INTERFACE TECNOLÓGICA

DESENVOLVIMENTO DE SISTEMA WEB PARA CONTROLE DE FLUXOS DE INFORMAÇÃO COM PLANO DE AÇÃO 5W2H

DEVELOPMENT OF A WEB SYSTEM TO CONTROL INFORMATION FLOWS WITH A 5W2H ACTION PLAN

Gustavo Rodrigues da Silva Ribeiro – gustavoribeiro73h@gmail.com
Faculdade de Tecnologia de Catanduva (Fatec) – Catanduva – São Paulo – Brasil

Karolina Fernanda Pereira – karolina.fernanda.pereira@gmail.com
Faculdade de Tecnologia de Catanduva (Fatec) – Catanduva – São Paulo - Brasil

Liriane Soares de Araújo – lirianearaujo@hotmail.com
Faculdade de Tecnologia de Catanduva (Fatec) – Catanduva – São Paulo – Brasil

Ronaldo Rodrigues Martins – ronaldo.martins@fatec.sp.gov.br
Faculdade de Tecnologia de Catanduva (Fatec) – Catanduva – São Paulo - Brasil

DOI: 10.31510/infa.v21i1.1897
Data de submissão: 10/04/2024
Data do aceite: 10/03/2024
Data da publicação: 20/06/2024

RESUMO

A programação para internet é utilizada por diversas empresas com o propósito de fornecer novas soluções para seus clientes, seja por meio de e-commerce ou prestação de serviços. Ademais, outra forma de empregar o ambiente web é por meio dos sistemas de gerenciamento on-line, que auxiliam o cliente interno da organização. Nesse contexto, os sistemas Kanbans, já presentes no mercado, apresentam-se como uma forma ágil de organizar atividades pelo uso de quadros e tarefas. Entretanto, em certos modelos de negócios, é visível uma lacuna que ocorre pela falta de detalhamento, e, a partir disso, foi decidido implementar uma ferramenta 5W2H, que corresponde a sete perguntas que devem ser respondidas para cada uma das tarefas com o objetivo de apresentar um sistema web capaz de apresentar um modelo de Kanban juntamente com o 5W2H que auxilie uma empresa no gerenciamento de tarefas. A metodologia adotada baseia-se em desenvolver diagramas de Engenharia de Software baseados no modelo da organização, reproduzi-las de acordo com a linguagem de marcação HTML; CSS; linguagens de programação JavaScript e C#; com banco de dados SQL e conhecimento sobre as arquiteturas e protocolos da internet de forma a gerar o sistema já mencionado. Os resultados mostram a aplicação, enfocando a aplicação de várias, visando novas soluções eficazes e eficientes para empresas.

Palavras-chave: Programação para Internet. Kanban. Ferramenta 5W2H. Engenharia de Software. Gestão.

ABSTRACT

Internet programming is used by many companies to provide new solutions for their customers, whether through e-commerce or the provision of services. Another way of using the web environment is through online management systems, which help the organization's internal customers. In this context, Kanban systems, which are already on the market, are an agile way of organizing activities using boards and tasks. However, in certain business models, there is a visible gap due to the lack of detail, which is why it was decided to implement a 5W2H tool, which corresponds to seven questions that must be answered for each of the tasks with the aim of presenting a web system capable of presenting a Kanban model together with the 5W2H to help a company manage tasks. The methodology adopted is based on developing Software Engineering diagrams based on the organization's model, reproducing them according to the HTML markup language; CSS; JavaScript and C# programming languages; with SQL database and knowledge of Internet architectures and protocols in order to generate the aforementioned system. The results show the application, focusing on the application of the techniques, aiming at new effective and efficient solutions for companies.

Keywords: Internet programming. Kanban. 5W2H tool. Software Engineering. Management.

1 INTRODUÇÃO

O desenvolvimento de produtos web emergiu para complementar o mercado, trazendo novas soluções para as necessidades dos clientes, sendo comum que as empresas busquem pela inovação por meio da programação para internet. As organizações podem e devem utilizar as técnicas de programação web para se sustentar em seu ramo, seja por meio de produtos, sites e-commerce ou até mesmo suporte interno para gerenciamento da empresa. Ao mencionar o termo “gerenciamento” tem-se a ideia de organizar todos os processos da empresa, havendo várias formas de organizar esses processos, entre eles está o uso do sistema Kanban (Torres, 2014).

A ferramenta Kanban busca separar, por cartões, cada uma das atividades da empresa, sendo possível visualizar o trabalho em andamento, fracionar um tempo adequado, tornar explícitas as tarefas, medir o fluxo para tomadas de decisões, entre outras (Boeg, 2012). Além disso, outra forma de monitoramento é chamada de Plano de ação 5W2H, que traz uma lista de perguntas eficientes que devem ser respondidas ao realizar cada atividade como: “O que?”, “Quando?” e “Por quê?”, o que envolve a participação de gerentes de operações e colaboradores (Sebrae, 2022). Entretanto, apesar das ferramentas de Kanban e 5W2H serem compatíveis para os mesmos objetivos, não é comum que existam aplicações que abrangem ambas, portanto, este trabalho propõe a explorar essa ideia em um cenário por via da colaboração com uma empresa real com o objetivo de empregar as técnicas envolvidas em programação para internet; engenharia de software; e banco de dados; elaborando diagramas do sistema e desenvolvendo

a aplicação web já mencionada. Ademais, este estudo justifica-se pela necessidade de inovações em relação à gestão, buscando compreender os conhecimentos envolvidos para uma situação prática.

2 FUNDAMENTAÇÃO TEÓRICA

Esta seção apresenta os conceitos principais da pesquisa.

2.1 Programação para Internet

Uma página da internet utiliza uma linguagem de marcação que age como uma estrutura externa aos textos contidos nela. Ou seja, é possível organizar o *layout* da página de forma personalizada, por meio de códigos que atuam como instruções incorporadas no próprio conteúdo da página. Com isso, esses códigos podem mudar a fonte de um parágrafo, tornar negrito, itálico, criar bordas, definir margens, adicionar um plano de fundo, posicionar elementos, entre muitas outras funções. Uma das mais conhecidas linguagens de marcação é o *HyperText Markup Language* (HTML) (Mann, 2008). Como complemento ao HMTL também é utilizado uma linguagem de estilização para melhor controle dos estilos visuais, como o *Cascading Style Sheets* (CSS). Do mesmo modo, para ser possível a comunicação, desenvolve-se um código em uma linguagem de programação interpretada, tal como o Javascript (Cunha, 2022).

Por fim, a última definição apresentada refere-se ao Model-View-Controller (MVC) que nada mais é do que um tipo de padrão de arquitetura de software definida em três camadas. A primeira, chamada de “Model”, é responsável pelo acesso e manipulação de dados; a segunda “View”, mostra as informações da interface ao usuário; e a terceira “Controller” que é encarregada de fazer a ligação das duas camadas anteriores, permitindo que o conteúdo da “Model” possa ser repassado para a “View” e vice-versa (Guedes, 2020). Vale ressaltar que, em um contexto de sistemas web e softwares, a segurança é avaliada como uma das qualidades essenciais ao oferecer esses serviços. A proteção é ameaçada quando um agente externo encontra uma vulnerabilidade não tratada. A palavra vulnerabilidade pode representar diversas situações – tal como erros de código, configurações e operações – que comprometem o funcionamento da aplicação (Lung, 2006).

2.2 Engenharia de Software

De acordo com Aggarwal (2005), a Engenharia de Software foi criada como uma forma de solucionar adversidades nos projetos e proporcionar, de forma organizada, sistemas de excelência, com fácil manutenção e sem ultrapassar os limites de tempo e dinheiro. Uma das técnicas mais conhecidas é a Programação Orientada a Objeto, que funciona como uma estratégia do Design de Software – que por sua vez compõe a Engenharia de Software (Gomaa, 2011), em que diversas classes definem vários objetos da vida real, por meio de métodos e atributos.

Outra abordagem para o desenvolvimento de um software é o chamado *Domain-Driven Design* (DDD), usado para lidar com regras de negócio, implementação de classes, permissões de banco de dados, entre outros. A finalidade dele é dividir os processos que podem ser naturalmente complexos. Com isso, cada parte dessa arquitetura fica com uma responsabilidade, sendo possível facilitar no desenvolvimento desse sistema (Alves, 2018).

Em sequência, a Linguagem de Modelagem Unificada (UML), tal como o nome sugere, é uma linguagem visual usada na modelagem de um software. Ela existe com o objetivo de documentar os componentes de um sistema, mostrando a relação entre eles e seus comportamentos dinâmicos. A UML abrange diversos diagramas para o desenvolvimento de um software, dentre eles o Diagrama de Caso de uso, Diagrama de Classe e Diagrama de Sequência (Gomaa, 2011).

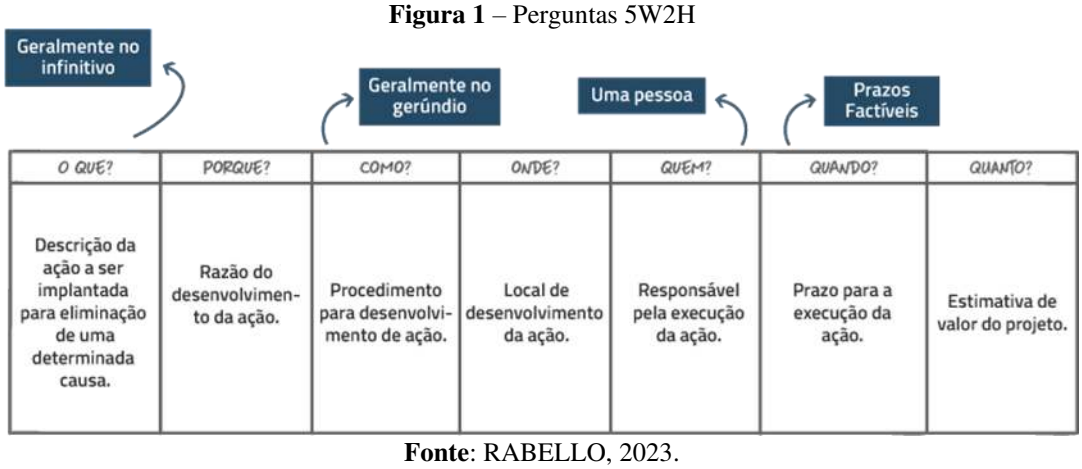
Primeiramente, de acordo com Pressman (2011), o Diagrama de Caso de Uso demonstra a interação de um ator com um sistema, visando atingir um objetivo. Além disso, conforme Fonseca (2022, p.5), “os diagramas de caso de uso descrevem o relacionamento entre os usuários, o sistema e seus casos de uso.”. Em segundo lugar, o Diagrama de Classe traz tabelas nomeadas como classes, sendo cada uma delas objetos do mundo real que se relacionam entre si. Essas entidades físicas são identificadas e podem ser classificadas e definidas por meio de atributos e métodos (Pressman, 2011). Além disso, cada classe é representada por caixas, fazendo com que qualquer relacionamento entre elas seja demonstrando conectando-as por linhas (Gomaa, 2011, p.16). Em terceiro lugar, O Diagrama de Sequência possui várias definições. Entre elas, apresentada por Pressman (2011, p.56), que “podem ser usados para mostrar como os eventos provocam transições de objeto para objeto. Uma vez que os eventos tenham sido identificados pelo exame de um caso de uso, o modelador cria um diagrama de sequência - uma representação de como os eventos provocam o fluxo de um objeto para outro

em função do tempo". Por fim, também unido ao ciclo de vida de um software, tem-se o banco de dados. Esse conceito nada mais é do que um grupo de dados que se relacionam entre si, ou seja, um banco armazena vários dados de pessoas e processos de uma única empresa, por exemplo. Esses sistemas de banco de dados são geralmente modelados utilizando o Diagrama de Entidade e Relacionamento (DER) (Bagui, 2012).

2.3 Ferramenta 5W2H

O método 5W2H é definido pelo Sebrae (2022) como um checklist de perguntas para prevenir ou corrigir falhas em atividades de uma empresa. Também chamado de Plano de Ação, é implementado com o objetivo de planejar cada passo de uma tarefa – desde sua metodologia, definição de cronograma, responsáveis, entre outros. Esse plano é colocado em prática de forma padronizada, por meio de perguntas simples, possibilitando uma boa organização e agilidade. A sigla 5W2H é composta pelas iniciais de sete palavras em inglês: *What* (O quê), *Why* (Por quê), *Where* (Onde), *When* (Quando), *Who* (Quem), *How* (Como) e *How much* (Quanto custa) (Rabello, 2023). Em outras palavras: "[...] a função dessa ferramenta é definir o que será feito, por que, onde, por quem, quando, como e quanto isso custará." (Sebrae, 2022). A Figura 1 mostra a descrição desses conceitos e algumas regras.

Figura 1 – Perguntas 5W2H



Fonte: RABELLO, 2023.

Visando aumentar a clareza de objetivos, responsabilidades e prazos – é vantajoso para a empresa que cada tarefa definida seja acompanhada dos conceitos para que os funcionários e gerentes tenham conhecimentos sobre aquela atividade. Com isso, há a redução de falhas internas, aumento da produtividade dos colaboradores e colabora para uma comunicação eficaz (Rabello, 2023). Por exemplo, o gerente de uma empresa define que é necessário realizar uma

determinada tarefa. Ao repassar essa demanda para um funcionário, ele pode utilizar das sete perguntas, de forma que o colaborador entenda o que exatamente será feito; por que está fazendo; como proceder; onde será realizado; em que momento; por quem e por qual custo (Sebrae, 2022).

3 PROCEDIMENTOS METODOLÓGICOS

De início, foram marcados encontros com o sócio administrador da empresa parceira para discutir e identificar processos falhos, incompletos ou que necessitassem de melhoria. Assim, a organização mencionou que dependia de sistemas de terceiros para a comunicação interna entre gestores e funcionários. Os gestores estão responsáveis por organizar as tarefas diárias, semanais e mensais que os funcionários devem realizar. Na teoria, os gestores apenas criam tarefas com as determinadas descrições da atividade a ser feita e incluem elas em cartões divididos em “concluído”, “em andamento” e “a fazer”. Os demais colaboradores acessam as descrições e iniciam seu trabalho, podendo arrastar as tarefas para os campos determinados. Entretanto, na prática, há algumas falhas que atrasam o processo de conclusão das atividades, principalmente quando se pensa na comunicação entre esses dois profissionais. Ao preencher as devidas descrições para as atividades, os gestores incluíam informações muitas vezes incompletas, visto que, para eles, que já sabem o que exatamente deve ser feito torna-se fácil entender seus resumos. Porém, alguns funcionários não conseguiam compreender ao todo o processo da tarefa, sendo necessário ligar, enviar um email ou mensagem ao gestor perguntando sobre. Isso causava diversos atrasos tanto nas tarefas inconcluídas, quanto na demanda de outras atividades feitas pelo gestor.

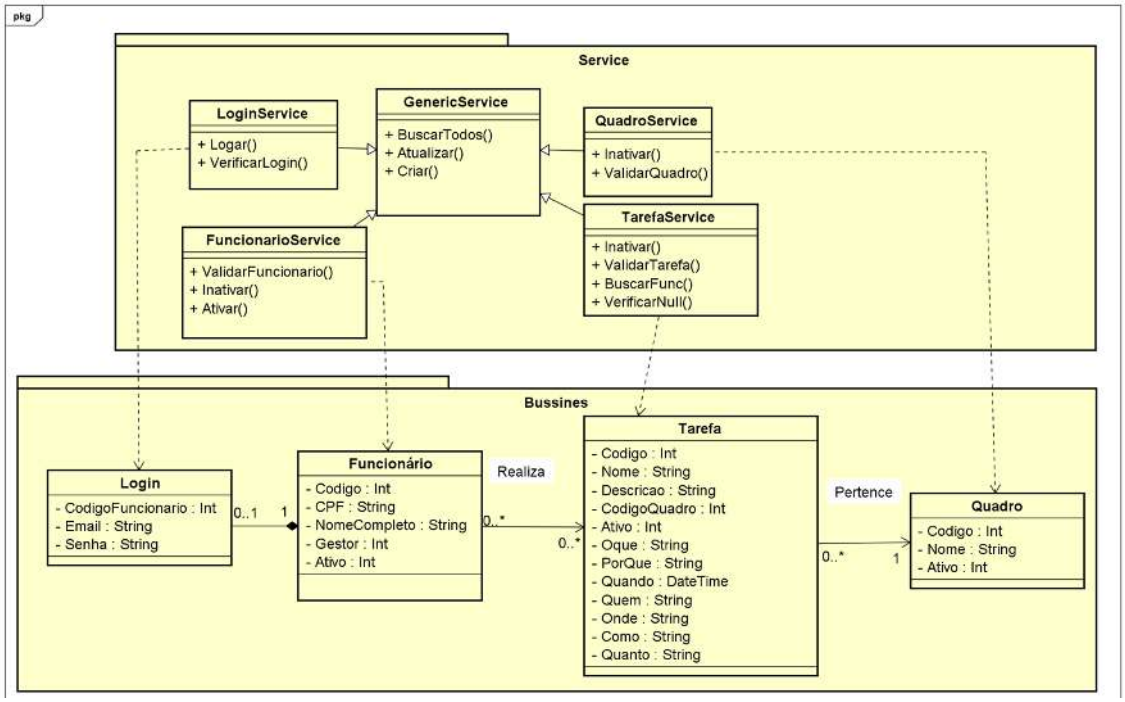
Assim sendo, foram estudadas formas de padronizar as tarefas, de forma a orientar o gestor da melhor forma ao incluir tarefas. Além disso, planejou-se um método de funcionários acessarem as informações das tarefas de forma mais rápida para agilizar o processo de cumprimento dessas funções. A ferramenta 5W2H já mencionada foi sugerida pelo próprio sócio, pedindo também para manter uma interface de Kanban já semelhante ao sistema já utilizado pela empresa, ou seja, com cartões e tarefas.

4 RESULTADOS E DISCUSSÃO

Por meio das várias bibliografias lidas, conceitos estudados e práticas desenvolvidas, foi possível implementar um sistema web capaz de contribuir para a comunicação interna de uma empresa. Primeiramente, houve o desenvolvimento de 4 diagramas baseados nos conceitos aprendidos de Engenharia de Software baseados na ideia analisada pela empresa. No contexto do **Diagrama de Caso de Uso**, tem-se o ator "Gestor" – responsável por gerir as atividades e funcionários – representando as interações do ator com o sistema. De início, o Gestor pode, portanto, realizar 4 ações: Logar no sistema, Cadastrar Funcionário, Criar Quadro e Criar Tarefa.

O **Diagrama de Classe** agrupa, de forma simplificada, todos os atributos e métodos que cada classe possui – informação muito importante para a programação do projeto. Os atributos se localizam abaixo ao nome da classe juntamente com seus tipos. Por último é representado os métodos, sempre com "();". Nesse caso, todos os atributos são privados, enquanto os métodos são públicos. A Figura 2 mostra o Diagrama de Classe.

Figura 2 – Diagrama de Classe



Fonte: Autoria Própria.

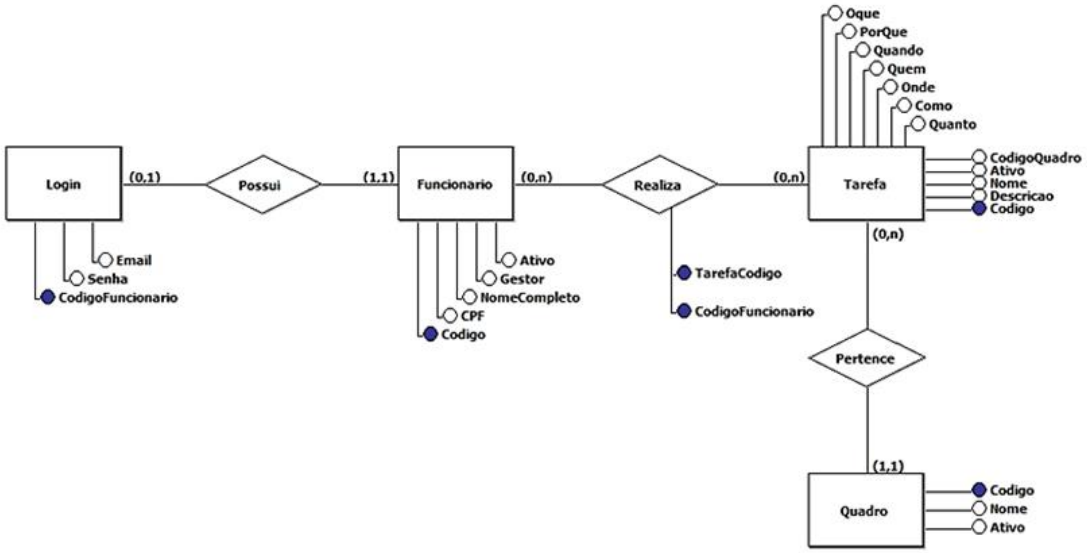
O **Diagrama de Sequência** tem como ator o gestor, sendo o responsável por gerenciar toda a área de trabalho da empresa. Ele inicia a navegação acessando a tela principal do sistema, por meio de um método GET, e em seguida a classe "HomeController" aciona três métodos principais: "BuscarTodos" no "QuadroService" que retorna uma lista dos quadros base do

sistema; o “BuscarFunc” em “TarefaService”, que retorna uma lista das tarefas cadastradas; e “BuscarTodos” em “FuncionarioService”, que retorna uma lista de todos os funcionários cadastrados. Todos esses retornos são recebidos e são adicionadas à tela principal. Em seguida, a ação para criar um quadro será feita através de um POST para “HomeController” acessando o método “ControleQuadro”. Depois é criado um objeto Quadro com o nome “quad” e atribuído os campos a ele. A seguir, o método “Criar” é chamado para o “QuadroService”, e o Gestor é redirecionado para a tela principal do sistema. O mesmo se segue com “Tarefa”, porém acessando o método “ControleTarefa”, que cria um objeto “novaTarefa”, e atribui os devidos campos. Em sequência para o “TarefaService” tem-se uma requisição para validar o objeto, chamando o método “ValidarTarefa|” e retornando se está de acordo com o banco de dados. Caso a resposta seja positiva, é chamado o método “Criar” passando “novaTarefa” como parâmetro e, em decorrência, o Gestor é redirecionado novamente para a tela principal. Vale ressaltar que, tal diagrama não é apresentado em figura aqui neste artigo por questão de tamanho.

Por fim, em relação ao **Diagrama de Entidade e Relacionamento**, tem-se as entidades: Login, Funcionario, Tarefa e Quadro. Nesse contexto, há relacionamentos entre essas tabelas, em que um funcionário pode possuir no máximo 1 login, enquanto um login deve ser atribuído a somente 1 funcionário. Um funcionário também pode estar realizando várias tarefas e essa, por sua vez, pode estar sendo realizada por vários funcionários. Esse relacionamento em específico tem uma peculiaridade: é necessário que o banco crie uma tabela secundária com os dados de funcionário e tarefa, ou seja, qual funcionário está fazendo qual tarefa. Ou seja, a relação “Realiza” tem os atributos: CodigoFuncionario e TarefaCodigo (ambas chaves primárias – chamada de chave primária composta). Por fim, uma tarefa deve pertencer à 1 quadro somente, enquanto um quadro pode possuir várias tarefas. Na Figura 3, tem-se o Diagrama de Entidade e Relacionamento.

INTERFACE TECNOLÓGICA

Figura 3 – Diagrama de Entidade e Relacionamento



Fonte: Autoria Própria.

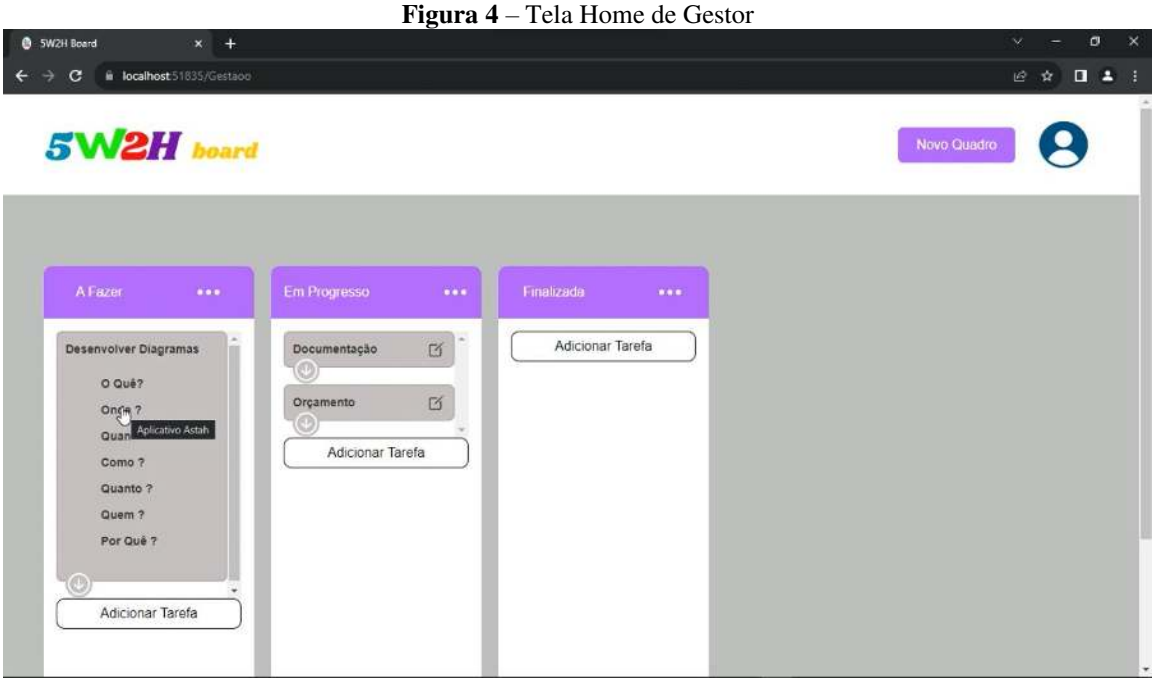
4.1 Camada de Negócios, de Persistência, de Repositório e Serviço e Telas do Negócio

A Camada de Negócios fica responsável por criar as classes com os seus atributos baseados nos diagramas. Após criar todas as classes na Camada *Business*, é necessário a instalação dos pacotes do Entity Framework para manipular o banco de dados. Nessa camada, será criado um arquivo de configuração denominado *context* onde será feito o mapeamento das classes para a criação das tabelas no banco de dados. O próximo passo foi criar um arquivo de repositório para cada classe, onde serão escritos os métodos definidos no diagrama, tal como inativar, buscar ativos, entre outras. Nessa camada também é adicionado uma classe para inicialização do banco de dados, onde haverá informações base para a criação do banco de dados na primeira execução. O próximo passo foi a criação de uma aplicação de console para que os métodos criados fossem testados antes da implementação do *frontend*. Esse aplicativo foi descartado posteriormente após os testes concluídos. Por fim, foi feita a Camada de Serviço, onde foram criados os Serviços Genéricos, que serão incluídos os métodos da camada de serviço, usados para todas as classes, assim como já foi mencionado anteriormente. Além disso foi implementado o controlador referente à tela de cadastro de funcionário, onde a ação index é responsável por retornar a tela que o gestor tem acesso. Na sequência, tem-se as *Views* do sistema, ou seja, as telas montadas com a programação de Razor HTML, CSS e JavaScript. Portanto, cada ação de um usuário que resulte na necessidade de uma troca de tela é criada *View*. Como por exemplo, no cadastro de funcionário, ao criar, editar ou consultar, tem-se a

necessidade de uma nova tela, logo, uma nova *View*. Além disso, cada uma é chamada especificamente por uma ação de um *Controller*. Assim, para cada nova tela, é criada uma ação *View Controller*.

Um passo muito importante é configurar a conexão automática com o banco de dados. Dessa forma, toda vez que a aplicação for inicializada, ela imediatamente é ligada ao banco. Ainda nesse arquivo, será configurada a rota padrão que será seguida pelo usuário, nesse caso, definiu-se que a tela de Login deve ser a primeira a ser mostrada. Ademais, é preciso interligar a camada de serviço e o controlador para, de fato, interagir com o sistema. Através dos recursos do Razor HTML, que nos permite usar diretivas do C# juntamente com *tags* específicas, para ser possível manipular o banco de dados, tal como criar, editar ou inativar os dados do sistema. A Figura 4 mostra a tela principal do sistema, em que são observados todos os quadros e tarefas presentes no banco de dados, além dos botões de cadastro de quadro e tarefa. Nota-se que essas funcionalidades são acessadas apenas pelo gestor.

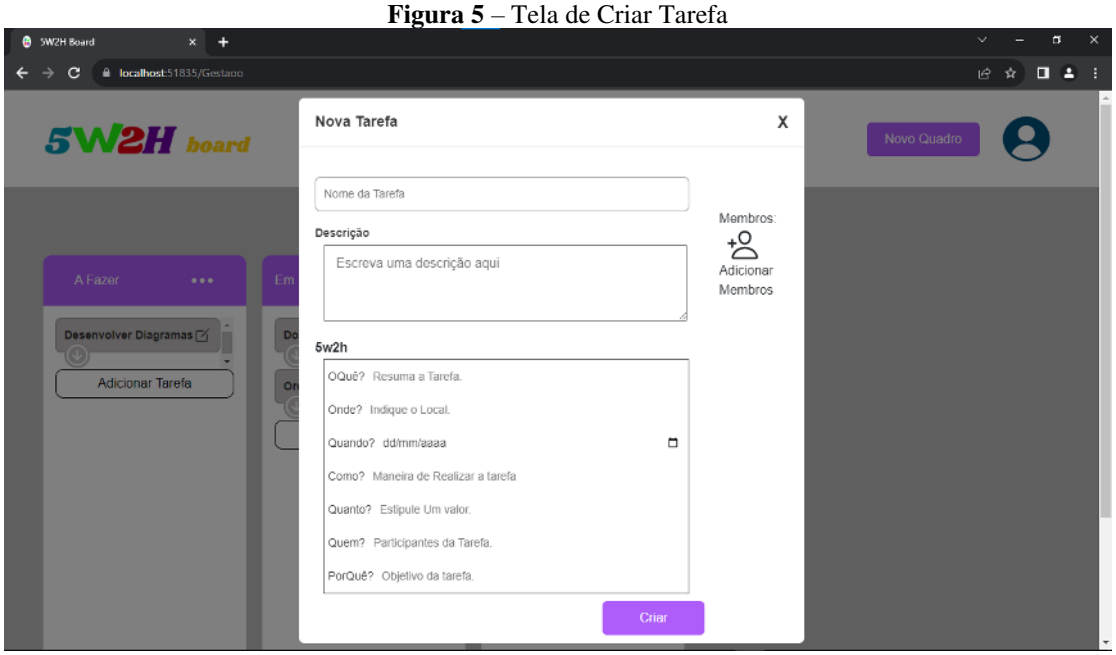
Figura 4 – Tela Home de Gestor



Fonte: Autoria Própria.

Já como funcionário, a única diferença da tela Home é que não apresenta os botões para criação de quadros e tarefas, pois apenas o gestor é capaz de criar tarefas e quadros. Já a criação das Tarefas é feita pela tela inicial, em que é mostrado um modal com os campos necessários para cadastrar a tarefa. É importante ressaltar que a edição, inativação e visualização são feitas de forma similar. Portanto, na Figura 5, tem-se essa tela.

Figura 5 – Tela de Criar Tarefa



Fonte: Autoria Própria.

O cadastro de funcionário é feito pelo Gestor da empresa, em uma tela própria.

5 CONCLUSÃO

Por meio deste estudo, foi possível compreender na prática os conceitos de programação para internet e Engenharia de Software, tais como a utilização das linguagens HTML, CSS e Javascript; métodos HTTP, arquitetura MVC, além das técnicas para elaboração de diagramas de Caso de Uso, de Classe, de Sequência, Entidade e Relacionamento e o próprio sistema desenvolvido. O trabalho viabilizou uma nova solução para os contratempos referentes à falta de informação nas tarefas atribuídas aos funcionários, por meio de um Kanban com quadros e tarefas. Sendo assim, a melhoria citada descreve um sistema que contém: quadros, responsáveis por armazenar tarefas; tarefas, que determinam uma atividade a ser feita por um ou mais funcionários; e seu diferencial sendo campos baseados na ferramenta 5W2H.

Da mesma maneira, é importante ressaltar que a manutenção do sistema deve ser constante, buscando resolver possíveis bugs e implementando novas funções que sejam necessárias de acordo com a regra de negócio da empresa. Por fim, conclui-se que a análise e as projeções aqui citadas colaboram para o desenvolvimento de um sistema web capaz de resolver problemas de comunicação entre gestor e funcionário, sendo essa solução o principal

objetivo do trabalho, além de compreender os conceitos de programação para internet já mencionados.

REFERÊNCIAS

AGGARWAL, Krishan Kumar; SINGH, Yogesh. **Software Engineering**. 2. ed. Nova Delhi: New Age International Publisher, 2005. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=dx2C9Zkez5YC&oi=fnd&pg=PA1&dq=Software+Engineering&ots=Ze_cVkLncF&sig=OlzsGda5t0wPZBQb1fxX8vItWd0#v=onepage&q=Software%20Engineering&f=false. Acesso em: 17 set. 2023.

ALVES, Alex. **Começando com .NET Core, com Arquitetura em Camadas**. Medium, 2018. Disponível em: https://alexalvess.medium.com/criando-uma-api-em-net-core-baseado-na-arquitetura-ddd-2c6a409c686#:~:text=O%20DDD%20(Domain%20Driven%20Design,para%20o%20dom%C3%ADnio%20do%20neg%C3%B3cio. Acesso em: 16 set. 2023.

BAGUI, Sikha; EARP, Richard. **Database Design Using Entity-Relationship Diagrams**. 2. ed. Nova Iobrque: CRC Press, 2012. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=ebMfR0rgq0cC&oi=fnd&pg=PP1&dq=entity+and+relationship+diagram&ots=rFDT5nv-ih&sig=k0KgmRCoWGV9KS-HUcxkYci4BMo#v=onepage&q=entity%20and%20relationship%20diagram&f=false. Acesso em: 17 set. 2023.

BOEG, Jesper. **Kanban em 10 passos:** Otimizando o fluxo de trabalho em sistemas de entrega de software. São Fransico: InfoQ Brasil, 2012. Disponível em: https://docplayer.com.br/1687924-Kanban-em-10-passos-sobre-o-autor-jesper-boeg.html. Acesso em: 24 ago. 2023.

CUNHA, Fernando. **Sistema Web:** o que é e como funciona?. Mestres da Web, 2022. Disponível em: https://www.mestresdaweb.com.br/tecnologias/sistema-web-o-que-e-e-como-funciona. Acesso em: 24 set. 2023.

FONSECA, Letícia. **10 exemplos de diagrama de caso de uso (e como criá-los)**. Venngage, 2022. Disponível em: https://pt.venngage.com/blog/diagrama-de-caso-de-uso-2/. Acesso em: 06 mai. 2023.

GUEDES, Marylene. **O que é MVC?**. Treinaweb, 2020. Disponível em: https://www.treinaweb.com.br/blog/o-que-e-mvc. Acesso em: 17 set. 2023.

GOMAA, Hassan. **Software Modeling & Design**: UML, Use Cases, Patterns, and Software Architectures. Nova Iorque: Cambridge University Press, 2011. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=TqZi-hAX17YC&oi=fnd&pg=PR7&dq=Software+modeling&ots=4WZfye8uQI&sig=puaLly8RvkMQ2BsepgJ_R8kAEZ0#v=onepage&q=Software%20modeling&f=false. Acesso em: 17 set. 2023.

LUNG, Lau Cheuk. **Minicursos do VI Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSeg 2006)**. Santos: Sociedade Brasileira de Computação, 2006. Disponível em: https://sol.sbc.org.br/livros/index.php/sbc/catalog/download/101/458/730-1?inline=1. Acesso em: 20 set. 2023.

MANN, Margi; ROBISCHON, Rose. HTML Standards-History and Future. **The Serials Librarian**, [S.l], v. 36, n. 1-2, p. 51-57, 2008. DOI: 10.1300/J123v36n01_10. Disponível em: https://www.tandfonline.com/doi/abs/10.1300/J123v36n01_10. Acesso em: 19 set. 2023.

PRESSMAN, S. Roger e Maxim, Bruce R. **Engenharia de Software**. 9 ed. Porto Alegre: Grupo A, 2021. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=FSE3EAAAQBAJ&oi=fnd&pg=PT33&dq=livro+engenharia+de+software&ots=kzOMiPtEzE&sig=MexSg-BJKEa7zaVw9O3mEjZoR-o#v=onepage&q=livro%20engenharia%20de%20software&f=false. Acesso em: 17 set. 2023.

RABELLO, Guilherme. **O que é 5W2H e como ela pode aumentar produtividade?**. Siteware, 2023. Disponível em: https://www.siteware.com.br/metodologias/o-que-e-5w2h/. Acesso em: 06 mai. 2023.

SEBRAE. **5W2H:** o que é, para que serve e por que usar na sua empresa. Serviço Brasileiro de Apoio às Micro e Pequenas Empresas (SEBRAE), 2022. Disponível em: https://www.sebrae-sc.com.br/blog/5w2h-o-que-e-para-que-serve-e-por-que-usar-na-sua-empresa. Acesso em: 21 set. 2023.

TORRES, Joaquim. **Guia da Startup: Como startups e empresas estabelecidas podem criar produtos web rentáveis**. São Paulo: Casa do Código, 2014. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=g2SCCwAAQBAJ&oi=fnd&pg=PT11&dq=sites+web+para+solu%C3%A7%C3%B5es+empresas&ots=ApAIiT0gZY&sig=yqq46OGnnwjDU0gHX8Cs80M7zNI#v=onepage&q=sites%20web%20para%20solu%C3%A7%C3%B5es%20empresas&f=false. Acesso em: 24 ago. 2023.