

MODELAGEM DE SISTEMA DE INFORMAÇÃO PARA ESCRITÓRIO CONTÁBIL

ÉVELIN CORSINI DE MARTINI¹, MATEUS DOMINGOS DOS REIS², JACIARA
SILVA CAROSIA³

¹ Discente do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas /
evelin.martini@fatec.sp.gov.br

² Discente do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas /
mateus.reis03@fatec.sp.gov.br

³ Docente do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas /
jaciara.carosia@fatec.sp.gov.br

RESUMO

O objetivo deste trabalho é a elaboração da modelagem de um sistema para suprir as necessidades de um escritório contábil onde seu desenvolvimento visa eliminar problemas que são encontrados no dia a dia da empresa, facilitando e agilizando tarefas que atualmente custam tempo e deslocamento tanto de clientes quanto de funcionários, além de servir como um meio de publicações de notícias e atualidades. Com essa demanda houve a oportunidade para a criação de um sistema que contenha esses benefícios, onde o cliente com poucos cliques possa solicitar serviços oferecidos pela empresa de maneira intuitiva e segura, assim o escritório podendo requisitar todos os documentos e informações necessárias para o cliente construindo um ambiente com uma relação mútua. Para a criação do projeto de maneira a que atendesse melhor às necessidades exigidas, foram utilizadas técnicas de coleta de requisitos feita junto ao gestor da empresa, para o desenvolvimento da análise dos requisitos bem como a elaboração da modelagem do referido sistema de informação. **Palavras-chave:** Modelagem; Engenharia de Requisitos; UML; Prototipação; Sistema de Informação.

1 INTRODUÇÃO

A pandemia causou grandes impactos desde seu início, transformando a maneira como as pessoas utilizam diversos serviços e criando novos serviços, gerando um avanço tecnológico para suprir as necessidades que apareceram no seu decorrer. A facilidade e agilidade foram pontos que tiveram enormes destaques, buscando sempre a comodidade dos clientes e usuários. Mesmo após os impactos da pandemia essa demanda por tecnologia ainda continua fazendo com que, as empresas que não possuam tais requisitos, fiquem para trás nessa corrida perdendo assim clientes e deixando a desejar em seus serviços.

Com essa perspectiva, o trabalho tem como objetivo elaborar a proposta de um Sistema de Informação, através da modelagem de um site para um Escritório Contábil. Utilizando conhecimentos e técnicas da Tecnologia da Informação, com foco na facilidade e agilidade dos processos, o presente trabalho busca melhorar o atendimento aos clientes, que muitas vezes precisam se deslocar até o escritório para fazer pedidos, entregar ou buscar documentos ou até mesmo o deslocamento de funcionários do escritório para esse fim.

2 REFERENCIAL TEÓRICO

2.1 Sistema de Informação

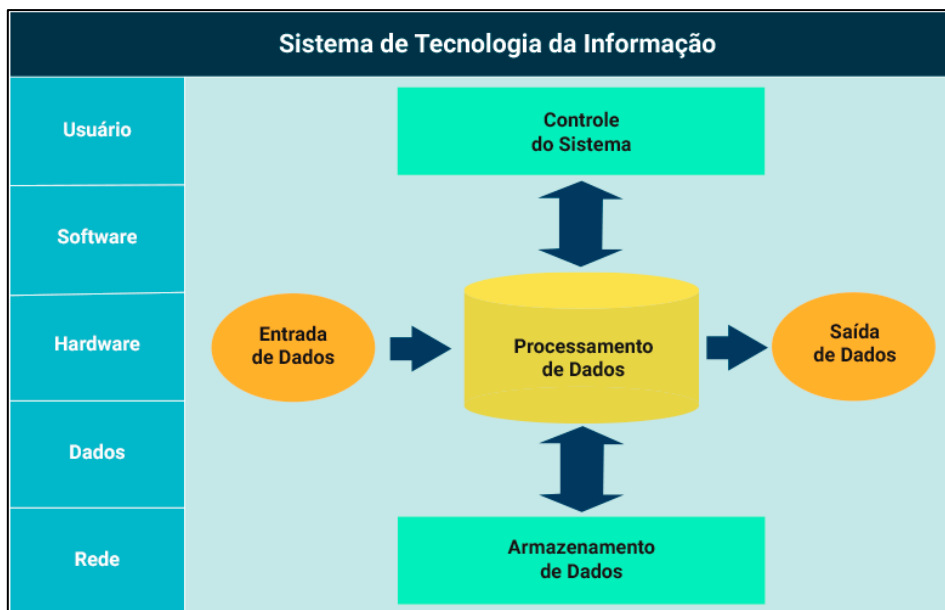
Gil (2010) define que os Sistemas de Informação se referem a uma reunião de recursos humanos, materiais, tecnológicos e financeiros ligados de acordo com uma sequência lógica para o processamento dos dados e a correspondente tradução em informações.

Com essa descrição pode-se afirmar que Sistemas de Informação são sistemas inter-relacionados, ligando softwares, hardwares e pessoas, podendo ser automatizados ou manual. Eles coletam dados do dia a dia da empresa, guardam e processam as informações mais relevantes, ajudando os administradores a verificar a situação em que a empresa se encontra para estarem preparados para situações inesperadas.

O'Brien (2011) criou uma adaptação de um conceito mais robusto dos Sistemas de Informação que recebe recursos de dados como entrada e os processa em produtos de informação como saída, sendo composto por componentes básicos para o seu funcionamento, que são hardware, software, dados e redes para realizar as ações de entrada, processos, produção, armazenagem e controle dos resultados obtidos chamados de informações, também sendo necessário para todo o processo a interação humana.

A imagem a seguir mostra os componentes dos Sistemas de Informações e suas interações.

Figura 1 - Componentes do Sistema de Informação



Fonte: Gadelha (2021)

Basicamente na entrada possui a captura de dados, podendo ser eles brutos ou informações já consolidadas para agregar mais aos resultados. No processamento é onde se transforma os dados em informações ou onde haja a junção de informações. Na saída acontece a distribuição dos resultados do processamento gerando novas informações que serão guardadas no armazenamento de dados, podendo ser consultadas ou utilizadas em novos processos. Já no controle do sistema é onde o usuário pode observar as informações que foram geradas pelo processamento, podendo assim ter um feedback do processo e fazendo ajustes para melhorar o seu funcionamento, gerando a possibilidade de um resultado mais aprimorado.

Para que se obtenha um Sistema de Informação eficiente é necessário um foco no seu processo de desenvolvimento, o que pode ser obtido através das boas práticas de Engenharia de Software.

2.2 Engenharia de Software

Segundo Pressman (2016), a Engenharia de Software é um conjunto de processos, métodos e ferramentas que possibilitam a criação de um sistema de software focando na alta qualidade, mesmo que os profissionais se encontrem em situações mais complexas e caóticas, tudo isso graças ao atendimento aos requisitos.

Sommerville (2011), afirma ainda que a Engenharia de Software pode ser comparada a um processo evolutivo acompanhando toda a vida desde todo o processo de criação da coleta de todas as características do sistema, o seu desenvolvimento, testes e sua implantação.

Sendo assim a Engenharia de Software visa a organização de todo o processo de criação de um software, desde a especificação até os testes para a implantação, coordenando todo o trabalho para que o profissional possa documentar, projetar, implementar e manter os sistemas de software ativos podendo garantir o máximo de seu desempenho e produtividade.

Com o uso da Engenharia de Software no desenvolvimento de sistemas busca-se garantir certo nível de qualidade no produto final, ou seja, no software, pois o problema é dividido em partes e tratado por um número de especialistas em cada uma das atividades do seu ciclo de vida.

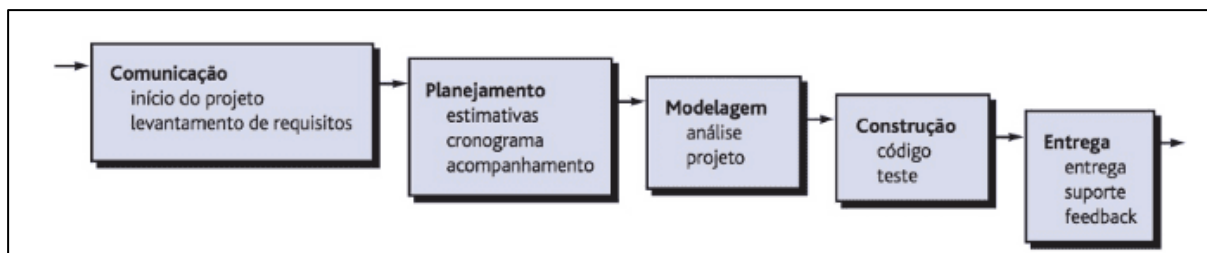
2.3 Ciclo de Vida

Cada software criado pode ser considerado único, já que cada empresa possui uma necessidade diferente da outra, com isso o processo de desenvolvimento também necessita de ser diferente em alguns casos. Com isso houve o surgimento de variados modelos de desenvolvimento de software onde cada um supre uma necessidade, tendo assim diferentes ciclos de vida, métodos de testes e análises, focos em diferentes categorias como segurança, manutenção, agilidade.

O Modelo Cascata, também conhecido como Ciclo de Vida Clássico, utiliza uma abordagem sequencial recomendada quando se possui os requisitos dos problemas bem compreendidos.

Ele se inicia com a coleta de requisitos junto ao cliente, após essa etapa é definido o planejamento com estimativas junto a criação de um cronograma. Na próxima etapa há a modelagem onde é feita toda a análise e projeto do sistema. Em seguida a construção verdadeira do software que é codificado, nessa parte também são realizados os testes do mesmo. Por último é a entrega para o cliente a partir dessa etapa é captado os feedbacks do cliente e dado o suporte.

Figura 2 – Ciclo de vida de Sistema de Informação



Fonte: Pressman (2016)

Segundo Paula Filho (2011) o Modelo Cascata possui alguns problemas principalmente para os dias atuais, como por ser um projeto linear alterações podem criar grandes problemas e confusões, além de muitas vezes em projetos reais não serem seguidas as sequências propostas pelo modelo.

No entanto, independente do modelo de processo utilizado no desenvolvimento do software, é fundamental o foco na coleta, análise, projeto dos requisitos, para que os mesmos sejam inseridos no produto a ser entregue.

2.4 Engenharia de Requisitos

A Engenharia de Requisitos ou análise de requisitos é considerada um processo que estuda os problemas que o cliente possui, buscando a melhor maneira de transformar essas necessidades em requisitos para o sistema (IEEE, 1990).

No desenvolvimento de software os requisitos são a base para o planejamento do projeto onde guiam seu andamento de acordo com as necessidades do cliente. Por isso, a Engenharia de Requisitos é extremamente importante, contribuindo na documentação do projeto, certificando que todos os requisitos sejam cumpridos e também faz com que os profissionais da área se mantenham organizados melhorando seu desempenho durante as fases do projeto. Além disso facilita a manutenção do sistema e diminui possíveis retrabalhos futuros causados por uma má elaboração do projeto em suas fases iniciais. Assim tem os dois tipos principais de requisitos:

Os Requisitos Funcionais são todos os problemas e necessidades que deverão ser resolvidos ou atendidos pelo sistema, ou seja, são as funcionalidades e serviços que vão definir a maneira que o sistema deve agir, assim ele foca no que deve ser feito.

Os Requisitos não Funcionais definirão as propriedades do sistema, como características do projeto final (tempo de resposta, confiabilidade, portabilidade, segurança, etc), características organizacionais (compatibilidade com sistema operacional, criptografia, linguagem de programação) e características externas (localização, legislação e política de proteção de dados), assim ele foca em como será feito (PAULA FILHO, 2010).

Segundo Pressman (2016) o processo de Engenharia de Requisitos pode ser dividido em sete etapas, sendo elas:

Concepção: é a identificação dos interessados pelo projeto (*stakeholders*) e quando se tem a primeira visão geral do sistema identificando as necessidades dos mesmos;

Elicitação: nesta etapa serão levantados os requisitos do sistema, sendo que deverão ter duas perspectivas, a categoria do requisito (requisitos funcionais e não funcionais) e a natureza do requisito, divididos por requisitos subconsciente, consciente e inconsciente demonstrados pelo cliente;

Elaboração: detalhamos os requisitos em modelos conceituais como o UML com o objetivo de deixá-los os mais claros possíveis diminuindo a chance de erros de entendimento;

Negociação: identificamos todos os conflitos entre os requisitos buscando uma negociação com o stakeholder para eliminar, combinar ou modificar os requisitos;

Especificação: nesta etapa o sistema é especificado em termos técnicos os requisitos, passando da visão do problema para a solução;

Validação: como o próprio nome diz é confirmado se realmente os requisitos atendem as necessidades propostas;

Gerenciamento: essa etapa permeia todo o ciclo de vida de desenvolvimento certificando que os requisitos sejam atendidos e em caso de mudanças ele possa mostrar os impactos que elas podem gerar.

2.5 Unified Modeling Language (UML)

Para Melo (2010), UML é uma linguagem para especificação, visualização, construção e documentação de artefatos de sistemas de software. Segundo Booch, Rumbaugh e Jacobson (2005) a UML é uma linguagem padrão para a elaboração da

estrutura de projetos de softwares, contudo é apenas a parte de um dos métodos utilizados para a criação de software.

Assim, UML auxilia o levantamento de requisitos oferecendo diversos recursos de modelagem de estruturas, e diagramas onde consegue abranger variados tipos de software, além disso vale ressaltar que ele não é preso a nenhuma tecnologia ou plataforma com vantagem de o padrão ver voltado para a orientação a objetos.

Segundo Booch, Rumbaugh e Jacobson (2005) os diagramas são utilizados para a visualização de blocos de construção básicos, como classes, interfaces, colaborações, componentes, nós, dependências, generalizações e associações com a finalidade de ajudar na compreensão do sistema.

Dentre os diagramas UML há os diagramas estruturais, sendo eles: Diagrama de classes, Diagrama de objetos, Diagrama de componentes, Diagrama de estruturas compostas e o Diagrama de implantação, estes que ajudam na visualização de partes estáticas. Há também os diagramas que ajudam na visualização de partes dinâmicas que são os diagramas comportamentais, sendo eles: Diagrama de caso de uso, Diagrama de sequências, Diagrama de comunicação, Diagrama de gráfico de estados, Diagrama de atividades.

O UML passou por diversas versões, cada uma delas trazendo novidades muito importantes, atualmente a última versão lançada foi a UML 2.5, segundo OMG (2013) lançada em outubro de 2012.

2.6 Prototipação

Segundo Pressman (2011), a prototipação é uma técnica utilizada pela área da Engenharia de Software onde a preocupação está em como o software será desenvolvido, compreendendo os requisitos do sistema e as funcionalidades necessárias e como elas serão apresentadas.

A prototipação é muito utilizada no início do processo de desenvolvimento do sistema, sendo uma maneira muito eficiente para conversar com o cliente e tentar entender suas necessidades. A prototipação pode ser feita através de desenhos em um papel ou até mesmo por softwares que mostram em tempo real a interface para o cliente, podendo mostrar diversas ideias e características de funcionamento que podem ser desconhecidas por ele.

Segundo Sommerville (2011) existem três tipos de prototipação. A prototipação de baixa fidelidade não é parecida com o produto final, na maioria das vezes são feitos esboços no papel e lápis. Há também a prototipação de média fidelidade, onde softwares demonstram um design que se aproxima mais do sistema final, mas não são as mesmas ferramentas que o desenvolvedor irá utilizar. Por último há a prototipação de alta fidelidade onde são utilizadas as mesmas tecnologias que são utilizadas podendo interagir com um sistema semelhante ao final, porém há um custo maior para a sua utilização além de conhecimento técnico.

2.7 Modelagem de Banco de Dados

Segundo Cougo (1997), pode-se considerar a modelagem de dados um processo para a criação de diagramas que consistem em simplificar a visão sobre um sistema muito complexo, muito parecido com o que ocorre na modelagem de classes, porém voltado apenas para o fluxo de dados, associando atributos de dados a entidades e com conceitos de relacionamento de tabelas.

Existem três estilos básicos para a modelagem de dados:

Modelo de dados conceituais: conhecidos como modelo de domínio onde define a estrutura de dados para o negócio, equipes ágeis o utilizam para o entendimento inicial dos requisitos do sistema explorando suas estruturas e conceitos, no desenvolvimento tradicional são usados como esboço para a criação dos modelos lógicos.

Modelo lógicos de dados: utilizados para explorar os conceitos apresentados pelo modelo de dados conceitual e seus relacionamentos, podendo ser usados para um projeto simples ou para uma empresa completa, servindo como base para a criação de um modelo de dados físico.

Modelo físico de dados: com ele se projeta o esquema inteiro de um banco de dados detalhando tabelas de dados, as colunas de dados das tabelas e o relacionamento entre as tabelas.

3 METODOLOGIA

O trabalho foi elaborado a partir de conhecimentos obtidos através dos ensinamentos do decorrer do curso, somando um levantamento bibliográfico através de pesquisas feitas em artigos acadêmicos, livros, revistas especializadas na área, visando sempre adquirir o máximo de informações sobre os temas abordados, os quais foram: Engenharia de Software, Engenharia de Requisitos, Modelagem de Sistemas, Técnicas de Prototipação e UML (*Unified Modeling Language*).

Tais aprendizados foram aplicados durante todo o processo da criação do artigo, iniciando no levantamento de requisitos visando entender as necessidades da empresa e como solucioná-las. Utilizando UML foi possível a criação de diagramas que fornecem múltiplas visões do sistema a ser construído, modelando camadas específicas para cada perspectiva, desde o cliente como um usuário do sistema até a equipe que planeja e desenvolve cada uma de suas funcionalidades.

Para a criação dos diagramas UML foi utilizado o software Draw.IO, um software de fácil utilização. Para a criação da modelagem de banco de dados foi utilizado o MySQL Workbench muito utilizado em sala de aula. Para a Prototipação foi utilizado o Figma que consegue suprir todas as demandas definidas, além disso todos os softwares podem ser utilizados gratuitamente liberados para diversas plataformas ou com seu uso sendo disponibilizado online.

4 RESULTADOS E DISCUSSÃO

4.1 Modelagem de Sistema de Informação para Escritório Contábil

O Escritório de Contábil aqui analisado busca por um Sistema de Informação pela necessidade de melhorar o atendimento a seus clientes, visando mais agilidade nos processos, ganhando mais tempo e economizando com deslocamentos (eventuais viagens ou combustíveis). Além disso é muito importante a organização que o Sistema de Informação irá trazer para o escritório, como ajuda para não perder de vista nenhuma solicitação e sempre estar ciente dos prazos de entrega das tarefas.

Outro ponto que vale ressaltar é a comodidade que o Sistema de Informação irá trazer ao cliente do escritório, já que será possível a este utilizar o sistema em

qualquer lugar em que houver Internet disponível, através de vários dispositivos e sistemas operacionais, sendo também sempre observado processos que visem deixar o sistema mais simples e intuitivo ao usuário, através de uma interface amigável.

4.2 Escopo do Sistema

O Sistema de Informação aqui proposto visa atender ao escritório de duas maneiras, a primeira uma página de notícias para o escritório colocar atualizações importantes de novas leis e novidades que possam ser interessantes para seus clientes e demais usuários, além disso irá mostrar quais serviços pode prestar aos seus clientes e suas principais especialidades. A segunda será uma área do cliente, restrita, onde o escritório disponibilizará uma senha e usuário genéricos a seu cliente que após logar poderá efetuar pedidos de serviços ao escritório com todos os requisitos e necessidades, e também enviar e receber documentos de maneira prática e simples.

4.3 Requisitos Funcionais

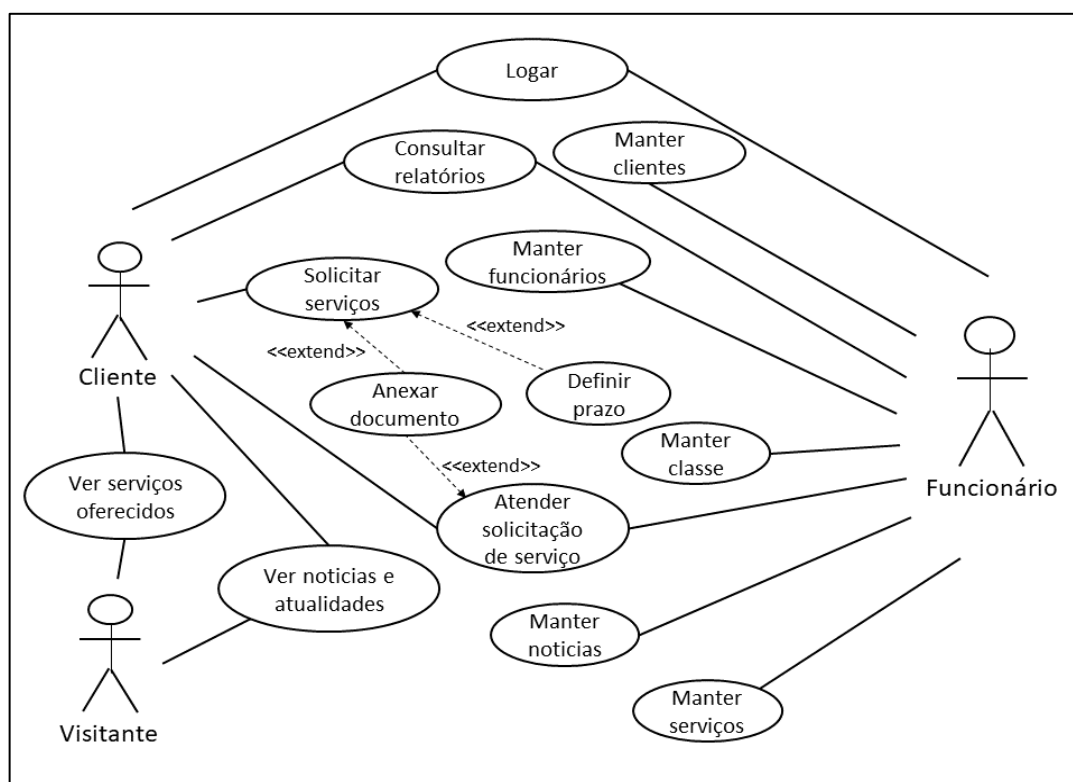
Os requisitos foram colhidos junto aos funcionários do escritório e com o acompanhamento de futuros usuários do sistema para que todas as especificações pudessem ser o mais próximo da realidade do escritório, diminuindo as chances de erros e de desvio de informações. Os requisitos levantados foram:

- Clientes e funcionários serão diferenciados no processo de acesso (login) ao site (após o login dependendo se o usuário for cliente ou funcionário terá um direcionamento a páginas com funções diferenciadas).
- Clientes e Visitantes poderão acessar o site para ver notícias e atualidades.
- Clientes e Visitantes poderão acessar o site para ver os serviços oferecidos pelo escritório.
- Funcionários poderão manter (cadastrar, consultar, alterar e excluir) as notícias.
- Funcionários poderão manter (cadastrar, consultar, alterar e excluir) serviços oferecidos.
- Funcionários poderão manter clientes (cadastrar, consultar, alterar e excluir).

- Funcionários poderão manter outros funcionários (cadastrar, consultar, alterar e excluir), definindo suas permissões conforme hierarquia organizacional e o departamento (classe) pertencente.
- Funcionários deverão manter as classes, isto é, os departamentos aos quais serão designados os serviços.
- Clientes poderão solicitar serviços pelo site (através da descrição dos serviços) e selecionar a classe responsável pela realização do serviço. O cliente poderá definir o prazo para entrega da sua solicitação que não poderá ser menor do que 24 horas. O cliente poderá anexar documentos a sua solicitação.
- Funcionários poderão atender as solicitações de serviços direcionados a sua classe. Podendo ver as solicitações feitas pelos clientes, executá-las, anexar documentos caso necessário e finalizar a solicitação.
- Funcionários e clientes poderão consultar relatórios mensais de serviços realizados.

De acordo com os requisitos descritos anteriormente, foi possível a criação de um diagrama de casos de uso que facilitasse o entendimento do sistema e sua maneira de distribuir as funcionalidades, como é mostrado na figura 3.

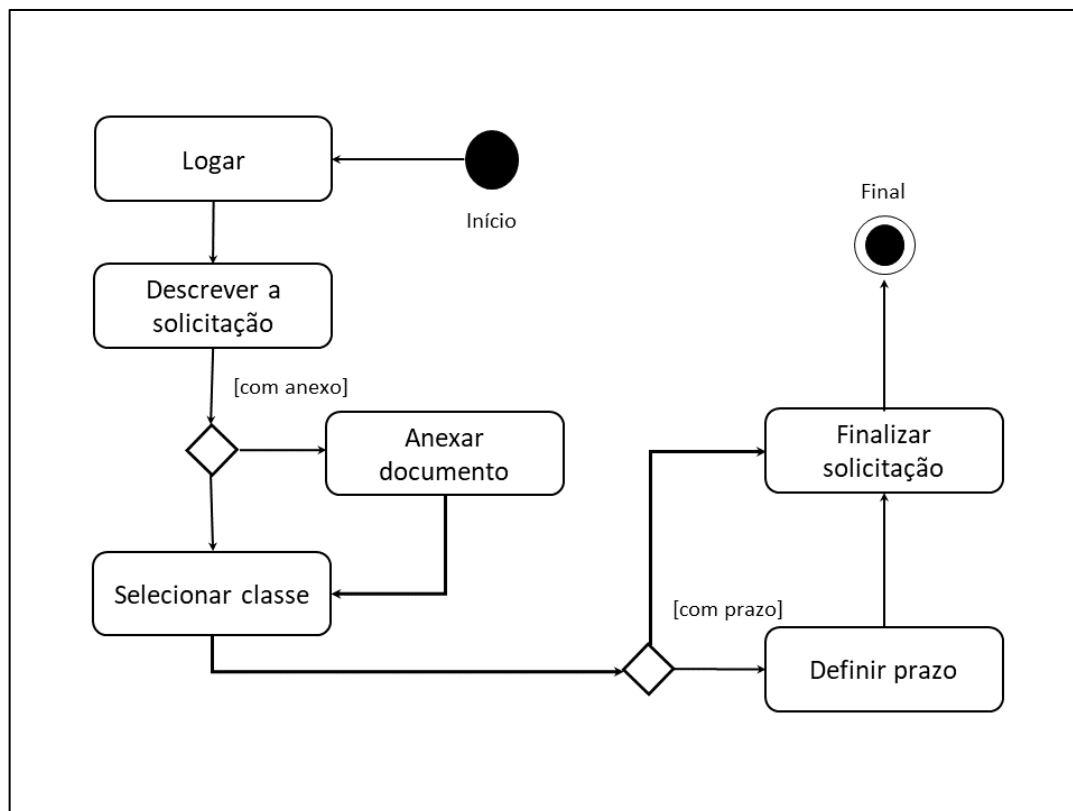
Figura 3 – Diagrama de casos de uso



Fonte: Autoria Própria (2022).

Com base no diagrama de casos de uso foi criado o diagrama de atividades, mostrado na figura 4. Esse diagrama demonstra a solicitação de um serviço por um cliente.

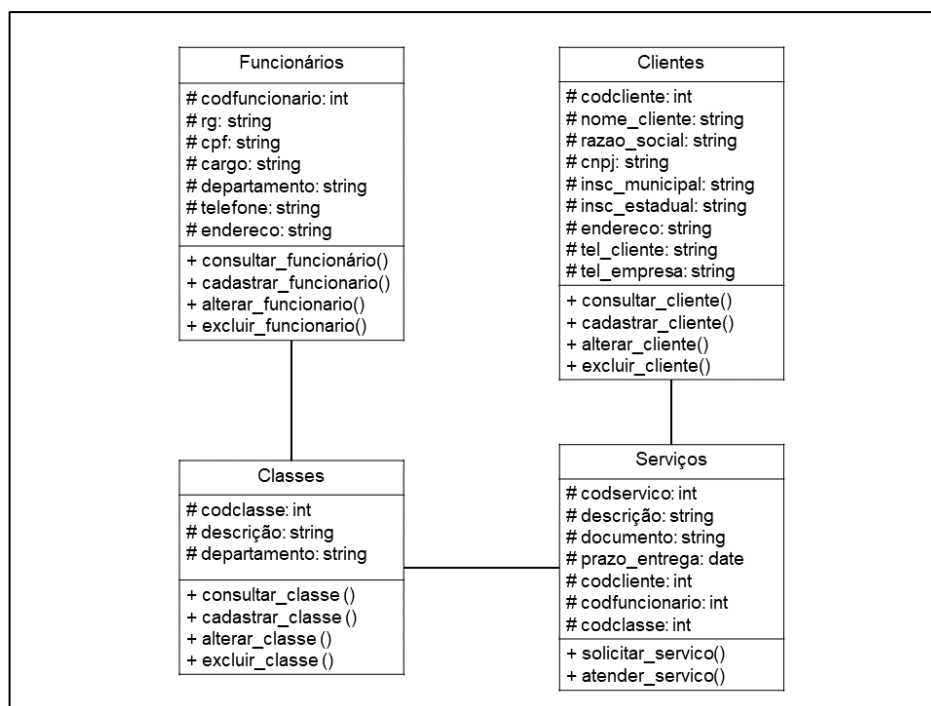
Figura 4 – Diagrama de Atividades



Fonte: Autoria Própria (2022).

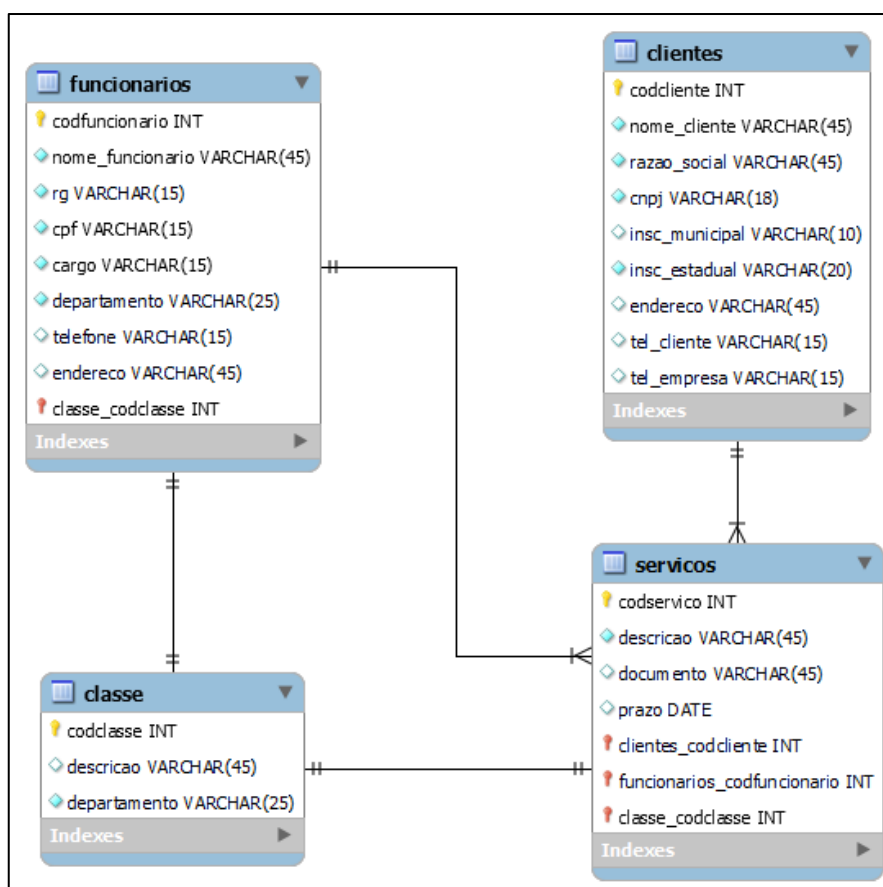
O diagrama de classes na figura 5 que demonstra as classes do sistema e o modelo de banco de dados para a criação de um banco de dados representado na figura 6.

Figura 5 – Diagrama de Classes.



Fonte: Autoria Própria (2022).

Figura 6 – Modelo de Dados



Fonte: Autoria Própria (2022).

4.4 Prototipação

Com o objetivo de aumentar a facilidade e a comunicação no momento do desenvolvimento da interface gráfica foram criados alguns protótipos de média fidelidade ilustrando as características gráficas que foram bem vistas pelos futuros usuários do sistema.

As figuras a seguir mostram alguns dos protótipos criados. O protótipo da página inicial do sistema é mostrado na figura 7, o modo de realizar o login na figura 8, o cadastro de funcionários na figura 9 e a solicitação de tarefas na figura 10.

Figura 7 - Protótipo da tela inicial.



Fonte: Autoria Própria (2022).

Figura 8 - Protótipo do login.

Escritório Contabil **Area do Cliente**

Usuario:
Senha:

Entrar

Area de Informações e Notificações

(19)0000-0000 Endereço email@email.com Horário de Atendimento

Fonte: Autoria Própria (2022).

Figura 9 - Protótipo do cadastro de funcionários.

Escritório Contabil **Usuario: nome usuario**

Area de cadastro e alterações do Funcionario

Buscar Cod funcionario: 0001

Nome:

RG: CPF:

Tel: Endereço:

Cargo: Departamento:

Cadastrar **Alterar** **Excluir** **Cancelar**

(19)0000-0000 Endereço email@email.com Horário de Atendimento

Fonte: Autoria Própria (2022).

Figura 10 - Protótipo de solicitação de tarefas.

Fonte: Autoria Própria (2022).

5 CONSIDERAÇÕES FINAIS

Um sistema de informação pode fazer falta em uma empresa, ajudando-a a superar obstáculos e a fazer uma melhor concorrência no mercado, destacando-a de seus concorrentes. A organização dos processos e das informações que o sistema pode oferecer á empresa é um ponto interessante, pois além de trazer mais produtividade e economia, pode gerar um melhor relacionamento entre os funcionários e entre a empresa e seus clientes.

Através da realização do presente trabalho, foi verificado que a modelagem gráfica auxilia muito no levantamento e documentação dos requisitos do sistema a ser desenvolvido. Permitindo uma melhor comunicação entre os envolvidos.

O entendimento dos processos empresariais foi crucial para poder realizar todas as etapas da modelagem do sistema e a modelagem facilitou o entendimento dos processos empresariais.

Com a documentação gerada durante esse trabalho, será possível a implementação do site, atendendo o máximo possível todas as necessidades e as expectativas da empresa.

REFERÊNCIAS

BOOCH, GRADY; JAMES RUMBAUGH; e IVAR, JACOBSON. UML: Guia do usuário. 6º Reimpressão. Rio de Janeiro: Elsevier, 2005.

COUGO, PAULO. Modelagem Conceitual e projeto de banco de dados. Rio de Janeiro: Elsevier, 1997.

GADELHA, A. Recursos e atividades de sistema de informação. 2021. Disponível em: <http://portaldoaluno.webaula.com.br/cursos_graduacao/funsis/aula2.html#0> acessado em 16 de abril de 2022.

GIL, Antonio Carlos. Como elaborar projetos de pesquisa. - 5. Ed. São Paulo: atlas, 2010.

GROUP, OBJECT MANAGEMENT. OMG. 2013. Disponível em: <<http://www.omg.org/spec/UML/2.5/Beta1/>> Acessado em 8 de março de 2022.

IEEE - Institute of Electrical and Electronics Engineers. Standards Glossary of Software Engineering Terminology: Std 610.12, N.Y., 1990.

MELO, ANA CRISTINA. Desenvolvendo aplicações com UML 2.2 do conceito à implementação. 3º edição. Rio de Janeiro: Brasport, 2010.

O'BRIEN, JAMES, Sistemas de Informação e as Decisões Gerenciais na Era da Internet. São Paulo: Saraiva, 2011.

PÁDUA PAULA FILHO, Wilson de. Engenharia de Software: fundamentos, métodos e padrões. 3 ed. Rio de Janeiro: LTC, 2011.

PRESSMAN, Roger S. Engenharia de Software — Uma Abordagem Profissional. 7. Ed. São Paulo: AMGH, 2011.

PRESSMAN, Roger S. Engenharia de Software — Uma Abordagem Profissional. 8. Ed. São Paulo: AMGH, 2016.

SOMMERVILLE, Ian. Engenharia de Software; tradução Ivan Bosnic e Kalinka G. de O. Gonçalves; revisão técnica Kechi Hiram. 9. ed. São Paulo: Pearson Prentice Hall, 2011.