

EasyProposal: Aplicação web para elaboração de propostas de vendas de máquinas de produção

Leonardo Schiarolli¹, Daiane Mastrangelo Tomazeti¹

¹Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas – Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) – Campus Hortolândia – São Paulo – SP – Brasil

leonardoschiarolli@hotmail.com, daianetomazeti@ifsp.edu.br

Abstract. *This work presents the process of developing a web application for the elaboration of sales proposals for production machines. The system was developed to facilitate and automate the control and preparation of proposals for companies that use manual forms for this flow. In the article in question we will see the way in which this project was developed, containing the methodology and methods used to build the system. For the development of the project, some tools such as Visual Studio and HeidiSQL were used, to obtain the final objective, which was the construction of the web application that will generate proposals on top of the data pre-registered in the system.*

Resumo. *Este trabalho apresenta o processo de desenvolvimento de uma aplicação web para a elaboração de propostas de venda de máquinas de produção. O sistema foi desenvolvido para facilitar e automatizar o controle e a elaboração de propostas para empresas que utilizam de formas manuais para este fluxo. No artigo em questão iremos ver sobre a forma na qual foi desenvolvida este projeto, contendo a metodologia e os métodos utilizados para a construção do sistema. Para o desenvolvimento do projeto foi utilizado algumas ferramentas como o Visual Studio e o HeidiSQL, para obter o objetivo final que foi a construção da aplicação web que irá gerar propostas em cima dos dados pré-cadastrados no sistema.*

1. Introdução

Um sistema de informação pode ser definido como um conjunto de componentes inter-relacionados trabalhando juntos para coletar, recuperar, armazenar e distribuir informação com a finalidade de auxiliar o planejamento, o controle, a coordenação, a análise e o processo decisório das organizações. (K. LAUDON; J. LAUDON, 1999b).

Baseando-se no fluxo de geração de proposta de uma empresa deste ramo, a geração da proposta começa com o preenchimento dos dados em uma planilha. Após esta etapa, a mesma é enviada ao cliente através de mensagem por e-mail. É importante enfatizar que estas duas ações são executadas de maneira manual.

As planilhas ficam privadas e com isto, caso seja necessário o reenvio aos clientes, é necessária uma pré-autorização que dificulta a agilidade da elaboração e envio das mesmas.

Todo esse processo é bastante trabalhoso para a equipe e poderia ser facilitado se houvesse um sistema que ajudasse a automatizar essa planilha e como Sommerville (2007) diz, a utilização de *softwares* com critérios adequados tornam-se requisitos obrigatórios para as empresas. Com a implantação da aplicação propõe um maior controle sobre o envio/status das propostas criadas diariamente além da possibilidade do reenvio das propostas de maneira simples e prática. Para o cliente, isto facilitaria de modo que ao solicitar uma máquina o mesmo receba instantaneamente o documento com os dados da sua proposta efetuada, facilitando uma futura transação na confirmação da venda, além de um histórico de todas as ações dessa área.

1.1 Justificativa

Após uma análise, foi verificado que havia uma necessidade de se automatizar um processo de elaboração e geração de propostas que anteriormente eram realizados através de formas manuais, no caso de planilhas. Essa situação mostrou uma oportunidade para a criação de uma aplicação *web*, está na qual é de fácil acesso a todos, para levar uma maior facilidade para a elaboração destas propostas, possibilitando a inserção/edição de seus dados de maneira dinâmica.

1.2 Objetivo Geral

Oferecer funções de controle, gestão de vendas e geração de propostas para automatizar serviços de empresas que utilizam de planilhas para o seu controle interno. Estas características deverão ser levadas ao usuário com interfaces simples e com fácil usabilidade por parte do sistema.

1.3 Objetivo Específico

O sistema deverá oferecer uma ferramenta para a elaboração de propostas, possibilitando a geração de um arquivo PDF sobre a mesma. Ele também deverá oferecer ao usuário a possibilidade de envio de e-mail com a proposta elaborada. A aplicação também oferecerá a possibilidade do cadastro de todas as etapas até a geração da proposta, como o cadastro dos clientes, cadastros das taxas, cadastro das máquinas e seus opcionais.

1.4 Trabalhos Correlatos

Afim de obter informações e aprimorar a aplicação, foi levantado um sistema no qual possui semelhanças com a proposta da aplicação desenvolvida, de forma que valide seus requisitos, êxitos e principais preocupações.

1.4.1 DocSales

O DocSales é um sistema de criação de documentos, o qual inclui propostas, vendas e cotações. Ele realiza a integração com outros ERP's de forma que com ele é possível realizar a assinatura digital de todos os documentos gerados. A limitação do DocSales se dá na edição das propostas geradas.

2. Materiais e Métodos

Nesta seção, serão apresentados os materiais e métodos que foram utilizados no desenvolvimento deste projeto, como: a metodologia utilizada, o modelo de desenvolvimento incremental e as ferramentas utilizadas para o desenvolvimento. Para melhor entendimento, cada conceito foi dividido em subseções.

2.1 Modelo de processo incremental

Para o processo de desenvolvimento foi definido o Modelo de Desenvolvimento Incremental. Assim como dito por Zuliani (2015) "O desenvolvimento incremental é justamente ir adicionando novas partes ao longo do tempo. É como adicionar tijolos em um muro na sua construção, depois de vários incrementos (tijolos) você tem o muro pronto." Ou seja, começar por um esboço ou descrição inicial do projeto, onde terá como ponto central os ciclos de especificação, desenvolvimento e validação. A cada ciclo é criada uma nova versão ou, como representado por Zuliani (2015), é adicionado um novo tijolo.

Na fase de especificação serão levantadas todas as necessidades do cliente ou dos usuários. Após isso, deve-se partir para o desenvolvimento e implementação do que foi requerido e assim finalizar o processo, validando através de testes tudo o que foi produzido. Como, pode ser visto na Figura 1, a cada ciclo de especificação, desenvolvimento e validação será gerada uma determinada versão.

Versão Inicial: A primeira versão do produto.

Versões intermediárias: São aquelas onde houveram novas implementações ou realização de ajustes.

Versão final: A última versão do produto, ou seja, o produto pronto.

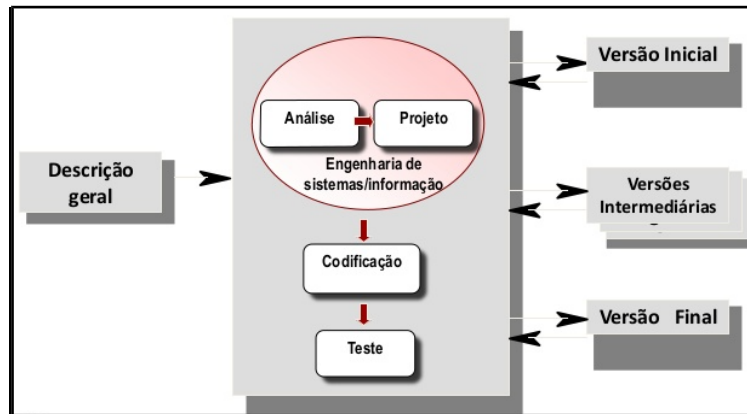


Figura 1. Exemplo de Modelo Incremental.

2.2 Metodologia

A primeira atividade é levantar os requisitos dos usuários para conhecer o escopo esperado do projeto. Para tal será levantado junto a uma empresa do ramo, as necessidades que eles possuem atualmente para a construção da aplicação.

Todas as etapas da proposta seguirão o desenvolvimento incremental.

Após delimitado o escopo do trabalho, as funcionalidades serão priorizadas e o desenvolvimento será feito.

2.3 Sistema Web

No decorrer dos últimos anos com o crescimento da população e conseqüentemente das empresas, houve uma maior demanda para utilização e desenvolvimento de aplicações *web*. O aumento desta demanda deu-se também, por conta da necessidade de maior acesso à informação a partir de micro-empresendedores, e empresas no geral. Assim, a fim de, tornar as informações mais acessíveis as aplicações *web* se apresentam como grande alternativa, uma vez que, permite conexão a informação de qualquer lugar com acesso à Internet ou até mesmo em redes locais isoladas.

Foram escolhidas as tecnologias (React, HTML, CSS e Javascript) para o desenvolvimento do *frontend* devido a sua facilidade de uso, e que não requer experiência avançada para poder produzir alguma coisa, pois existem inúmeros *sites*

com códigos já prontos para serem utilizados, ajudando diversas pessoas a construírem de maneiras diferentes seus projetos.

Já para o *backend* foi utilizado do ASP.NET Core para a criação das api's que irão ser consumidas pelo sistema *web*.

2.3.1 HTML

Para desenvolvimento da aplicação *web*, uma das linguagens utilizadas foi o HTML (HyperText Mark-up Language), que é uma linguagem de marcação de Hipertexto e a base para a criação e desenvolvimento de *sites*. Ela permitirá o desenvolvimento da aplicação na parte de criação de páginas e opções de customizações para a nossa aplicação.

Segundo Neves (2004) o HTML é uma linguagem de formatação que possibilita criar programas com a extensão em htm ou html, contendo *tags* (marcas) referenciadas no documento que será interpretada pelo navegador *Web*, na forma de texto, imagens ou suportes multimídia.

2.3.2 CSS

A parte de *design* deste sistema se dá a partir da utilização das chamadas folhas de estilo em cascata, a qual recebe o nome de CSS (Cascading Style Sheets). Ela serve para definir elementos e estilos de outra página composta com a sua, ajudando na organização e implementação de outras páginas conectadas ao CSS.

De acordo com Freeman e Freeman (2008), após ter um elemento Style no HTML é preciso o CSS para acrescentar um charme a sua página, e também irá ajudar na sua organização, permitindo quais elementos e estilos que serão utilizados para a aplicação e seletores que serão determinados para cada estilo de elemento.

2.3.3 JavaScript

Já o JavaScript é uma linguagem de programação desenvolvida para a internet, que permite adicionar interatividade para uma página e *design* criativo. O comando em JavaScript terá seu código criado dentro um programa para a construção de uma página *web* moderna, que será interpretado e executado pelo navegador (*browser*).

Para Michael Morrison (2008), o Javascript se coloca como parte da construção do HTML e o CSS para a construção de uma página *web* moderna. Para a criação de uma página o HTML seria a estrutura, CSS o estilo e o JavaScript só então agir como ação em cima da estrutura com base no estilo, dando vida à página.

2.3.4 React

Para o desenvolvimento do *frontend* do sistema, foi utilizado o React, que como seus criadores dizem é “uma biblioteca JavaScript declarativa, eficiente e flexível para a criação de interfaces de usuário (UI)”.

Por ser uma biblioteca de fácil utilização, o React segundo pesquisas realizadas pela STATISTICS AND DATA, baseadas no número de repositórios exportados do GitHub, foi o segundo *framework* mais utilizado do mundo de 2011 até 2020.

Ele é uma biblioteca orientada a componentes, a qual possibilita a separação das páginas em diversos setores, possibilitando assim a reutilização destes componentes nos mais diversos locais, respeitando suas características individuais.

2.4. Banco de Dados

Os bancos de dados servem para guardar informações desde o princípio e segundo Korth (1994), eles são uma coleção de dados inter-relacionados, representando informações sobre um domínio específico. Com a evolução digital esses métodos também evoluíram, e com isso surgiram os SGBDs (Sistema de Gerenciamento de Banco de Dados) que é um *software* capaz de interagir com o usuário possuindo diversos recursos para o gerenciamento do banco de dados, permitindo agrupar registros utilizáveis para um mesmo fim, sendo assim uma de suas vantagens é permitir o acesso aos dados de maneira simples e a possibilidade do acesso simultâneo de usuários a fim de inserir, remover e modificar informações do mesmo.

O MySQL é um gerenciador de banco de dados usado para armazenar informações, seu uso é gratuito e possui o código aberto, permitindo a maioria de aplicações para a criação da base de dados. A sua linguagem SQL (Structure Query Language – Linguagem de Consulta Estruturada) permite o acesso e o gerenciamento dos dados contidos em um banco de dados.

Iremos utilizar o MySQL por ele ser popular e também pela sua facilidade e simplicidade em relação a integração com o .NET Core. E de acordo com o Ranking DB-Engines, na data 01 de novembro 2020, o MySQL é o segundo mais usado para o gerenciamento de banco de dados no mundo. A Figura 2 mostra o diagrama entidade-relacionamento do projeto.

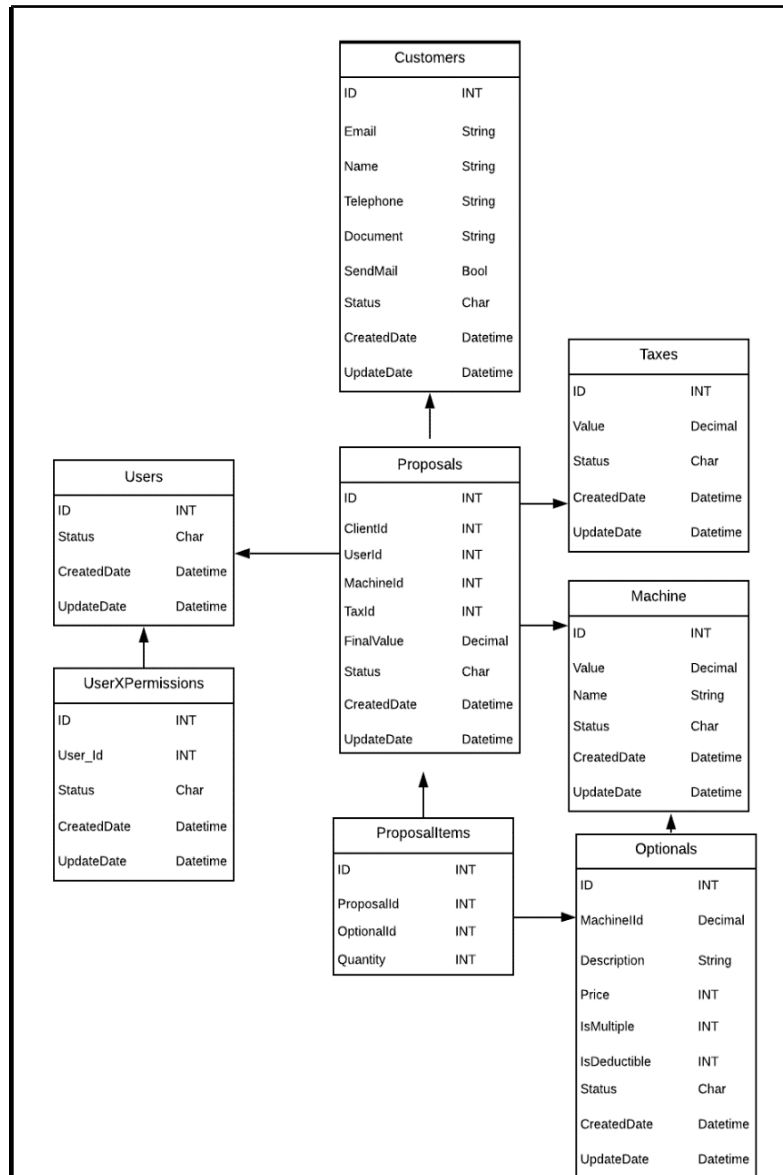


Figura 2. Diagrama entidade-relacionamento do banco de dados.

2.5. Programação orientada a objetos

Para o desenvolvimento do sistema, foi utilizado o conceito de programação orientada a objetos, esta na qual nos possibilita uma melhor estruturação e utilização das coisas relacionadas ao mundo real, sendo o “objeto” que compõe o nome do conceito algo genérico, que por sua vez, poderá representar algo tangível.

Segundo Kindler e Krivy (2011), na Programação Orientada a Objetos, programas de computadores são projetados por meio da composição de objetos que interagem com outros. Baseado nesta questão da interação dos objetos, foi utilizado desta interação adjunto a linguagem CSharp, para o desenvolvimento do *backend* do sistema desenvolvido neste trabalho.

2.6. ASP.NET Core

O Asp.net Core é um *framework* de código aberto, multiplataforma utilizado para desenvolvimento de aplicações *Web* utilizando o padrão de arquitetura MVC (Model-View-Controller).

O *framework* tem como característica principal, o suporte ao desenvolvimento de aplicações para Windows, Linux e Mac OS, o alto desempenho, além de possuir uma fácil integração com os *frameworks* que serão utilizados no projeto.

Para melhor esclarecimento sobre a arquitetura do *Framework*, (RAMOS, 2015) afirma que o "MVC é nada mais que um padrão de arquitetura de *software*, separando sua aplicação em 3 camadas. A camada de interação do usuário (*view*), a camada de manipulação dos dados (*model*) e a camada de controle (*controller*)."

2.7. Engenharia de Requisitos

Para analisar as funções nas quais a aplicação deveria possuir, foi realizado a análise de requisitos em cima de um estudo de caso a partir de uma empresa que realiza a venda de máquinas de produções, afim de identificar o que o *software* deveria possuir, como suas funcionalidades e requisitos, sendo eles os requisitos não funcionais e os funcionais.

Segundo Laudon e Laudon (2011), ela define cuidadosamente os objetivos do sistema modificado ou do novo sistema e desenvolve uma descrição detalhada das funções que o novo sistema deve desempenhar.

Os requisitos não funcionais de um sistema se referem a requisitos nos quais um sistema deverá possuir, mas não especificamente se referem a funcionalidades de negócio. A Tabela 1 mostra os requisitos não funcionais do projeto.

Requisitos Não-Funcionais	
RNF - 01	O <i>software</i> deverá ser uma aplicação <i>web</i> .
RNF - 02	Interfaces devem ser de fácil utilização, sendo claras e padronizadas.
RNF - 03	Garantir a persistência dos dados utilizando tecnologias de armazenamento de dados como um banco de dados.
RNF - 04	Autenticação do usuário para consumo da aplicação.

Tabela 1. Tabela de Requisitos Não-Funcionais da aplicação.

Os requisitos funcionais descrevem o comportamento de funcionalidades do sistema e especificações de tarefas e serviços aos usuários. A Tabela 2 mostra os requisitos funcionais do projeto.

Requisitos Funcionais	
RF01 - Efetuar Login	Acesso ao sistema com os dados do usuário cadastro previamente.
RF02 - Cadastrar Usuário	Criação de usuário de sistema afim de possibilitar o acesso a aplicação <i>web</i> .
RF03 - Edição e consulta de clientes	Possibilita o cadastro, a edição e a consulta dos dados dos clientes nos quais receberão as propostas geradas.
RF04 - Edição e consulta de máquinas	Será possível realizar a edição, cadastro e a consulta das máquinas que serão informadas durante a criação de uma proposta.
RF05 - Edição e consulta de taxas	Permitirá o cadastro, edição e consulta das taxas que poderão ser implementadas as propostas geradas.
RF06 - Criação de propostas	O sistema possibilitará a criação de propostas, utilizando as máquinas, clientes e taxas previamente cadastradas.

RF07 - Geração de pdf da proposta	Será possível gerar a proposta diretamente em um documento em formato de PDF, para ser consultado todos os dados da mesma.
RF08 - Envio de e-mail com a proposta	O sistema ao finalizar a elaboração de uma proposta possibilitará o envio de um e-mail contendo o pdf gerado da proposta para o cliente desejado.

Tabela 2. Tabela de Requisitos Funcionais da aplicação.

3. Desenvolvimento

Para o desenvolvimento a arquitetura do projeto foi dividida em três partes:

- **Frontend:** Ele é representado pela utilização do usuário ao sistema através de diferentes dispositivos.
- **Backend:** É representado através da *API REST* criada para a comunicação com o *frontend* e realização da persistência de dados com o banco de dados.
- **Persistência dos dados:** O banco de dados escolhido para persistir e armazenar os dados da aplicação na qual será acessado através da *API*.

A Figura 3 abaixo, mostra a comunicação entre os três elementos da arquitetura do projeto.

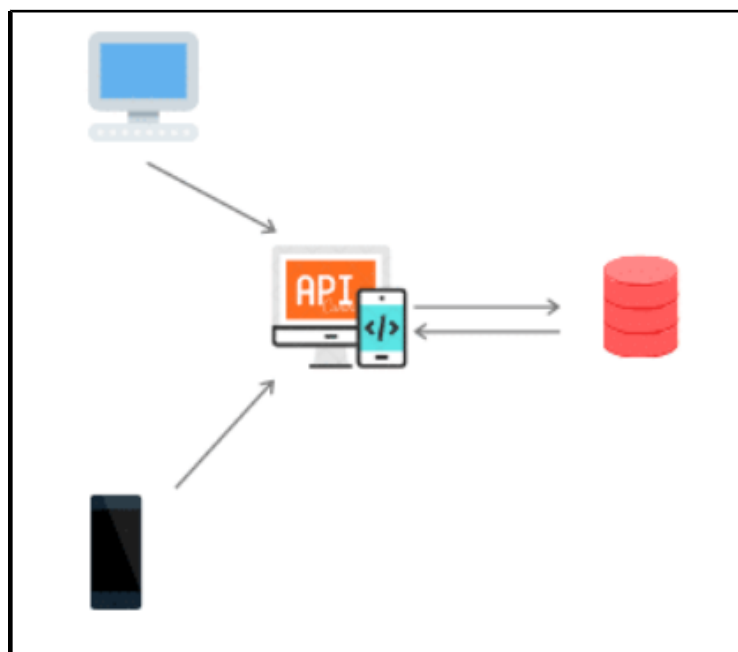


Figura 3. Representação da arquitetura do sistema.

3.1. Diagrama de Classes

De acordo com Booch, Jacobson e Rumbaugh (2000, p 94), o diagrama de classes mostra um conjunto de classes, interfaces e colaborações e seus relacionamentos. Os diagramas de classes são os diagramas mais encontrados em sistemas de modelagem orientados a objetos. Sendo assim, observa-se através do diagrama de classes elaborado para o projeto como funcionará o relacionamento entre as entidades criadas, conforme a Figura 4.

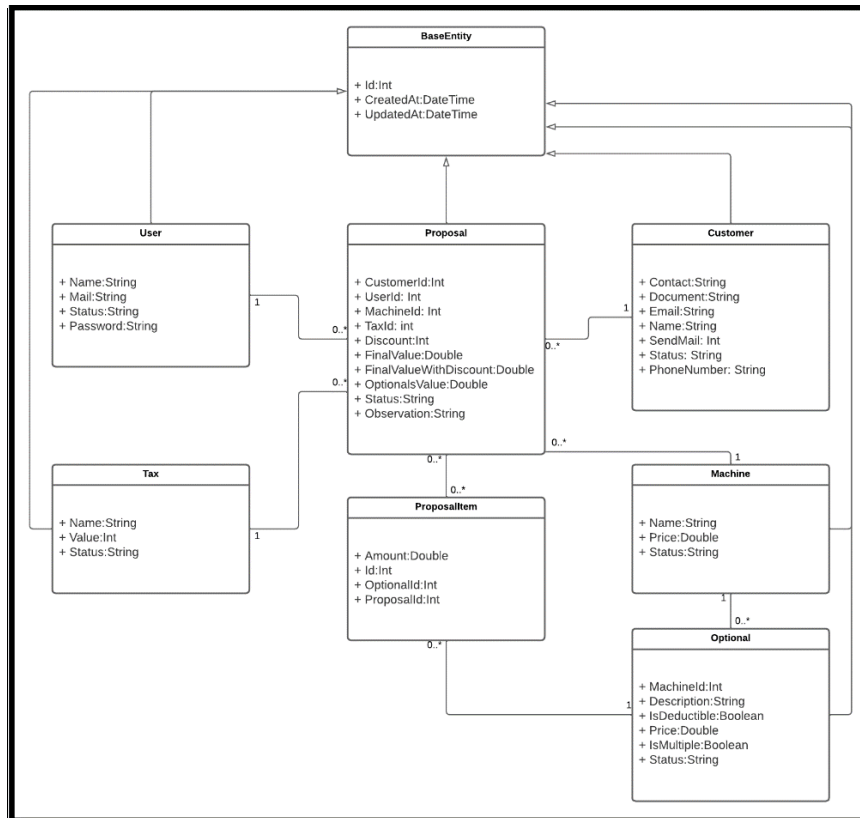


Figura 4. Diagrama de Classes.

3.2. Diagrama de Casos de Uso

O diagrama de casos de uso tem como função resumir os detalhes e interações de um usuário com o sistema. Ele faz parte da Linguagem de Modelagem Unificada (UML).

Segundo [Vazquez 2016], um caso de uso é um documento narrativo que descreve a sequência de eventos de um ator que usa um sistema para completar um processo. Um caso de uso representa uma unidade discreta da interação entre um ator (humano, dispositivo ou outro *software*) e o sistema.

A Tabela 3 mostra o detalhamento de uma ação que pertence ao caso de uso do sistema construído, de forma que foi especificado o caso de uso do processo de elaboração de uma proposta.

<p>Manter proposta</p> <p>Ator: Usuário do sistema</p> <p>Objetivo: Criar uma proposta de venda.</p> <p>Pré-Condições: O usuário deve estar logado e existir dados para o preenchimento da proposta, sendo eles o cadastro de máquinas, taxas e clientes.</p> <p>Fluxo principal:</p> <ul style="list-style-type: none">• O usuário deverá selecionar a opção de criação de proposta.• A página contará com os campos que deverão ser preenchidos para a criação da proposta.• O usuário deverá selecionar/preencher os campos de máquina, cliente, imposto, desconto e observação.• O sistema irá calcular automaticamente o valor final da máquina e mostrará ao usuário.• O usuário deve finalizar a elaboração da proposta <p>Fluxo alternativo: Não há.</p>
--

Tabela 3. Tabela Detalhamento de Caso de Uso.

Em relação as funcionalidades que o sistema irá conter, observa-se através do diagrama de casos de uso, que o sistema irá conter as funcionalidades de criação/edição de clientes, taxas e máquinas, assim como a emissão/envio das propostas geradas. A Figura 5 mostra o diagrama de casos de uso do sistema.

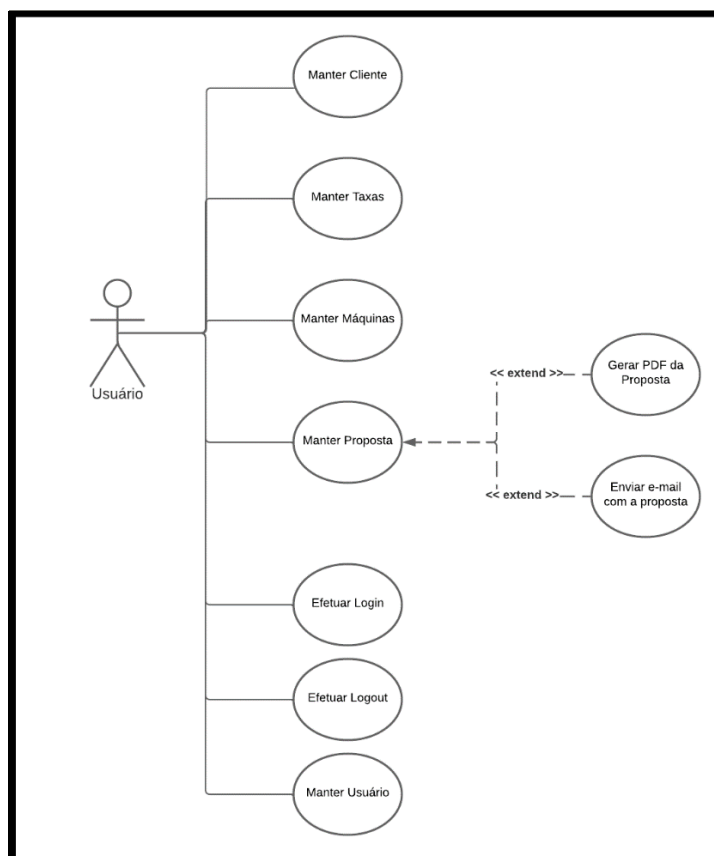


Figura 5: Diagrama de casos de uso.

3.3. Telas do sistema

Após a realização do levantamento de requisitos com pessoas que poderão utilizar o sistema, foi realizado o início da criação das telas e de suas funcionalidades, como poderá ser visto nas explicações e figuras abaixo, que mostram as telas criadas e suas funcionalidades.

As Figuras 6 e 7, representam as telas de criação/edição de clientes no sistema e a sua listagem, de modo que um cliente após ser criado poderá ter todos os seus dados editados, exceto o campo de identificador. Essa tela irá representar os clientes/empresas que poderão ter propostas de compras de máquinas, de modo que, será fornecido e-mail e o contato destes clientes para que ao ser gerada uma proposta para o mesmo, ser enviado o documento em pdf. O envio do e-mail após finalização da proposta é parametrizável de modo que apenas os clientes que possuem esta configuração ativa através do seletor “**Enviar email**” receberão o e-mail de proposta ao finalizá-las.

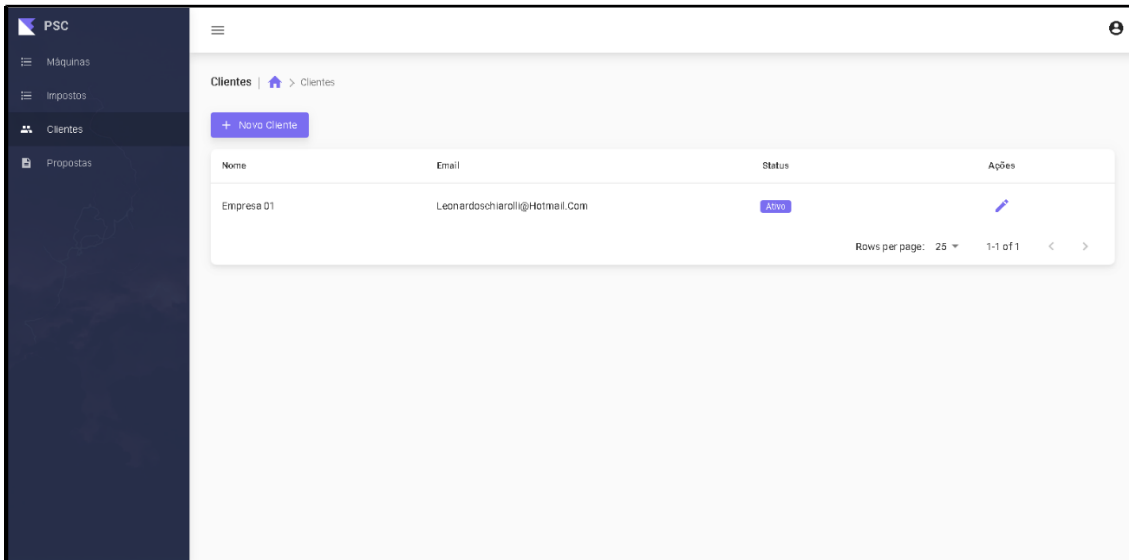


Figura 6. Consulta de clientes.

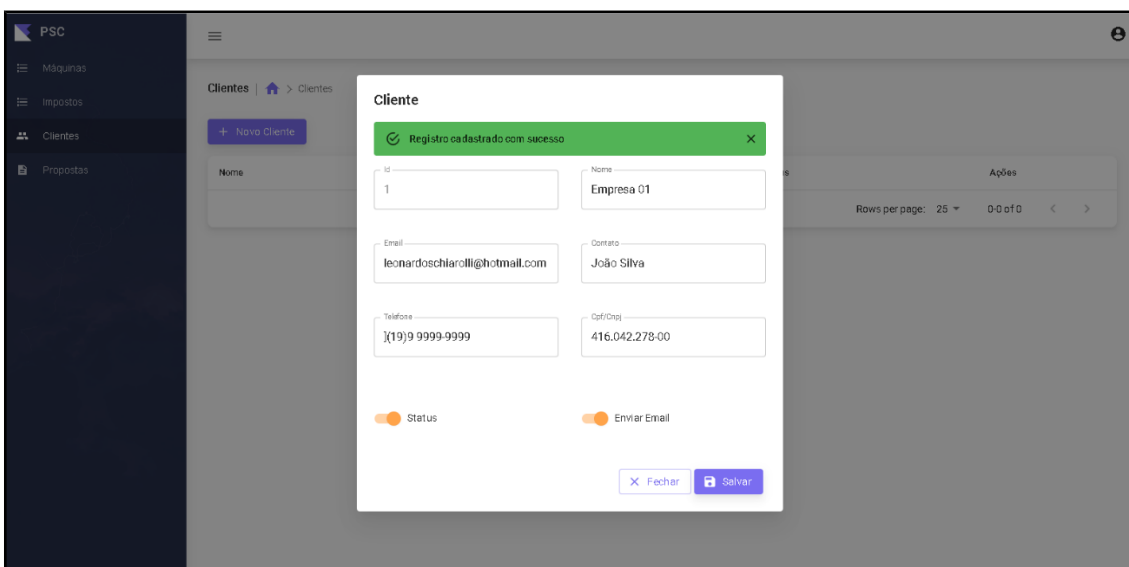


Figura 7. Cadastro/Edição de clientes.

As Figuras abaixo, 8 e 9, representam a criação/edição de máquinas que irão ser disponibilizadas para venda no sistema, assim como a consulta das mesmas. Em relação ao cadastro/edição de máquinas, as mesmas possuirão apenas as definições de nome e valor, assim como seu identificador, além de ter a opção da adição de opcionais de máquinas, que possuem a função de adicionar funções às máquinas com um adicional de preço.

Estes opcionais podem ser múltiplos ou não, além de possuírem a opção de serem dedutíveis ao valor da máquina. Para cada máquina poderão ser adicionados um ou mais adicionais, de modo que os mesmos não sejam obrigatórios.

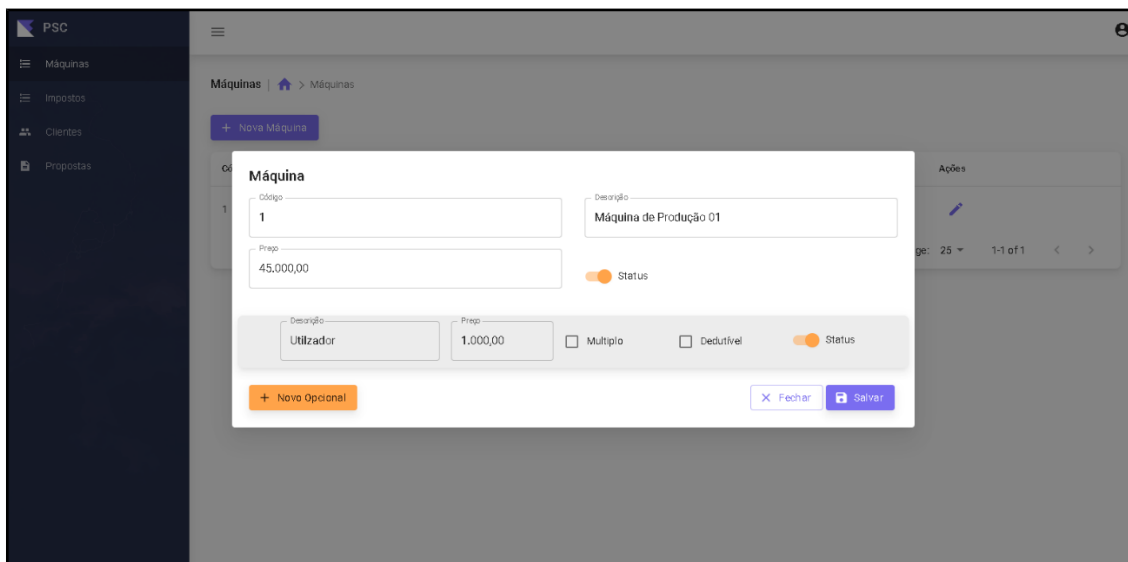


Figura 8. Cadastro/Edição de máquinas.

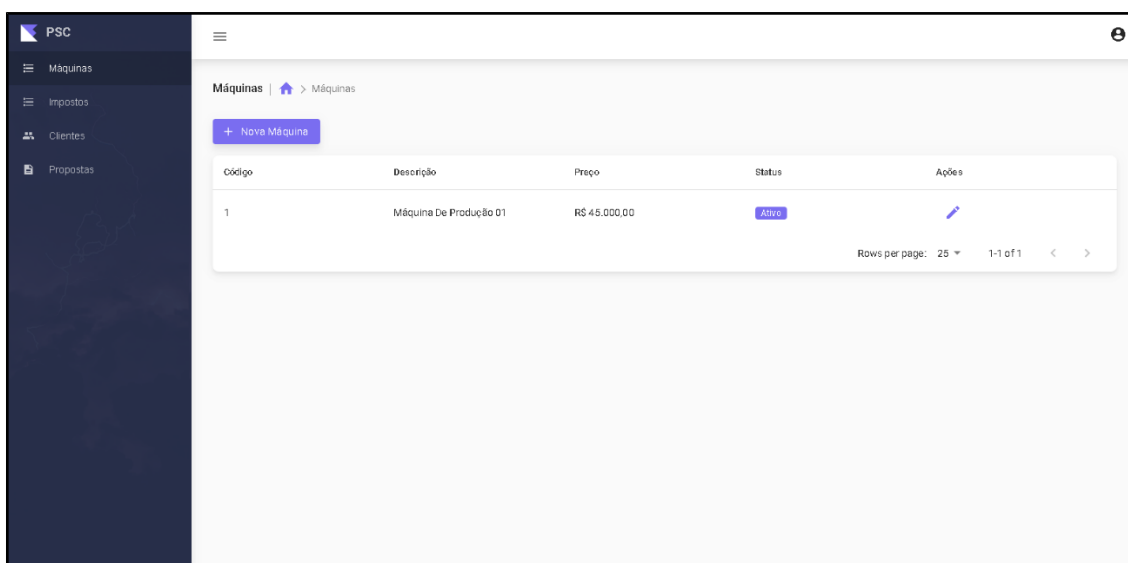


Figura 9. Consulta de máquinas.

As Figuras 10 e 11 abaixo, representam o cadastro/edição das porcentagens de taxas que poderão ser aplicadas a uma proposta, assim como a consulta das mesmas.

Essas taxas serão parametrizáveis, de forma que o usuário do sistema poderá cadastrar qualquer imposto necessário, adicionando o seu nome, o seu valor e o status

do imposto, já que caso o mesmo esteja inativo, não será possível utilizá-lo durante a criação da proposta.

Estes impostos ao serem adicionados em uma proposta terão o seu cálculo aplicado através da subtração da porcentagem cadastrada no imposto em cima do valor final da máquina.

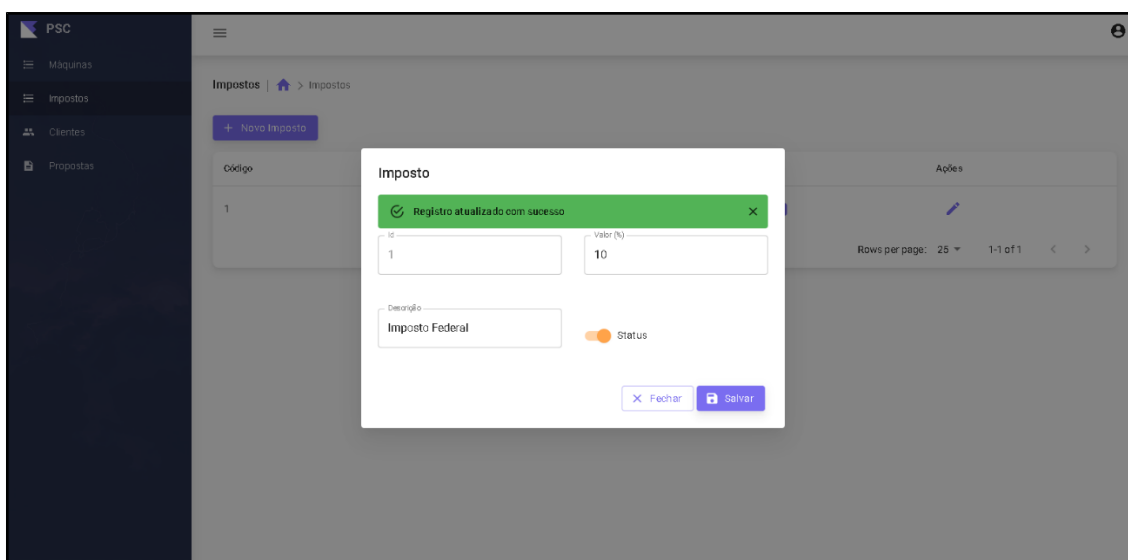


Figura 10. Cadastro/edição de porcentagem de taxa.

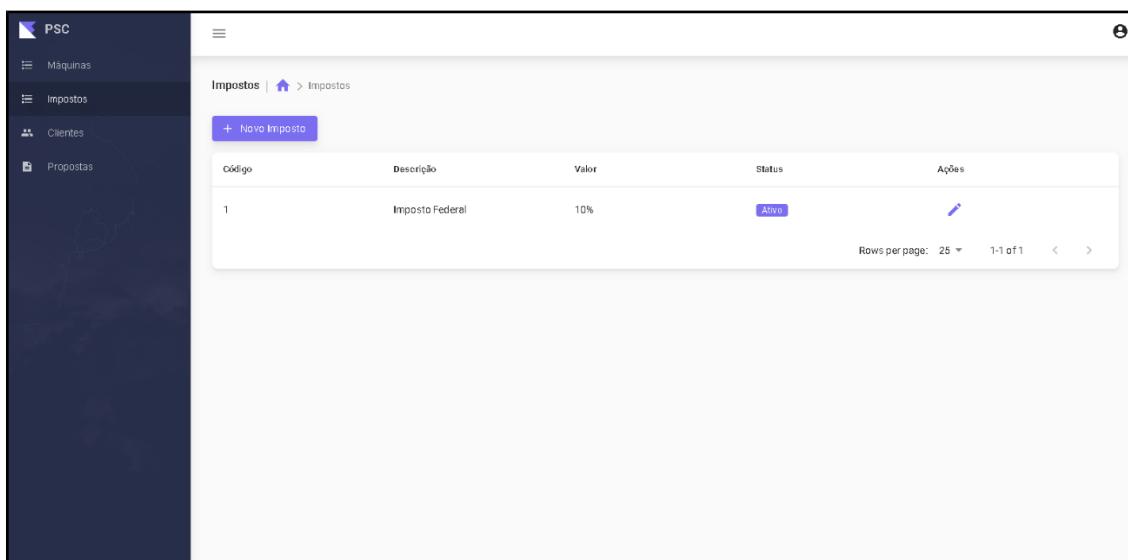


Figura 11. Consulta de taxas.

As Figuras abaixo, 12, 13, 14 e 15 representam as telas de criação/edição de propostas, a sua consulta e o documento pdf gerado ao realizar uma proposta.

A Figura 12 mostra o cadastro de uma proposta, está na qual deverá possuir para ser gerada alguns campos obrigatórios a serem preenchidos como o campo de cliente e de máquina. Os outros campos, como a opção de adicionar ou não uma porcentagem referente aos impostos cadastrados, observações referentes a compra da máquina e valor de desconto não serão obrigatórios. No lado direito da tela, também é mostrado/atualizado o valor final da proposta, de acordo com as opções selecionadas.

Figura 12. Cadastro de proposta.

A Figura 13, representa o resumo das informações da proposta gerada, de modo que a mesma contém dois botões, de envio de e-mail e de geração do pdf da proposta. O botão de envio de e-mail é parametrizável de modo que apenas os clientes que possuem esta função irão ter o mesmo disponível.

Figura 13. Resumo da proposta.

Na Figura 14, é apresentado o pdf da proposta que foi gerada, de modo que ela representa as informações que estão na tela de resumo da proposta. Esse pdf pode ser visualizado tanto pelo sistema, quanto pelo e-mail que é enviado ao cliente, caso o mesmo tenha esta opção habilitada em seu cadastro. Em relação a este documento, ele possui uma marca d'água com os dizeres "Documento não oficial", que é uma forma de representar que o documento não um contrato oficial.

20/03/2021		PSC		
Número Da Proposta:	Cliente:	Representante:	Máquina:	Data Da Proposta:
# 1	Empresa 01	Leonardo Schiarolli	1	20/03/2021
Observações:				
Entregar Até O Dia 20/03/2021				
Opcional	Valor base	Quantidade	Valor total	
Utilizador	R\$ 1.000,00	1	R\$ 1.000,00	
			Valor máquina:	R\$ 45.000,00
			Valor do imposto:	R\$ 5.111,11
			Total de opcionais:	R\$ 1.000,00
			Valor Total:	R\$ 51.111,11
			Valor com desconto:	R\$ 46.000,00

Figura 14. Proposta criada.

Por fim, a última tela, Figura 15, representa a consulta das propostas, sendo que nesta tela é possível para cada registro realizar o cancelamento da proposta ou a geração do pdf, através dos botões em forma de ícones.

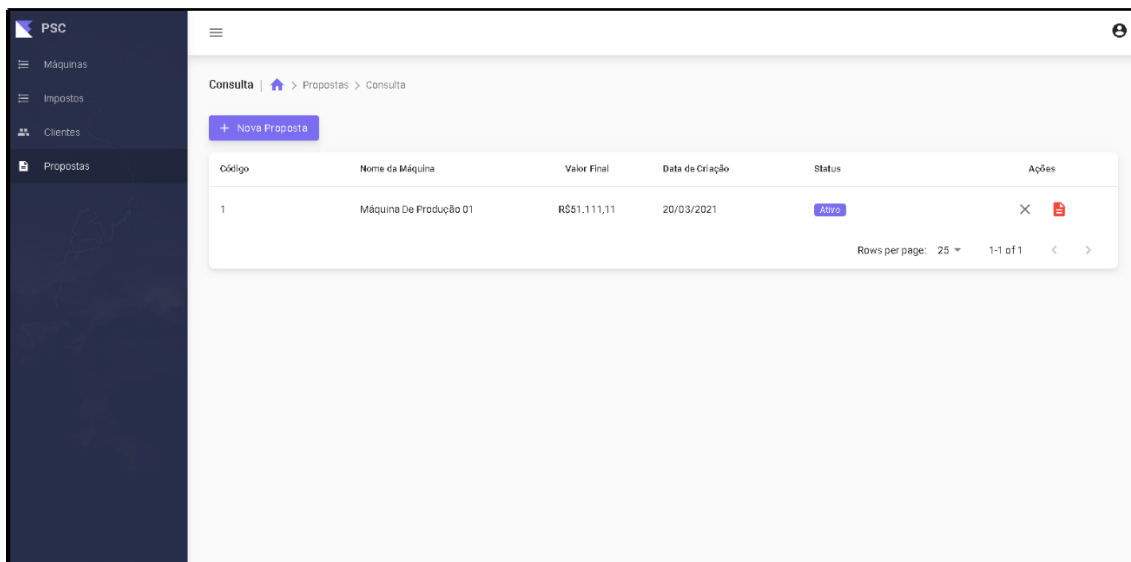


Figura 15. Consulta de propostas.

4. Resultados

Para realizar os testes da aplicação e validar o desenvolvimento, foi realizado uma pesquisa com três pessoas, sendo uma delas uma pessoa que trabalha com ferramentas administrativas e dois desenvolvedores para validar a experiência da aplicação.

Em relação aos testes, foi disponibilizado três *logins* para os três usuários, de forma que os mesmos executaram todos os fluxos da aplicação, sendo desde o cadastro de clientes até a finalização da elaboração da proposta e envio para o e-mail. Foi mostrado a eles e validado também, a padronização construída para todas as telas.

Por fim, foram idealizadas três perguntas em relação à experiência de utilização do sistema e também aos desenvolvedores sobre as tecnologias utilizadas na aplicação, conforme a Tabela 4 abaixo.

Usuários	Usaria para aprender novas tecnologias?	O sistema é de fácil utilização?	Você usaria o sistema novamente?
Pessoa 1	N/A	Sim	Sim
Analista 1	Provavelmente sim	Sim	Provavelmente sim
Analista 2	Provavelmente sim	Sim	Sim

Tabela 4. Tabela de testes de usuários.

5. Conclusão

O intuito deste desenvolvimento foi a construção de um sistema *web* que auxilie na geração de proposta de máquinas de produção. O sistema possui os recursos necessários para oferecer agilidade e facilidades para a elaboração destas propostas, além de gerar um controle em cima das propostas que estão sendo geradas pelos utilizadores.

Os itens de desenvolvimento que foram levantados durante a elicitação de requisitos e na construção do *backlog* foram implementados dentro do prazo estipulado.

Devido à necessidade deste campo em que houve o estudo e levantamento dos requisitos, é conclusivo que os objetivos apresentados foram atingidos, levando a geração de proposta ao usuário final.

Para o desenvolvimento deste trabalho e até mesmo destacando as competências e habilidades adquiridas no decorrer do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, houve aplicação de conceitos associados às disciplinas de Banco de Dados, Desenvolvimento *Web*, Algoritmos e Programação, Engenharia de *Software*, Linguagem de Programação, Análise Orientada a Objetos, Interação Humano-Computador, Metodologia de Pesquisa e Projeto de Sistemas para planejamento e desenvolvimento teórico-técnico. Outros conhecimentos necessários para o desenvolvimento do trabalho foram agregados através de pesquisa bibliográfica e estudo técnico por meio da ferramenta apresentada.

6. Trabalhos Futuros

Para futuras implementações no sistema criado, alguns assuntos como a implementação de funcionalidades para o controle das permissões das ações existentes no sistema, poderiam ser abordados, além da inclusão de um módulo para a gestão das vendas efetuadas.

7. Referências

BOOCH, G.; JACOBSON, I.; RUMBAUGH, J. UML: Guia do usuário. Rio de Janeiro: Campus, 2000.

FREEMAN, Elisabeth ; FREEMAN, Eric. Use a Cabeça!: HTML com CSS e XHTML. 2. ed. Rio de Janeiro: Alta Books, 2008.

KINDLER, E.; KRIVY, I. Object-Oriented Simulation of systems with sophisticated control. International Journal of General Systems, 2011.

- KORTH, H.F. e SILBERSCHATZ, A.; Sistemas de Bancos de Dados, Makron Books, 2a. edição revisada, 1994.
- LAUDON, Kenneth C.; LAUDON, Jane Price. Gerenciamento de sistemas de informação. 3. ed. Rio de Janeiro: LTC, 1999a. 433p
- LAUDON, Kenneth; LAUDON, Jane. Sistemas de Informação Gerenciais. 9. ed. São Paulo: Pearson, 2011.
- MORRISON, Michael. Use a Cabeça JavaScript. Rio de Janeiro: Alta Books, 2008.
- NEVES, Pedro M C. O GUIA PRÁTICO DA HTML. Lisboa: Centro Atlântico, 2004.
- RAMOS, Allan, O que é MVC? Disponível em: <<https://tableless.com.br/mvc-afinal-e-o-que/>>. Acesso em: 02 nov. 2020.
- SOMMERVILLE, I. Engenharia de Software. 8a Edição. Addison Wesley. 2007.
- STATISTICS AND DATA. The Most Popular JavaScript Frameworks – 2011/2021. Disponível em:<https://statisticsanddata.org/data/the-most-popular-javascript-frameworks-2011-2021/#pll_switcher/>. Acesso em: 28 de fev. 2022.
- VAZQUEZ, C. E. Engenharia de Requisitos: software orientado ao negócio: Brassport, 2016.
- ZULIANI, Emerson. Desenvolvimento iterativo e incremental. Disponível em: <<http://emersonzuliani.com.br/desenvolvimento-iterativo-e-incremental/>>. Acesso em: 02 nov. 2020.

Documento Digitalizado Público

Artigo de TCC do aluno Leonardo Schiarolli

Assunto: Artigo de TCC do aluno Leonardo Schiarolli
Assinado por: Daiane Tomazeti
Tipo do Documento: Comprovante
Situação: Finalizado
Nível de Acesso: Público
Tipo do Conferência: Cópia Simples

Documento assinado eletronicamente por:

- **Daiane Mastrangelo Tomazeti, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 13/06/2022 19:59:00.

Este documento foi armazenado no SUAP em 13/06/2022. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifsp.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1006782

Código de Autenticação: 973b2eba2e

