
**CADSYSTEM: DESENVOLVIMENTO DE UMA APLICAÇÃO WEB DE GESTÃO
PARA CONTROLE DO CUSTO DE PRODUTOS**

Luis Otávio Gonçalves Rodrigues SANTOS
Graduando em Sistemas de Informação – Uni-FACEF
luisotavio05@gmail.com

Mateus Cintra MANIGLIA
Graduando em Sistemas de Informação – Uni-FACEF
mahmaniglia14@gmail.com

Ely Fernando do PRADO
Mestre em Ciência da Computação – Uni-FACEF
elyfprado@gmail.com

Resumo

Dentro de um contexto competitivo como o atual cenário nas empresas, a gestão de custos se constitui num dos aspectos administrativos mais relevantes. Este artigo objetiva propor um sistema de Gestão de Controle de Custos de Produtos, oferecendo ao pequeno empreendedor um negócio duradouro. O desenvolvimento da aplicação *web* baseia-se em um sistema que calcula as quantidades e os valores gastos na fabricação de produtos, apresentando para o usuário o total gasto na fabricação. Visando o desenvolvimento de um *Software* de boa qualidade, foram usadas ferramentas e técnicas de Engenharia de *Software* que ajudaram o levantamento de requisitos, com questionários, entrevistas e análise do processo de produção. Propõe-se que este sistema de gerenciamento de custos seja de fácil manuseio, permita a empresa ser gerenciada, dando a ela maior margem de lucratividade.

Palavras-chave: Gestão. Custos. Aplicação Web. Manufatura. Produção.

Abstract

Within a competitive context as the current scenario in companies, cost management is one of the most relevant administrative aspects. This article proposes a product cost control management system, favoring the small entrepreneur a lasting business. The web application development is based on a system that calculates the quantity and values spent on manufacturing your product, returning to the user the values spent. Aiming at development a good quality software, tools were used, software engineering techniques that helped in a good requirement gathering, with questionnaires, interviews and analysis of the production process. It is proposed that this cost management system be easy to handle, allow the company to be management, giving it a greater margin of profitability.

Keywords: Management. Costs. Web application. Manufacture. Production.

1 Introdução

Uma das principais metas de todo empreendedor é tornar seu negócio sólido e duradouro. Essa missão pode estar relacionada a uma boa gestão contábil

de custos para administrar e controlar os gastos gerados na fabricação de produtos. É de suma importância que a organização faça o planejamento de custos, para que ela não tenha gastos desnecessários.

Para uma gestão de custo com qualidade é importante que o empreendedor conheça o seu negócio. Quando não se tem uma gestão de custo eficaz, a empresa pode cobrar valores que não são compatíveis com a realidade, podendo afetar de forma negativa seus lucros (ENDEAVOR BRASIL, 2015).

Como um gestor de controle de custos de produtos pode influenciar os resultados do custo de produtos? A partir desta pergunta temos a definição do problema de pesquisa deste trabalho. Devido a necessidade para controlar os custos de produtos fabricados, tendo em vista o planejamento e controle de custos, trazemos a solução de criar uma aplicação *web* para gerenciar os custos de cada produto de uma pequena empresa, oferecendo para os empreendedores uma visão do que está sendo gasto para fabricação de tais produtos cadastrados, permitindo o controle de gastos altos e desnecessários para a empresa. Com o sistema *web* o empreendedor pode obter vantagem competitiva ao saber os resultados precisos dos gastos de cada produto que antes não tinha com o controle manual das despesas e receitas. O empreendedor terá a visão pelo relatório final de controle de custos.

Este artigo tem como objetivo apresentar o desenvolvimento de um sistema capaz de calcular as quantidades e os valores gastos na fabricação de produtos e mostrar ao usuário o valor do custo daquele produto fabricado. Tendo em vista o planejamento e controle de produção, o usuário deverá cadastrar os produtos e os seus respectivos composições e, no final do cadastro, o sistema oferece ao usuário o custo de fabricação dos produtos cadastrados.

Esta pesquisa tem a finalidade de apresentar os resultados obtidos na análise e desenvolvimento de uma aplicação *web* que auxilie os empreendedores a controlar seus custos na fabricação de seus produtos de forma a ter uma melhor gestão de controle de custos.

Para o processo de desenvolvimento foram utilizadas várias ferramentas. Na gerência do projeto foi usada a metodologia Scrum para dividir as tarefas e organizar os entregáveis. Na documentação foram usadas ferramentas para gerar o BPMN (*Business Process Model and Notation*), Diagramas UML (*Unified Modeling Language*) e Matrizes de Rastreabilidade. No desenvolvimento foram utilizadas a biblioteca da interface de usuário o React e como sistema de gerenciamento de banco de dados o MongoDB.

Na segunda seção são apresentados os conceitos teóricos sobre a utilização de Sistema de Informação (SI) nos negócios, a importância dos custos para as organizações, a biblioteca de desenvolvimento React e definições de empreendedorismo e *startup* enxuta. Na seção três é apresentado o processo para o desenvolvimento da aplicação, mostrando o levantamento de requisitos, interface de usuário e resultados obtidos. Já a seção quatro apresenta a solução implementada com imagens de como ficou o sistema.

2 Referencial Teórico

Esta seção irá abordar temas relacionados a sistemas de informação aplicado na organização e administração das empresas, a importância dos custos para uma boa gestão financeira, conceitos de empreendedorismo e *startup* enxuta, a biblioteca de desenvolvimento da interface de usuário React e o gerenciador de banco de dados MongoDB.

2.1 Sistema de Informação nos negócios

Um sistema de informação é um conjunto de componentes relacionados que coletam, processam, armazenam e distribuem informações que servem para apoiar a tomada de decisões, a ordenação e o controle de uma organização (LAUDON e LAUDON, 2011). Além disso, os sistemas de informações auxiliam os gestores a analisar problemas, a observar assuntos relacionados a organização e a criar novos produtos e serviços.

O avanço da tecnologia nas últimas décadas levou as organizações à chamada era da informação. Uma nova era que mudou a forma como as pessoas e organizações vivem e trabalham com novos padrões de produtividade e competitividade (ELEUTERIO, 2015). Essa era tornou a informação uma ferramenta poderosa para determinar a qualidade das decisões das organizações.

No contexto organizacional um sistema é importante e útil, pois ele consegue armazenar dados e gerar informações com o propósito de apoiar funções ou processos dentro da organização.

A eficácia no uso de sistemas de informação consiste em implantar ou desenvolver sistemas que melhor se adaptem às necessidades da área de negócios da empresa, que sejam consistentes com a estratégia global da corporação, que contribuam para aperfeiçoar as atividades e as funções (LAURINDO, 2012, p. 74).

Para uma boa gestão da informação é possível obter apoio de recursos de Sistemas de Informação e outras tecnologias. Dentre esses recursos, existem os Sistemas de Informações Gerenciais (SIG), que são sistemas de apoio à gestão empresarial que atendem ao nível gerencial da organização, que suprem as necessidades de muitos gerentes e são capazes de gerenciar metas corporativas, oferecendo registros ou relatórios de desempenhos da organização, podendo monitorar e controlar a empresa apoiando no planejamento, controle e decisão. Normalmente, delimita-se o uso dos SIGs ao ambiente interno da organização (LAUDON; LAUDON, 2011).

2.2 A importância da Gestão de Custos

Os sistemas de custos são constituídos por uma sequência de procedimentos de identificação, acumulação, distribuição e apropriação de gastos aos produtos, obedecendo a critérios lógicos. Geralmente estes critérios são elaborados a partir do processo produtivo realizado pela empresa. Eles têm a finalidade de coletar os dados de custos que, de forma direta ou indireta, sejam identificados com algum objeto de custeio, organizá-los de forma que possam oferecer informações, as quais irão atender às necessidades de níveis diferentes da administração. Dessa forma, os sistemas de custos vão variar de empresa para empresa ficando à mercê das

necessidades de informações que estas necessitam e as particularidades do seu sistema produtivo (BLACK; EDWARDS, 1979).

Os sistemas de custos precisam sempre levar em consideração a qualidade do pessoal envolvido em sua alimentação e em processamento, a necessidade de informação do usuário final, a adaptação às condições específicas da empresa, a utilização de quantidades físicas associadas aos valores monetários e, acima de tudo, a relação entre sua utilidade ou a de cada informação e o sacrifício envolvido em sua obtenção (MARTINS, 2000, p. 126).

O ambiente no qual as empresas estão inseridas está em constante mudança, verificando-se o aumento da concorrência entre elas. As empresas precisam cada vez mais de novas formas para gerenciar seus recursos de modo a reduzir os custos, aumentar a produtividade e satisfazer os clientes (BRAGA; BRAGA; SOUZA, 2010). Os gestores não necessitam apenas de informações para controle, mas, principalmente, informações que permitam que eles tenham uma visão holística do negócio e que possam lhes auxiliar no desempenho de suas diversas funções (CARDOSO; BEUREN, 2010). Desse modo, a Gestão de Custos tornou-se uma importante ferramenta de auxílio no processo de tomada de decisão.

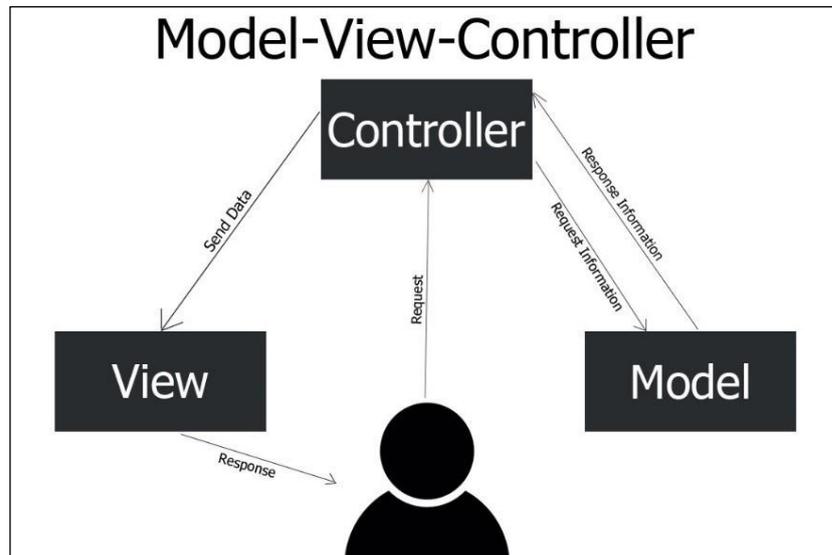
O custo é o consumo relativo a um bem ou serviço que é empregado na produção. O custo também é classificado como um gasto, mas somente reconhecido como custo no ato de sua utilização para fabricação do produto ou execução de um serviço (MARTINS, 2001, p. 26).

São exemplos de custos a matéria-prima e a embalagem. Os custos podem ser identificados como direto e indireto e como fixo e variável.

2.3 React

De acordo com Kostrzewa (2018) e Pandit (2017), o React é uma biblioteca JavaScript que foi criada em 2011 pela equipe de desenvolvedores do Facebook com o intuito de resolver problemas que causavam considerável transtorno na manutenção de elementos que compõem a interface do usuário. Sendo assim, desenvolveram uma biblioteca JavaScript com o objetivo de renderizar, tornar aplicativos *web* mais rápidos, escaláveis e simples. Aplicações em React podem manipular e alterar dados sem a necessidade de carregar páginas. Isso corresponde a camada visual do modelo Modelo-Visão-Controlador (MVC), conforme ilustra a Figura 1.

Figura 1 – Model-View-Controller



Fonte: MARGAN (2018, *online*).

Segundo Margam (2018), MVC é um padrão de arquitetura de software usado nas mais recentes linguagens de programação para desenvolver projetos escaláveis, ou seja, normalmente será usado para o desenvolvimento de interfaces de usuários que demandam grande quantidade de páginas.

No desenvolvimento em React ao invés de utilizar o JavaScript comum para a modelagem da interface de usuário, foi utilizado o JSX. De acordo com Pandit (2017), JSX é uma sintaxe que permite a citação em (Hipertexto de Marcação de Linguagem) HTML para renderizar subcomponentes. A sintaxe HTML é processada em chamadas JavaScript do React.

2.4 MongoDB

Segundo Banker (2011), o MongoDB é um sistema de gerenciamento de banco de dados projetado para aplicativos, que foi criado em 2007, com o intuito de criar um sistema flexível e escalável. O MongoDB é um SGBD orientado a documentos, ou seja, é um SGBD não relacional projetado para armazenar e consultar dados permitindo que seus documentos possam evoluir de acordo com a necessidade dos aplicativos.

Os SGBDs *NoSQL*, abreviação de *Not Only SQL* (não apenas SQL), são amplamente adotados em empresas como Facebook, Amazon e Google com o intuito de atender as suas demandas de estabilidade, alta disponibilidade e dados não estruturados. E vale ressaltar que, atualmente, diversos banco de dados *NoSQL* de código livre estão disponíveis, como exemplo o Cassandra, Hypertable, MongoDB, Redis, CouchDB e Dynamo.

O MongoDB foi projetado para expandir seu modelo de dados orientado a documentos facilita a divisão de dados em vários servidores. O MongoDB cuida automaticamente do equilíbrio dos dados, redistribuindo documentos e encaminhando as solicitações do usuário para a máquina correta, isso pode permitir que desenvolvedores concentrem no desenvolvimento de seus aplicativos sem ter que se preocupar com a arquitetura de armazenamento dos dados (CHODOROW, 2013).

2.5 Empreendedorismo

De acordo com Dornelas (2008), no Brasil o principal fator pelo interesse no termo empreendedorismo é de fato a preocupação em criar pequenas empresas duradouras e a necessidade da diminuição do fechamento dessas empresas. A mortalidade da empresa é um fator que todos os empresários querem evitar. Segundo o Instituto Brasileiro de Geografia e Estatística (IBGE), do total de 733,6 mil empresas abertas em 2010, somente 277,2 mil – 37,8% do total sobreviveram até 2015 (IBGE, 2017).

Ainda, de acordo com Dornelas (2008), é preciso um estudo mais profundo a respeito do empreendedorismo, tendo em vista que a maior parte dos negócios criados no país é concebido por pequenos empresários. Esses empresários nem sempre possuem conceitos de gestão de negócios, agindo geralmente sem prática e sem planejamento.

Schumpeter (1983) associa o empreendedor ao desenvolvimento da economia, à inovação e ao aproveitamento de oportunidades de negócios. O termo está relacionado à execução de planos para a realização de um novo negócio ou para a introdução de uma inovação de gestão dentro de uma organização.

De forma simples, pode-se entender como empreendedor aquele que inicia ou cria algo novo, que inova em algo, que aproveita as oportunidades transformando ideias num negócio lucrativo, que corre riscos e incertezas (ALVARENGA, 2012).

2.5.1 Startup Enxuta

O termo *startup* começou a se popularizar a partir da década de 90, quando surgiu a bolha da internet. Entretanto, somente no período de 1996 a 2001 que o conceito *startup* em empreendedorismo começou a ser empregado por empreendedores brasileiros, segundo especialista em *startup* (GITAHY 2016).

A definição de *startup* pode ser entendida de várias maneiras:

É uma empresa em fase embrionária, geralmente no processo de implementação e organização das suas operações. Pode não ter ainda iniciado a comercialização dos seus produtos ou serviços, mas já está a funcionar ou, pelo menos, em processo final de instalação (TABORDA, 2006, p. 6).

Segundo Ries (2012), uma *startup* pode ser considerada como uma instituição humana planejada para oferecer um novo serviço ou produto em condições de extrema incerteza. Dadas essas definições, entende-se que nada foi dito a respeito do tamanho da empresa, logo, qualquer indivíduo dentro de uma organização que esteja envolvido na criação de um produto ou serviço que esteja em um cenário de grande incerteza está envolvido numa *startup*.

Comecei a estudar outros setores, especialmente o industrial, que origina a maioria das teorias modernas de administração. Estudei a manufatura enxuta, processo que nasceu no Japão com o Sistema de Produção Toyota: um modo totalmente novo de pensar acerca da manufatura de bens físicos.

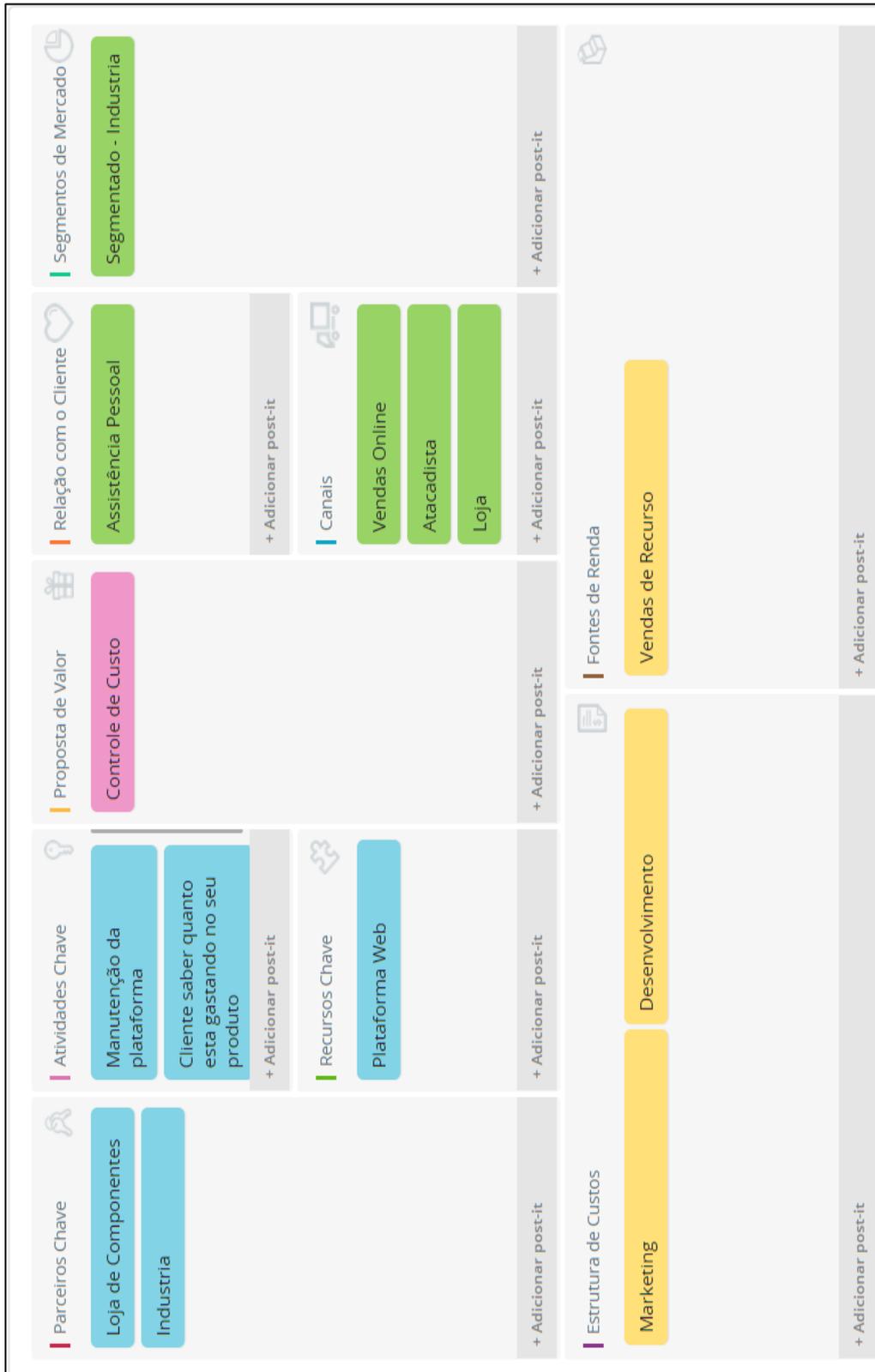
Essa linha de pensamento evoluiu até a startup enxuta: a aplicação do pensamento enxuto ao processo de inovação (RIES, 2012, p. 12).

Assim, a proposta de *startup* enxuta define-se a partir do método *Lean* da Toyota, onde é preciso evitar desperdícios, buscando uma forma de gestão de empresas, que seja característico aos fundamentos já existentes em uma *startup*. Dessa forma, Ries (2012, p. 15) afirma: “dispomos de um conjunto comprovado de técnicas para administrar grandes empresas[...]. No entanto, quando se trata de startups e inovação, ainda estamos atirando no escuro.” Uma *startup* deve buscar antes de tudo validar suas hipóteses, obtendo aprendizagem junto aos clientes, criando o mais rápido possível um produto pelo qual os clientes pagarão (NARDES; MIRANDA, 2014).

2.5.2 Canvas

O primeiro passo no desenvolvimento de um modelo de negócios é a definição do segmento de clientes, consistindo em dois aspectos importantes. O primeiro aspecto vem da palavra “segmento”, que possui um conceito implícito sobre a escolha de um pedaço do mercado, ou seja, é necessário que se escolha um nicho específico de clientes. O outro aspecto de um negócio deve ser desenvolvido a partir da perspectiva do cliente, enxergar por meios dos olhos de quem vai pagar pelo produto ou serviço que irá entregar (NARDES; MIRANDA, 2014).

Figura 2 – Modelo de Negócios (Canvas)



Fonte: Os autores.

O segundo passo é definir a Proposta de Valor do seu negócio, entender que o valor neste caso significa benefício. Portanto, é necessário que você saiba explicar quais os benefícios que o produto vai entregar ao cliente.

O terceiro passo é descrever quais os caminhos pelos quais a empresa comunica e entrega para o cliente, os canais podem ser de comunicação, vendas e distribuição do produto, qualquer tipo de interface da empresa com o cliente. Servem para que o seguimento definido possa tomar conhecimento e avaliar a proposição de valor do produto, efetuar a compra e uso do produto.

O bloco de Relação com o Cliente basicamente descreve quais as estratégias para que clientes não corram para um outro concorrente por questões como preço mais baixo. Uma boa estratégia de relacionamento com o cliente é fundamental para a retenção de clientes.

O lado direito do modelo de negócio, a Fonte de Renda é o bloco que determina a maneira como o cliente pagará pelos benefícios. Há várias formas de renda que podem ser adotadas no modelo de negócio Canvas, como, por exemplo: venda de produtos, aluguel, licença, etc.

O primeiro bloco do lado esquerdo é o de Recursos Chave, que são os ativos fundamentais para fazer o modelo de negócio funcionar. Esse bloco deve ser limitado a descrever o que realmente importa dentro dos ativos de sua empresa.

De forma complementar ao bloco de recursos chave, as atividades devem tratar dos deveres mais importantes que a empresa deve fazer de forma constante para que o modelo de negócio Canvas funcione corretamente.

As Parcerias Chave em um modelo de negócios referem-se primordialmente a terceirização. Como dito, o lado esquerdo trata de quesitos operacionais, portanto o bloco parcerias lista outras empresas que ajudam seu negócio a entregar a sua oferta (OSTERWALDER; PIG, 2011)

Seguindo todos esses passos foi elaborado o modelo de negócio do projeto, que está apresentado na Figura 2.

Para que possamos finalizar o modelo de negócio, o último bloco descreve todos os principais custos que tem peso no financeiro. Neste bloco de custos todos os outros blocos como o de recurso, atividades e parcerias chave. Porém, também será possível que custos de canais sejam considerados como, por exemplo, comissão de vendedores.

3 Projeto

Nesta seção são apresentadas as ferramentas que auxiliaram no desenvolvimento do sistema. Durante o processo foram usadas ferramentas que auxiliaram desde o início do projeto. Inicialmente foi realizado o levantamento de requisitos de uma pesquisa com alguns clientes, permitindo maior esclarecimento sobre os processos de fabricação dos produtos até o resultado do produto final.

3.1 Levantamento de Requisitos

Visando a boa qualidade do *software*, foram usadas técnicas de engenharia de software que auxiliaram a documentação do projeto. O levantamento de requisitos foi o primeiro passo para documentar o projeto. Uma das técnicas usadas foi a elaboração do questionário e a entrevista com os donos do negócio para esclarecer dúvidas e necessidades junto ao entrevistado. Para tanto, foi utilizado como estudo de caso, uma pequena fábrica de bolsas, de maneira que os requisitos pudessem abordar todo o processo produtivo deste tipo de produto.

Um requisito funcional importante é o Cadastro de Produtos. Onde deve conter a identificação do produto, descrição, tipo de produto, modelo, metragem do couro, metragem do zíper, quantidade de metal, metragem do forro, valor da mão de obra e um último campo que é calculado o valor do produto.

O sistema contém a tela de Cadastro de Matéria Prima para que o usuário faça o cadastro de todas as matérias primas que compõem os produtos, tendo a identificação da matéria prima, o seu valor unitário e botões para cancelar ou salvar o cadastro.

3.2 Documentação de Requisitos

Segundo Sommerville (2011), o Documento de Requisitos de *software* é um termo oficial esclarecendo o que os desenvolvedores do sistema devem implementar. A engenharia de requisitos está relacionada com a definição do que o sistema deve fazer. Deve incluir tanto os requisitos de usuários para um sistema quanto uma especificação detalhada dos requisitos do sistema.

O documento de requisitos abrange todos os usuários do sistema, que vão desde os administradores da organização até os engenheiros responsáveis pelo desenvolvimento do projeto. A Tabela 1 mostra uma parte da documentação de engenharia de requisito do sistema proposto.

Tabela 1 – Documentação de Requisitos

RF 002 – Cadastro de Produto	Categoria: () Oculto (X) Evidente	Prioridade: () Altíssima (X) Alta () Média () Baixa		
Descrição: O sistema contém a tela de Cadastro de produto para o usuário poder fazer o cadastro dos produtos fabricados.				
Requisitos funcionais				
Nome	Restrição	Categoria	Obrigatoriedade	Permanência

RNF 1.1 Identificar o código do produto.	O sistema irá gravar o código do produto que estiver sendo cadastrado.	Performance	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF 1.2 Identificar descrição do produto.	O sistema irá gravar as descrições dos produtos.	Performance	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF 1.3 Identificar tipo do produto.	O sistema irá gravar o tipo de produto que está sendo cadastrado.	Performance	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF 1.4 Identificar modelo do produto.	O sistema irá gravar o modelo do produto.	Performance	(X) Desejável () Obrigatório	(X) Permanente () Transitório
RNF 1.5 Identificar tipo e metragem do couro.	O sistema tem a função de gravar tipo e metragem do couro.	Performance	(X) Desejável () Obrigatório	(X) Permanente () Transitório
RNF 1.6 Identificar o zíper e a metragem de zíper.	O sistema irá gravar o zíper e a metragem do zíper.	Performance	(X) Desejável () Obrigatório	(X) Permanente () Transitório
RNF 1.7 Identificar tipo e quantidade de metal.	O sistema irá gravar a quantidade e o tipo do metal.	Performance	(X) Desejável () Obrigatório	(X) Permanente () Transitório
RNF 1.8 Identificar tipo e metragem do forro.	O sistema deve salvar a metragem e o tipo de forro.	Performance	(X) Desejável () Obrigatório	(X) Permanente () Transitório

RNF 1.9 Informar o valor da mão de obra.	Salvar o valor que é gasto com mão de obra.	Performance	(X) Desejável () Obrigatório	(X) Permanente () Transitório
RNF 2.0 Campo calculável.	O campo tem a função de fazer o cálculo de quanto vai custar a unidade do produto.	Performance	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF 2.1 Cancelar Cadastro.	O sistema deve cancelar tudo que foi preenchido na tela de cadastro de produto.	Performance	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF 2.2 Mensagem de cadastro salvo.	Depois de preencher os campos do cadastro do produto e salvar, o sistema mostra uma mensagem de sucesso.	Interface	() Desejável (X) Obrigatório	(X) Permanente () Transitório
RNF 2.3 Mensagem de erro na hora de salvar.	O sistema deve informar o usuário que tem algo inválido ou algum campo obrigatório não preenchido.	Interface	() Desejável (X) Obrigatório	(X) Permanente () Transitório

Fonte: Os autores.

Os requisitos do sistema não apenas especificam os serviços ou as características necessárias ao sistema, mas também a funcionalidade necessária para que esses serviços ou características sejam entregues corretamente (SOMMERVILLE, 2011, p.59). A documentação de requisitos completa do projeto está disponível no GitHub (SANTOS; MANIGLIA, 2019).

3.2.1 Requisitos Funcionais e Requisitos Não Funcionais

Os Requisitos Funcionais do sistema definem como um serviço ou característica do sistema deve agir ou se comportar em certas situações. Alguns requisitos funcionais especificam o que o sistema deve fazer (SOMMERVILLE, 2011).

Como o nome sugere, Requisitos Não Funcionais são requisitos que não estão relacionados a funcionalidades do sistema, podendo estar relacionados a propriedades emergentes de sistema, como por exemplo: confiabilidade, tempo de resposta, espaço em disco, entre outras (SOMMERVILLE, 2011).

3.3 Modelagem do negócio com BPMN

O principal objetivo do BPMN é fornecer uma notação padrão que seja facilmente compreensível por todos os intervenientes do negócio. Estas partes interessadas no negócio incluem os analistas de negócios que criam e refinam os processos, os desenvolvedores técnicos responsáveis pela implementação dos processos e os gerentes de negócios que monitoram e gerenciam os processos (WOLFARTH, 2014).

A notação BPMN também é útil para definir melhorias em processos, documentar processos existentes, identificar e automatizar processos, além de servir como uma linguagem comum para fazer a ponte de comunicação entre o design de processos de negócios e a implementação.

Atualmente existem vários padrões concorrentes para linguagens de modelagem de processos de negócio utilizados por ferramentas de modelagem e processos. A adoção generalizada do BPMN ajudará a unificar a expressão de conceitos básicos de processos de negócio (por exemplo, os processos públicos e privados, coreografias), bem como conceitos avançados de processos (por exemplo, tratamento de exceção, a compensação de transações).

A partir do levantamento de requisitos podemos formular uma documentação mais precisa sobre o projeto. A criação do BPMN auxiliou no entendimento dos processos de negócio.

3.4 Diagrama de Atividades

A Linguagem de Modelagem Unificada inclui diversos subconjuntos de diagramas, incluindo diagramas de estrutura, de interação e de comportamento. Diagrama de atividade, junto com Diagrama de Caso de Uso e de Máquina de Estado, são considerados diagramas de comportamento porque descrevem o que é necessário acontecer no sistema (LUCIDCHART, 2019).

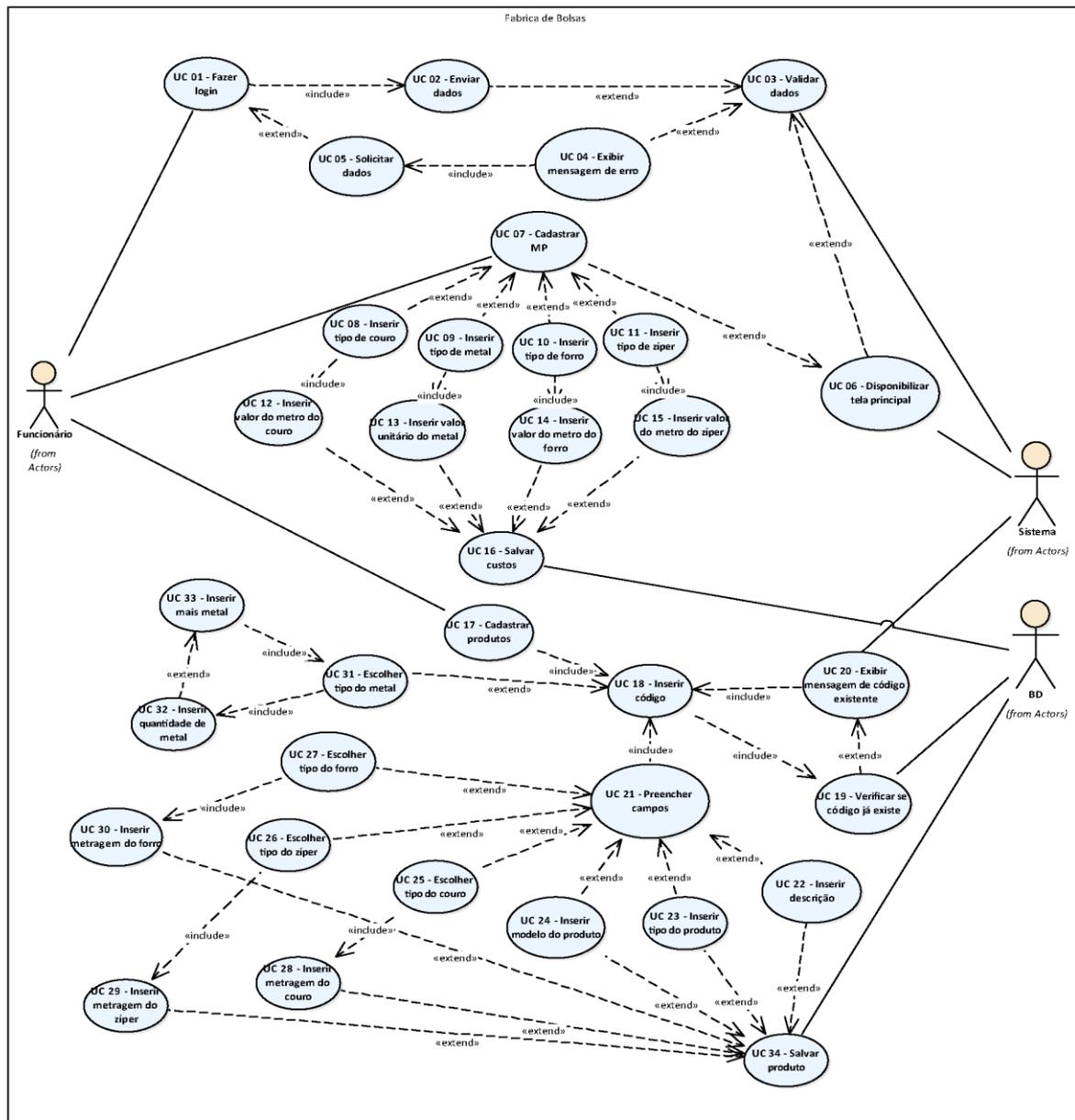
O objetivo do Diagrama de Atividade é mostrar o fluxo de atividades em um único diagrama, mostrando que uma atividade está diretamente relacionada a uma outra atividade.

3.5 Modelagem do Caso de Uso

O Diagrama de Caso de Uso documenta o que o sistema faz do ponto de vista do usuário. Ele descreve as principais funcionalidades do sistema e a interação dessas funcionalidades com os usuários do sistema. Ele é construído por um processo interativo no qual as discussões entre o cliente e os desenvolvedores do sistema conduzem a uma especificação do sistema da qual todos estão de acordo (RIBEIRO, 2012, *online*).

Um Caso de Uso descreve as operações que o sistema precisa fazer. Um caso de uso apresenta uma lista completa das interações entre um usuário e o sistema para cumprir uma lista de tarefas. Uma lista completa significa que o Caso de Uso descreve as interações desde o início da tarefa, até o fim, como mostra a Figura 3.

Figura 3 – Diagrama de Caso de Uso



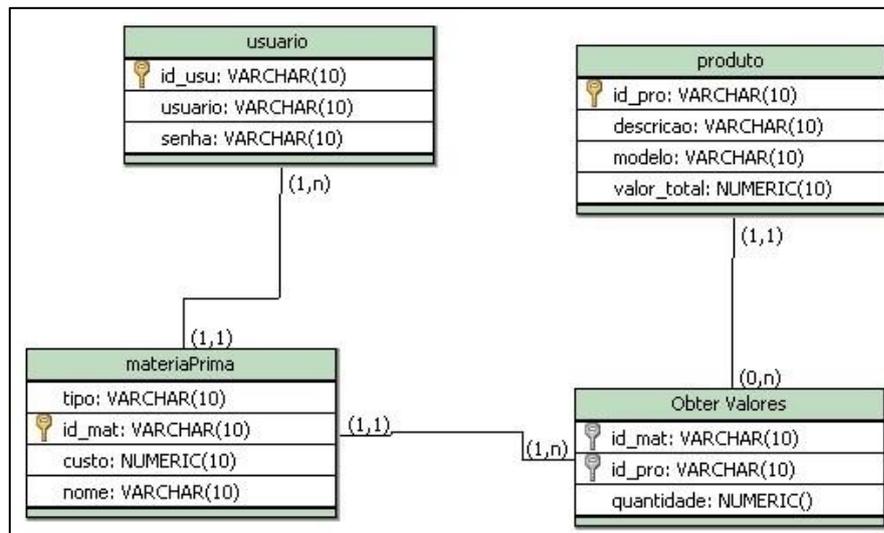
Fonte: Os autores

No Modelo de Caso de Uso consiste de Diagramas de Casos de Uso que mostram os atores e seus relacionamentos (LARMAN, 2000).

3.6 Modelagem do Banco de Dados

Antes de começar o desenvolvimento do banco de dados precisa-se fazer uma notação da modelagem. Como apresentado anteriormente, o MongoDB é um SGBD não relacional. Porém é indispensável o uso dos conceitos de relacionamento no desenvolvimento dos sistemas. Veja o modelo de entidade e relacionamento do projeto na Figura 4.

Figura 4 – Modelo Entidade e Relacionamento (MER)



Fonte: Os autores

Ainda assim, ao usar o MongoDB, que é uma ferramenta não relacional, pode-se criar o banco de dados com relacionamentos de duas maneiras: por referência, onde pode-se referenciar documentos de uma coleção em outra; por documentos embutidos, onde todas as referências serão englobadas dentro de um único documento (ALENCAR, 2018, *online*).

Na figura 5 podemos ver a estrutura do arquivo *JSON* contendo os dados e relacionamento em forma de referência de outro documento (tabela) do banco de dados (SANTOS; MANIGLIA, 2019).

Figura 5 – Estrutura do arquivo JSON

```

1 - [
2 -   {
3     "_id": "5d7ea258ad86732b28f290e2",
4     "description": "Bolsa feminina pequena",
5     "modelo": "X123",
6     "vl_montagem": 15,
7     "vl_tear": 8,
8     "__v": 0,
9     "createdAd": "2019-09-15T20:43:04.298Z",
10 -    "valores": [
11 -      {
12 -        "_id": "5d7e92bcf0346b123ce90f0d",
13 -        "mat_tipo": "Couro",
14 -        "mat_nome": "Nobuk marrom",
15 -        "mat_custo": 9.99,
16 -        "__v": 0,
17 -        "createdAd": "2019-09-15T19:36:28.105Z"
18 -      },
19 -      {
20 -        "_id": "5d7ea1f8ad86732b28f290e1",
21 -        "mat_tipo": "Zipper",
22 -        "mat_nome": "Zipper preto de metal 5MM",
23 -        "mat_custo": 5.45,
24 -        "__v": 0,
25 -        "createdAd": "2019-09-15T20:41:28.151Z"
26 -      }
27 -    ]
28 -   }
29 - ]

```

Fonte: Os autores

3.7 Scrum

Scrum tem o objetivo de fornecer um processo conveniente para projeto e desenvolvimento orientado a objeto. Scrum apresenta uma abordagem empírica que aplica algumas ideias de teoria de controle de processos industriais para o desenvolvimento, reintroduzindo as ideias de flexibilidade, adaptabilidade e produtividade. O foco da ferramenta é encontrar uma forma de trabalho dos membros da equipe para produzir o software de forma flexível e em um ambiente em constante mudança (SCHWABER, 2002).

A ideia principal do Scrum é que o desenvolvimento de softwares envolve muitas variáveis técnicas e do ambiente, como requisitos, recursos e tecnologia, que podem mudar durante o processo. Isto torna o processo de desenvolvimento imprevisível e complexo, requerendo flexibilidade para acompanhar as mudanças. O resultado do processo deve ser realmente útil para o cliente.

A ferramenta é baseada em princípios de: equipes pequenas, requisitos pouco estáveis ou desconhecidos e iterações (*sprints*) curtas para promover visibilidade para o desenvolvimento.

O Scrum divide o desenvolvimento em iterações. Equipes pequenas, de até dez pessoas, são formadas por projetistas, programadores, engenheiros e gerentes de qualidade. Estas equipes trabalham em funcionalidades definidas no início de cada *sprint*. A equipe é responsável pelo desenvolvimento desta funcionalidade.

No Scrum existem reuniões de acompanhamento diárias. Nessas reuniões, que são de curta duração (proximamente quinze minutos), são discutidos pontos como o que foi feito desde a última reunião e o que precisa ser feito até a

próxima. As dificuldades encontradas e os fatores de impedimento são identificados e resolvidos.

O ciclo de vida do Scrum é baseado em três fases principais, divididas em sub-frases: Pré-planejamento, Desenvolvimento, Pós-planejamento (SCHWABER, 1995).

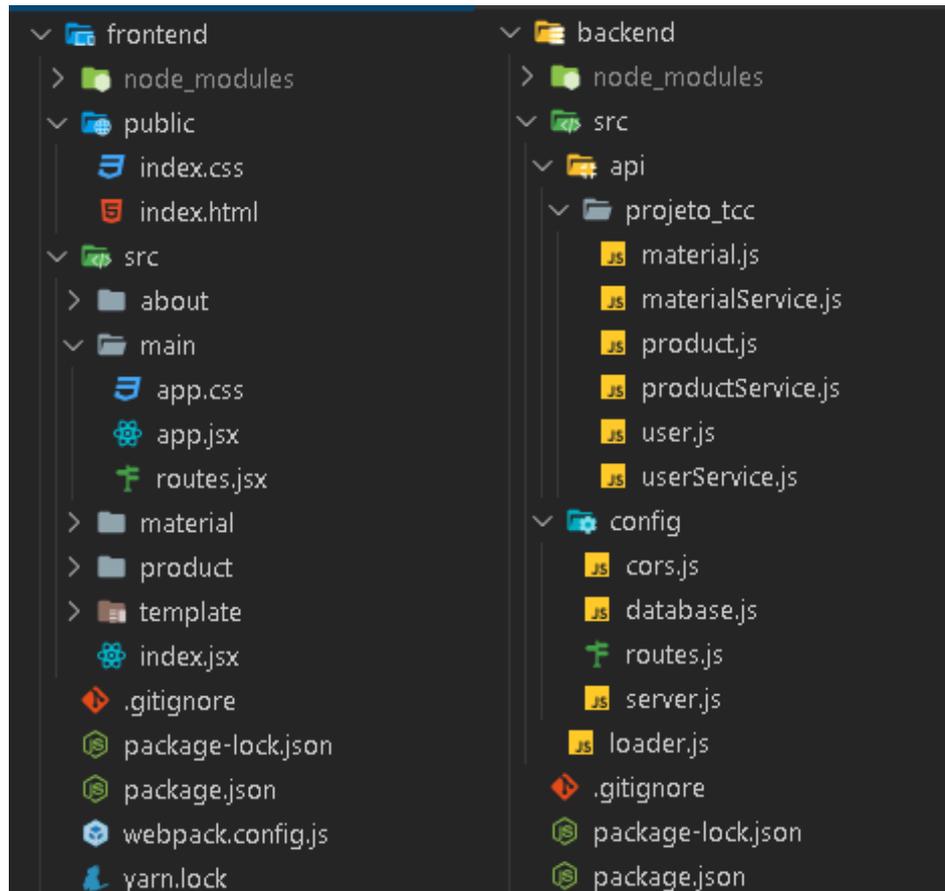
4 Resultados

Nesta seção serão apresentados a estrutura do código fonte, algumas funções em JavaScript e imagens do sistema com suas telas de cadastro. Entende-se que os resultados obtidos no desenvolvimento da aplicação *web* proporcionará ao usuário gerenciar os custos dos produtos, por meio de cadastros e relatórios obtidos a partir do acesso à internet.

4.1 Desenvolvendo o código

No desenvolvimento da aplicação *web* foi utilizado como ferramenta de edição de código o *Visual Studio Code*, um editor gratuito que foi desenvolvido pela Microsoft. A estrutura do projeto foi dividida em duas partes, pelo *frontend* pasta que contém a interface do usuário feito com React e pelo *backend* pasta que contém a parte interna do sistema como criação e conexão do banco de dados MongoDB. A figura 6 mostra a estrutura do código.

Figura 6 – Estrutura do código fonte



Fonte: Os autores

Os componentes são funções em JavaScript que manipulam os eventos da aplicação, como por exemplo, o botão de adicionar um novo produto ou matéria prima (Figura 7). Eventos são ações programadas para o usuário ou o sistema executar.

Figura 7: Evento alterar e adicionar

```

product.jsx ×
frontend > src > product > product.jsx
51   handleChange(key, value) {
52     let product = this.state.product
53     product[key] = value
54     this.setState({ ...this.state, product})
55   }
56
57   handleAdd() {
58     axios.post(URL, this.state.product)
59       .then(resp => this.refreshProduct())
60   }
61
  
```

Fonte: Os autores

Componentes em React permitem dividir a interface de usuário em partes independentes, reutilizáveis e cada parte isoladamente (REACT, 2019, *online*).

4.2 Telas

Com as telas implementadas, pode-se ter acesso a interface do usuário. As Figuras 8, 9 e 10 apresentam o resultado do desenvolvimento do projeto.

A Figura 8 apresenta a tela de Cadastro de Matérias Primas usados na fabricação das bolsas. A tela tem um botão no topo para cadastrar novas matérias primas. O usuário preencherá os campos informando o tipo, descrição e valor de custo do material. Logo após o botão vê-se a lista de Materiais que foram cadastrados e o usuário poderá apagar ou editar um registro.

Figura 8: Tela de cadastro e listagem de matéria prima

Tipo	Descrição	Custo	
Zíper	Zíper preto de metal 5MM	R\$ 5.45	Apagar Editar
Metal	Prata	R\$ 0.69	Apagar Editar
Couro	Nobuk marrom	R\$ 9.99	Apagar Editar
Forro	Listrado	R\$ 4.55	Apagar Editar

Fonte: Os autores.

A Figura 9 apresenta a tela de Cadastro de Produtos em que o usuário vai preencher o cadastro com os dados necessários para gerenciar o custo do produto. Para cadastrar um novo produto o usuário clica no botão de cadastro no topo da tela. Insere a descrição do produto, tanto quanto o modelo. Escolhe o tipo de couro, metal, forro e zíper e as quantidades que serão usadas de cada material. Logo após, existem dois campos para informar o valor da montagem e do tear, sendo que esse tipo de serviço é feito por empresas terceirizadas. Por fim, existem os botões de salvar ou cancelar o cadastro.

Figura 9: Tela de cadastro de produtos

Produto Cadastro

Cadastrar novo produto

Descrição do produto

Modelo

Material	Valor custo	Qtde.
Couro	R\$ 0,00	0,000
Metal	R\$ 0,00	0,000
Forro	R\$ 0,00	0,000
Zipper	R\$ 0,00	0,000

Valor montagem: R\$ 0,00 | Valor tear: R\$ 0,00 | Total: R\$ 0,00

Q Pesquisar | Salvar | Cancelar

Fonte: Os autores

Além do usuário poder cadastrar um novo produto ou matéria prima, pode também editar ou apagar algum registro.

A Figura 10 apresenta uma simulação de um cadastro de uma bolsa, contendo todas as matérias primas necessárias e suas quantidades. Logo abaixo dos campos preenchidos pelo usuário aparece o valor Total do custo do produto que está sendo visualizado para o usuário fazer a conferência e posteriormente salvar seu cadastro.

Figura 10: Simulação de cadastro de um produto

Produto Cadastro

Cadastrar novo produto

Descrição do produto

Modelo

Material	Valor custo	Qtde.
Couro	R\$ 9,99	1,785
Metal	R\$ 0,69	10
Forro	R\$ 4,55	2
Zipper	R\$ 5,45	1

Valor montagem: 15 | Valor tear: 8 | Total: R\$ 62,28

Q Pesquisar | Salvar | Cancelar

Fonte: Os autores

5 Conclusão

Devido à necessidade de controlar os custos de produtos fabricados, em vista do planejamento e controle de custos, as pequenas empresas podem nortear-se pela Gestão de Custos a fim de obter eficiência de seus processos internos no sentido de redução de custos e melhoria de seus produtos. Na verdade, tal flexibilidade é conquistada através da inovação, que permite à empresa operar, no mercado com margens de lucratividade.

Durante o desenvolvimento foram encontradas algumas dificuldades em relação ao relacionamento entre a modelagem do sistema utilizando o SGBD MongoDB, por ser um banco de dados não relacional em que esse tipo de banco de dados não se encaixa muito bem em um modelo relacional. Porém, ainda assim, foi possível criar esse relacionamento por meio de referências. Futuramente, pensa-se em migrar para um banco de dados relacional.

Como projeto futuro, será desenvolvida a tela de acesso, onde cada usuário poderá ter seu *login* e senha com suas permissões de acesso, dando melhor controle para o empreendedor. Outros projetos futuros serão a análise dos custos fixos e variáveis dos produtos da empresa, e a implantação do sistema em uma empresa real a fim de validar os requisitos desenvolvidos.

Referências

ALENCAR, Thais. MongoDB: como relacionar dados. 2018. Disponível em: <<https://medium.com/@thaisdalencar/mongodb-como-relacionar-dados-3e6e8f136590>>. Acesso em 23 de set. de 2019.

ALVARENGA, Adriane. O que é ser empreendedor. 2012. Disponível em: <<https://bis.sebrae.com.br/bis/download.zhtml?t=D&uid=A2EEEAD6407D759003256D520059B1F8>>. Acesso em: 25 de out. de 2019.

BANKER, Kyle. MongoDB in action. Manning Publications Co., 2011.

BRAGA, D. P. G.; BRAGA, A. X. V.; SOUZA, M. A. Gestão de custos, preços e resultados: um estudo em indústrias conserveiras do Rio Grande do Sul. Revista Contabilidade, Gestão e Governança – UNB, Brasília, v.13, n.2, p. 20-35, Maio/Ago., 2010.

BLACK, Homer A., EDWARDS, James DQM. The managerial and cost accountant's handbook. Home-wood, Dow Janes-R\Irwin, 1979.

CARDOSO, R. S.; BEUREN, I. M. Gestão de custos de matérias-primas em indústrias de conserva de pescado do Brasil e da Espanha. Revista ABCustos, v.5, n.2, p. 29-44, Maio/Ago., 2010.

CHODOROW, Kristina. MongoDB: the definitive guide: powerful and scalable data storage. " O'Reilly Media, Inc.", 2013.

DORNELAS, José Carlos Assis. Empreendedorismo. Elsevier Brasil, 2008.

ELEUTERIO, Marco Antonio Masoller. Sistemas de informações gerenciais na atualidade. Editora Intersaberes, 2015.

ENDEAVOR BRASIL. Gestão de Custos: como ter um bom controle financeiro. 2015. Disponível em: <<https://endeavor.org.br/financas/gestao-de-custos/>>. Acesso em: 04 de out. de 2019.

GITAHY, Yuri. O que é uma startup? 2016. Disponível em: <<https://exame.abril.com.br/pme/o-que-e-uma-startup/>>. Acesso em: 12 de set de 2019.

IBGE- Instituto Brasileiro de Geografia e Estatísticas. Demografia das Empresas 2015. Rio de Janeiro, 2017.

KOSTRZEWA, Denis. Is React.js the Best JavaScript Framework in 2018? 2018. Disponível em: <<https://hackernoon.com/is-react-js-the-best-javascript-framework-in-2018-264a0eb373c8>>. Acesso em: 06 de set. de 2019.

LARMAN, Craig. Utilizando UML e Padrões - Uma introdução à análise e ao projeto orientados a objetos. BookMan, 2000.

LAUDON, K.; LAUDON, J. Sistemas de informação gerenciais. Tradução Luciana do Amaral Teixeira. 9. Ed. São Paulo: Pearson Prentice Hall, 2011.

LUCIDCHART. O que é um diagrama de atividades? 2019. Disponível em: <<https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-atividades-uml/>>. Acesso em: 24 de set. de 2019.

LAURINDO, Fernando José Barbin. Tecnologia da Informação - Eficácia nas Organizações. São Paulo: Futura, 2002.

MARGAN, Girish. Model-View-Controller (MVC). 2018. Disponível em: <<https://medium.com/datadriveninvestor/model-view-controller-mvc-75bcb0103d66>>. Acesso em: 14 de set de 2019.

MARTINS, Eliseu. Contabilidade de Custo. Ed. 7, São Paulo: Atlas, 2000.

MARTINS, Eliseu. Contabilidade de Custo. Ed. 8. São Paulo, 2001.

NARDES, Felipe Bruno Souza; MIRANDA, Roberto Campos. Lean Startup e Canvas: uma proposta de metodologia para startups. Revista Brasileira de Administração Científica, v. 5, n. 3, p. 252-272, 2014.

OSTERWALDER, A; PIG, Y. Business Model Generation: Inovação em Modelos de Negócios. 1°. ed. Rio de Janeiro: Alta Books, 211.

PANDIT, Nitin. What Is ReactJS and Why Should We Use It? 2017. Disponível em: <<https://www.c-sharpcorner.com/article/what-and-why-reactjs>>. Acesso em: 27 de set de 2019.

REACT. Documentação: Componentes e Props. 2019. Disponível em: <<https://pt-br.reactjs.org/docs/components-and-props.html>>. Acesso em: 24 de set. de 2019.

RIBEIRO, Leandro. O que é UML e diagramas de caso de uso: introdução prática à UML. 2012. Disponível em: <<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408/>>. Acesso em: 24 de abr. De 2019.

RIES, Eric. A startup enxuta: The lean startup. 1 ed. São Paulo, Brasil: LEYA BRASIL, 2012. 276p.

SANTOS, Luis Otavio; MANIGLIA, Mateus. Documentação do projeto no GitHub. 2019. Disponível em: <https://github.com/luis-otavio/projeto_tcc/wiki>. Acesso em: 14 de set de 2019.

SCHUMPETER, Joseph Alois. Histoire de l'analyse économique: L'âge des fondateurs, des origines à 1790. Gallimard, 1983.

SCHWABER, K. "Scrum Development Process", OOPSLA'95 Workshop on Business Object Desing and Implementation. Springer-Verlag. (1995).

SOMMERVILLE, Ian. Engenharia de Software. 9. ed. São Paulo: Pearson Prentice Hall, 2011. 544p.

TABORDA, Ana. O que é uma start-up? 2006. Disponível em: <http://www.gesentrepreneur.com/pdf/o_que_e_uma_start_up.pdf>. Acesso em: 12 de set de 2019.

WOLFARTH, Goetz. BPMN. 2014. Disponível em: <<http://http://www.bgtbrasil.com/bpmn/>>. Acesso em: 10 de set de 2019.