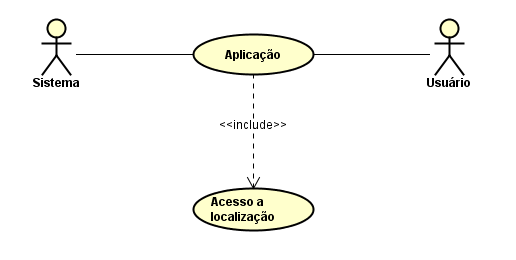


FIGURA 21 - ID: UC-019

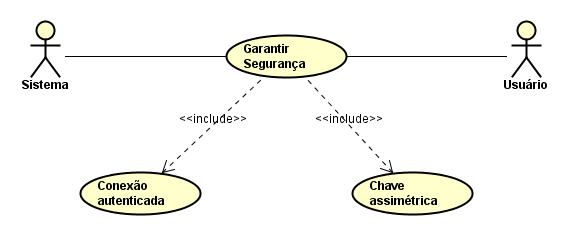


FIGURA 22 - ID: UC-020

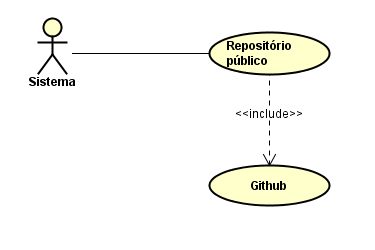


FIGURA 23 - ID: RNFA-UC-021

1. **Análise do Sistema**

**- Modelo do Banco de Dados**

Modelo Conceitual: Apresentar o Diagrama Entidade-Relacionamento desenvolvido para o banco de dados do sistema.

Modelo Lógico: Apresentar o esquema relacional (gráfico ou textual) do banco de dados normalizado e apresentando as tabelas com os atributos e restrições (chaves).

Dicionário de dados: Apresentar o dicionário de dados do banco de dados. Documentar cada tabela com seus atributos mostrando nome do atributo, tipo, tamanho, descrição, se é obrigatório ou não, e o que mais for necessário para descrever os dados. Documentar também usuários, stored procedures, funções e qualquer outra implementação ligada ao banco de dados.

**- Diagrama de Classes**

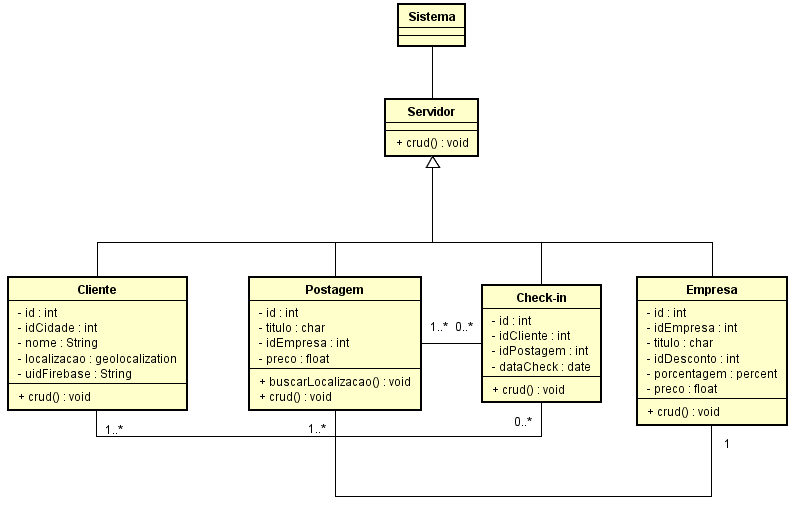


FIGURA 24 - ID: CD-001

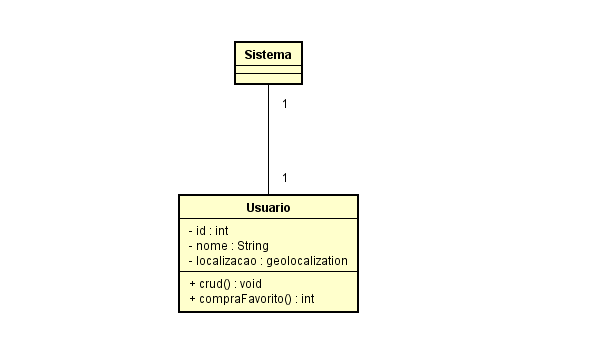


FIGURA 25 - ID: CD-002

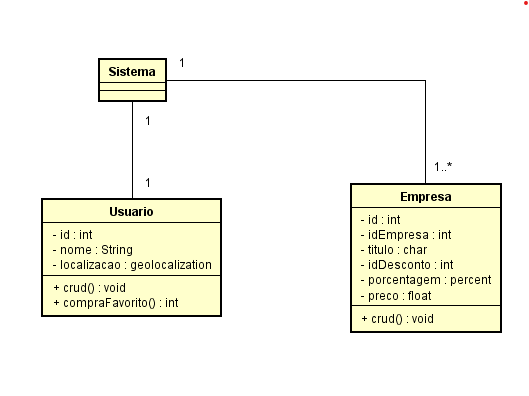


FIGURA 26 - ID: CD-003

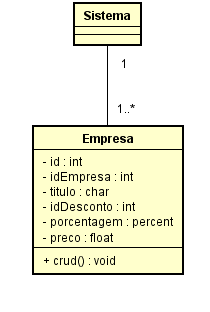


FIGURA 27 - ID: CD-004

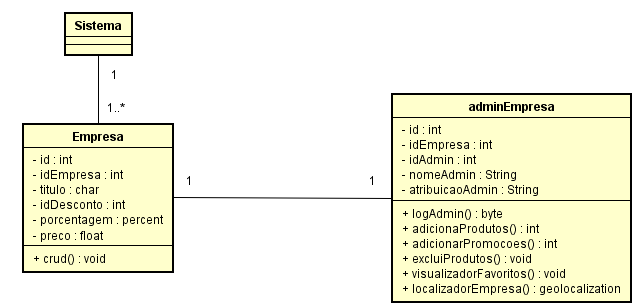


FIGURA 28 - ID: CD-005

**- Diagrama de Atividades**

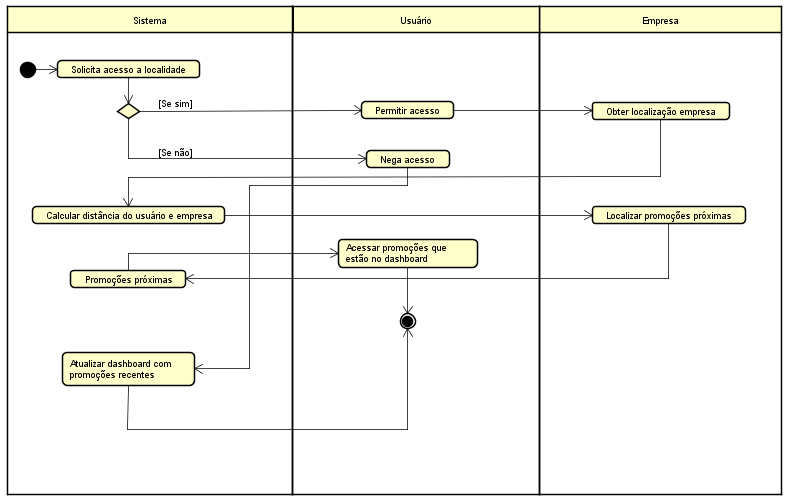


FIGURA 29 - ID: AD-001

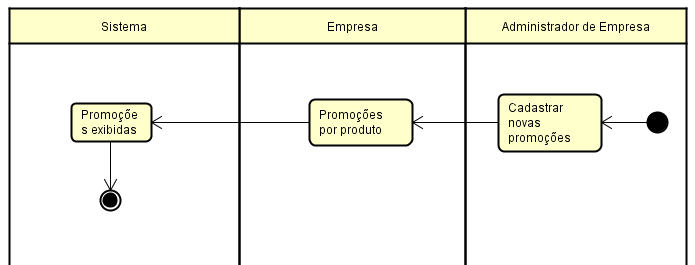


FIGURA 30 - ID: AD-002

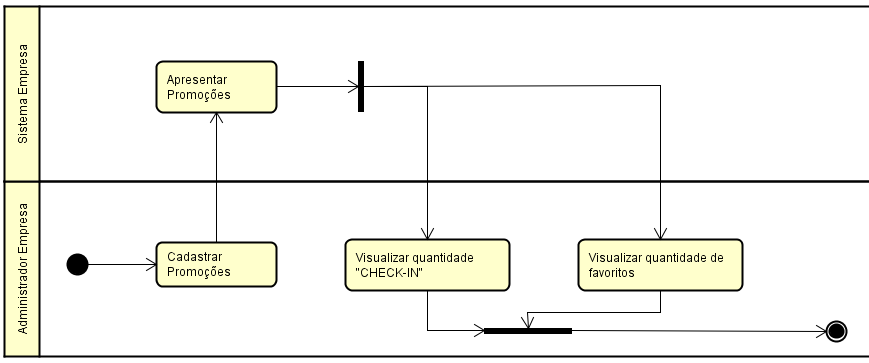


FIGURA 31 - ID: AD-003

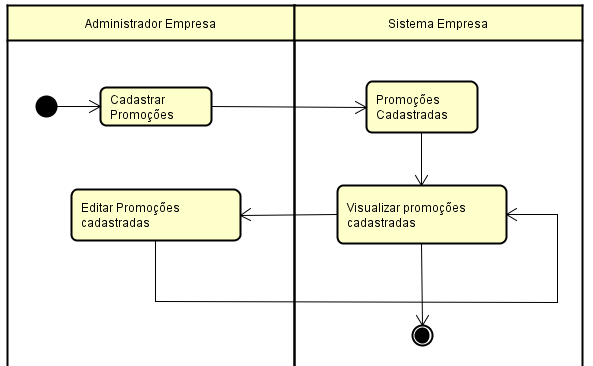


FIGURA 32 - ID: AD-004

1. **Implementação**

* **Protótipos de Telas**

Apresentar o protótipo do sistema, que consiste na interface preliminar contendo um conjunto de funcionalidades e telas. O protótipo é um recurso que deve ser adotado como estratégia para levantamento, detalhamento, validação de requisitos e modelagem de interface com o usuário (usabilidade).

As telas do sistema podem ser criadas na própria linguagem de desenvolvimento ou em qualquer outra ferramenta de desenho. Cada tela deve possuir uma descrição do seu funcionamento, constando pelo menos o objetivo da tela e dinâmica de navegação (de onde é chamada e que outras telas pode chamar). A descrição das telas deve registrar informações que possam ser consultadas para facilitar a implementação e a execução de testes, assim como a que requisitos funcionais se referem.

Se os protótipos de tela foram usados nas fases anteriores, esse item deve ser preenchido conforme os protótipos forem desenvolvidos e concluído com a versão final dos protótipos.

* **Descrição do código**

Descrever o sistema quanto ao código gerado. Explicar a organização dos arquivos, pacotes, classes ou quaisquer estruturas utilizadas no desenvolvimento do projeto, listando os componentes criados e sua estrutura. Use diagramas (Diagrama de Componentes, Diagrama de Pacotes) para ilustrar a implementação.

Descrever também convenções e padronizações para comentários no código, nomenclatura de classes, objetos, funções, etc. Se necessário, use exemplos.

1. **Considerações Finais**

Apresentar e discutir os resultados obtidos e sua aplicabilidade. Abordar o que foi atingido e o que não foi, as limitações, possíveis integrações com outros projetos e continuação do sistema em trabalhos futuros.

1. **Bibliografia**

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.

PRICEWATERHOUSECOOPERS. **19° Pesquisa Global de Entretenimento e Mídia 2018-2022**. Disponível em: <<https://www.pwc.com.br/pt/outlook-18.html>> Acesso em: 17 mar. 2019.

ERICSSON. **Ericsson Mobility Report November 2018**. Stockholm, 2018. Disponível em: [<https://www.ericsson.com/assets/local/mobility-report/documents/2018/ericsson-mobility-report-november-2018.pdf>](https://www.ericsson.com/assets/local/mobility-report/documents/2018/ericsson-mobility-report-november-2018.pdf). Acesso em: 25 mar. 2019.

PAULO, F. de S. **Google e Facebook têm 20% da publicidade global, diz consultoria**. Disponível em: [<https://www1.folha.uol.com.br/mercado/2017/05/1880881-google-e-facebook-tem-20-da-publicidade-global-diz-consultoria.shtml>](https://www1.folha.uol.com.br/mercado/2017/05/1880881-google-e-facebook-tem-20-da-publicidade-global-diz-consultoria.shtml). Acesso em: 15 mar. 2019.

AGRELA, Lucas. **Brasileiros estão cada vez mais viciados no celular**. Disponível em: <<https://exame.abril.com.br/tecnologia/brasileiros-estao-cada-vez-mais-viciados-no-celular/>>. Acesso em: 27 mar. 2019.

MOLLA, R. **Advertisers will spend $40 billion more on Internet ads than on TV ads this year**. Disponível em:<[https://www.recode.net/2018/3/26/17163852/online-internet-advertisers-outspe nd-tv-ads-advertisers-social-video-mobile-40-billion-2018](https://www.recode.net/2018/3/26/17163852/online-internet-advertisers-outspe)> Acesso em: 16 mar. 2019.

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

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

DICKEY, Jeff. **Write modern web apps with the MEAN stack: Mongo, Express, AngularJS, and Node. js.** Pearson Education, 2014.

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

GOOGLE. **Angular.** Disponível em: <<https://angular.io/>>. Acesso em: 29 mar. 2019.

GOOGLE. **Angular Material 2.** Disponível em:<<https://material.angular.io/>>. Acesso em: 29 mar. 2019.

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

APACHE. Overview. Disponível em: <<https://cordova.apache.org/docs/en/latest/guide/overview/index.html>>. Acesso em: 30 mar. 2019.

FOLEY, Mary Jo. Microsoft takes the wraps off TypeScript, a superset of JavaScript. ZDNet. CBS Interactive. Disponível em: <<https://www.zdnet.com/article/microsoft-takes-the-wraps-off-typescript-a-superset-of-javascript/>>. Acesso em: 29 mar. 2019.

NODEBR, **O que é Node.js?** Disponível em: <<http://nodebr.com/o-que-e-node-js/>>. 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.

NPM, Inc. **Most Depended upon Packages 2018.** Disponível em: <<https://www.npmjs.com/browse/depended#product-navigation/>>. Acesso em: 27 mar. 2019.

EXPRESS.JS. **Fast, unopinionated, minimalist web framework for Node.js.** Disponível em: <<https://expressjs.com/>>. Acesso em: 27 mar. 2019.

MICROSOFT. **Documentation for Visual Studio Code.** Disponível em: <<https://code.visualstudio.com/docs>>. Acesso em: 27 mar. 2019.

GOOGLE. **Firebase.** Disponível em:<<https://firebase.google.com/>>. Acesso em: 23 mai. 2019.

DB-Engines Ranking. **DB-Engines Ranking - Trend of Document Stores Popularity.** Disponível em: <<https://db-engines.com/en/ranking_trend/document+store>>. Acesso em: 27 mar. 2019.

MONGODB. **What Is MongoDB?** Disponível em: <<https://www.mongodb.com/what-is-mongodb>>. Acesso em: 27 mar. 2019.

TRELLO, inc. **About Trello.** Disponível em: <<https://trello.com/about>>. Acesso em: 29 mar. 2019.

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.

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

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

DE PÁDUA PAULA FILHO, Wilson. **Engenharia de software.** LTC, 2003.