

UNIVERSIDADE FEDERAL DE SANTA CATARINA

Sistema para gerenciamento de tarefas com gamificação

GUILHERME DE LIZ DO NASCIMENTO

Florianópolis - SC

2022

UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Sistema para gerenciamento de tarefas com gamificação

GUILHERME DE LIZ DO NASCIMENTO

Trabalho de Conclusão do Curso de Graduação em
Sistemas de Informação do Centro de Sistemas de
Informação da Universidade Federal de Santa
Catarina como parte dos requisitos para a obtenção
do título de Bacharel em Sistemas da Informação.
Orientador: Prof. Dr. Elder Rizzon Santos

Florianópolis - SC

2022

Guilherme de Liz do Nascimento

Sistema para gerenciamento de tarefas com gamificação

Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Sistemas da Informação.

Orientador: Prof. Dr. Elder Rizzon Santos

Banca examinadora:

Membro: _____

Avaliador Frank Augusto Siqueira

Membro: _____

Avaliador Jose Eduardo de Lucca

Florianópolis - SC

2022

RESUMO

Com o avanço da complexidade da sociedade, principalmente graças à evolução das tecnologias, a demanda por produtividade das pessoas também aumenta, porém o tempo disponível para o cumprimento dessas tarefas ainda é o mesmo. Com o intuito de auxiliar na necessidade do cumprimento de diversas atividades em determinado prazo, este trabalho busca apresentar um modelo de gerenciamento de tarefas que as priorize conforme suas características (como urgência e importância), baseado na matriz de Eisenhower, e encaixe em uma agenda com espaços de tempo disponíveis. Para testar a eficiência do modelo, será desenvolvida uma aplicação web com técnicas de gamificação, buscando aumentar o engajamento e produtividade dentro da plataforma. Desta forma, este trabalho será útil para direcionar da melhor maneira os esforços do usuário no cumprimento de suas responsabilidades.

Palavras-chave: Gerenciamento de tarefas, Matriz de Eisenhower, Gamificação, Aplicação web.

LISTA DE FIGURAS

Figura 1: Matriz de Eisenhower.....	17
Figura 2: Interface do Microsoft To Do.....	18
Figura 3: Interface do Google Tarefas.....	19
Figura 4: Cadastro de lembrete no Google Agenda.....	20
Figura 5: Tarefa no Trello.....	21
Figura 6: Quadro de tarefas no Trello.....	22
Figura 7: Tela do aplicativo Habitica.....	23
Figura 8: Gamificação no radar de trânsito.....	32
Figura 9: Adesivo de mosca do <i>Urinal Fly</i>	32
Figura 10: Círculo mágico, segundo o autor.....	33
Figura 11: Comparação entre Matriz de Eisenhower e Diagrama de Sung.....	39
Figura 12: Diagrama de casos de uso.....	57
Figura 13: Diagrama de atividades.....	60
Figura 14: Diagrama de classes.....	60
Figura 15: Wireframe - Tela de login.....	62
Figura 16: Wireframe - Tela inicial.....	63
Figura 17: Wireframe - Cadastro de tarefas.....	64
Figura 18: Wireframe - Configuração da plataforma.....	65
Figura 19: Wireframe - Comparação entre modelos.....	65
Figura 20: Wireframe - Comparação qualitativa entre modelos.....	66
Figura 21: Código da implementação da priorização sem modelo, porém reordenada.....	68
Figura 22: Código da implementação do modelo básico.....	69
Figura 23: Código da implementação do modelo avançado.....	70
Figura 24: Código da implementação do modelo alternativo.....	71
Figura 25: Tela de visão geral do sistema web.....	72
Figura 26: Tela para cadastro de tarefas.....	73
Figura 27: Tela para configurar a plataforma.....	74
Figura 28: Tela para comparar os modelos.....	74
Figura 29: Tela para avaliar os modelos.....	75
Figura 30: Tela para visualizar as tarefas em forma de agenda de execução.....	75

LISTA DE TABELAS

Tabela 1 - Número de recursos e tarefas por pessoa. Recursos online estão sombreados.....	36
Tabela 2 - Comparação dos trabalhos relacionados.....	41
Tabela 3 - Resultado do primeiro experimento, métrica de tempo livre.....	49
Tabela 4 - Resultado do primeiro experimento, métrica de taxa de entrega no prazo.....	50
Tabela 5 - Resultado do segundo experimento.....	51
Tabela 6 - Resultado do terceiro experimento, medidas da métrica de tempo livre...	52
Tabela 7 - Resultado do terceiro experimento, medidas da métrica de taxa de entrega no prazo.....	53
Tabela 8 - Resultado do terceiro experimento, agregado das métricas.....	53

LISTA DE SIGLAS

CIMM - Centro de Informação Metal Mecânica

PWA - Progressive Web App

MOGA - Multiobjective Genetic Algorithm

AHP - Analytic Hierarchy Process

PMBOK - Project Management Body of Knowledge

RPG - Role-playing game

HTTP - Hypertext Transfer Protocol

HTML - HyperText Markup Language

CSS - Cascading Style Sheet

JRA - Jogos de Realidade Alternativa

NFC - Near-field Communication

DARPA - Defense Advanced Research Projects Agency

SUMÁRIO

RESUMO	3
SUMÁRIO	7
1 INTRODUÇÃO	9
1.1 OBJETIVOS	10
1.1.1 Objetivo Geral	10
1.1.2 Objetivos Específicos	11
1.2 ESCOPO	11
1.2.1 Resultados Esperados	11
1.2.2 Premissas	11
1.2.3 Restrições	11
1.2.4 Critérios de aceitação	12
1.3 MÉTODO DE PESQUISA	12
1.4 ORGANIZAÇÃO	13
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 CONCEITOS	14
2.1.1 Tarefa	14
2.1.2 Gerenciamento de tarefas	14
2.1.3 Gamificação	15
2.1.4 Algoritmo	15
2.1.5 Matriz de Eisenhower	16
2.2 SOLUÇÕES SIMILARES	17
2.2.1 Microsoft To Do e Google Tarefas	17
2.2.2 Google Agenda	19
2.2.3 Trello	20
2.2.4 Habitica	22
2.3 Desenvolvimento da plataforma	24
3 TRABALHOS RELACIONADOS	26
3.1 Social scheduler: a proposal of collaborative personal task management [10]	26
3.2 WORK RIFT - RPG de Realidade Alternativa para gerenciamento de tarefas [11]	29
3.3 What a To-Do: Studies of Task Management Towards the Design of a Personal Task List Manager. [2]	35
3.4 The Sung Diagram: Revitalizing the Eisenhower Matrix. [12]	38
3.5 Considerações	40
4 Modelo de Gerenciamento de Tarefas	42
4.1 Conceito geral	42

4.1.1 Gamificação	44
4.2 Forma de implementação	44
4.3 Métricas de avaliação	45
4.4 Experimentos realizados	46
4.5 Análise dos resultados	48
4.5.1 Tarefas reais	48
4.5.2 Tarefas aleatórias	50
4.5.3 Lote de conjuntos de tarefas aleatórias	51
5 Desenvolvimento do sistema web	55
5.1 Tecnologias utilizadas	55
5.1.1 Servidor	55
5.1.2 Plataforma web	56
5.2 Modelagem	56
5.2.1 Casos de uso	57
5.2.2 Diagrama de atividades	59
5.2.3 Diagrama de classes	61
5.3 Elementos de gamificação	61
5.4 Estudo da interface	62
5.5 Aplicação do modelo de priorização de tarefas	66
5.5.1 Sem modelo	67
5.5.2 Modelo básico	68
5.5.3 Modelo avançado	69
5.5.4 Modelo alternativo	71
5.6 Resultado	72
6 CONSIDERAÇÕES FINAIS	77
6.1 Trabalhos futuros	78
REFERÊNCIAS	80
APÊNDICES	83
APÊNDICE A - Conjunto de tarefas reais - semana 1	83
APÊNDICE B - Conjunto de tarefas reais - semana 2	84
APÊNDICE C - Conjunto de tarefas com dados aleatórios	85
APÊNDICE D - Códigos-fonte	87
APÊNDICE E - ARTIGO	88

1 INTRODUÇÃO

No nosso dia-a-dia executamos diversas tarefas, tanto no âmbito profissional quanto pessoal e familiar, e com o avanço das tecnologias em geral podemos perceber essas tarefas mudando e aumentando em quantidade e complexidade.

Isso nos leva a ter uma necessidade de organizar da melhor forma possível o recurso escasso e precioso que temos: o tempo. Segundo os autores (Ohmukai; Takeda; Miki, 2003), a complexidade em gerir os recursos pessoais aumenta juntamente com o avanço da complexidade da nossa sociedade.

Uma boa administração do tempo é essencial para uma boa produtividade e conforme Rodrigues (2018, apud Chaves, 2008):

[...] gerir o tempo é saber usá-lo para fazer atividades consideradas importantes e prioritárias, tanto na vida pessoal, quanto na profissional. Assim, a afirmação de “não ter tempo suficiente” é decorrente do uso demasiado do tempo em tarefas que são consideradas dispensáveis ou não prioritárias.

Além disso, a pandemia que vem ocorrendo desde o ano de 2020 trouxe ainda mais desafios para a execução das tarefas, segundo Gonçalves (2020), como por exemplo ter que conciliar o trabalho com a vida familiar, adaptando um ambiente domiciliar que começou a dividir atividades de trabalho, escolares, domésticas e de lazer ao aderir a modalidade de *home office*. E essa adaptação vem evoluindo conforme gestores entendem melhor a medida certa da demanda de tarefas dos profissionais, enquanto estes encontram a melhor forma de organizar e executar as suas atividades.

Desta forma, este trabalho se propõe a desenvolver um modelo de gerenciamento de tarefas baseado na Matriz de Eisenhower, considerando que o “seu objetivo consiste em evitar que tarefas se acumulem e se sobreponham umas às outras, minando o fluxo dos processos” (SILVA, 2020). Para isso, a matriz originalmente utiliza uma relação entre a urgência e importância de cada tarefa para priorizá-las, mas podemos adicionar mais parâmetros para aprimorar a priorização destas tarefas, como a carga de trabalho (em horas).

No entanto, para conseguir visualizar este modelo em ação será desenvolvido um sistema web para inserir as tarefas com seus parâmetros e visualizar os resultados. Para isso, será utilizado o modelo de *Progressive Web App (PWA)*, que

segundo Tandel e Jamadar (2018) é uma forma de construir sistemas web que consigam entregar além da versão para computadores, uma boa experiência móvel parecida com o nativo dos dispositivos.

Como a execução de tarefas pode ser algo um pouco menos atrativo para as pessoas, a aplicação utilizará algumas técnicas de gamificação. Navarro (2013) explica o objetivo da gamificação, que se enquadra bem nesta situação, onde o objetivo é aumentar o engajamento dos usuários ao utilizar a aplicação:

Percebe-se que o objetivo principal da gamificação é criar envolvimento entre o indivíduo e determinada situação, aumentando o interesse, o engajamento e a eficiência na realização de uma tarefa específica, buscando mudar o comportamento desse indivíduo.

O conceito de gamificação serve para qualquer tipo de tarefa, mas no escopo deste trabalho estarão focados na produtividade dos usuários. Conforme Costa e Marchiori (2015), a gamificação conseguiu adentrar no meio empresarial justamente por apresentar uma abordagem de envolvimento e produtividade dos envolvidos. Isto pode ser feito com elementos como pontuações, sistemas de níveis e até premiações, que nos remetem as mesmas técnicas utilizadas em jogos desde o primórdio destes.

Com isso, um método para gerenciamento de tarefas utilizando a matriz de Eisenhower pode ser utilizado por diferentes indivíduos que procuram priorizar uma sequência de tarefas, enquanto a aplicação com características de gamificação pode ser utilizada com diferentes objetivos, mas principalmente por estudantes priorizando atividades universitárias ou por funcionários priorizando tarefas da atividade profissional.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O objetivo consiste em propor um método, representado por uma expressão matemática, para gerenciamento de tarefas baseado na Matriz de Eisenhower adaptado à gamificação e testá-lo mediante sua implementação em um sistema web.

1.1.2 Objetivos Específicos

1. Analisar o estado da arte em gerenciamento de tarefas, técnicas de gamificação e desenvolvimento de *Progressive Web Apps* (PWA);
2. Desenvolver um modelo de gerenciamento de tarefas adaptado à gamificação, utilizando a matriz de Eisenhower e algumas características específicas de gamificação.
3. Realizar experimentos com o modelo proposto para medir o seu desempenho.
4. Definir a melhor tecnologia para o desenvolvimento do sistema web.
5. Testar o modelo desenvolvido por meio da implementação de um sistema web.

1.2 ESCOPO

1.2.1 Resultados Esperados

- Modelo para gerenciamento de tarefas.
- Plataforma web para aplicação do modelo.
- Testes na priorização de um conjunto de tarefas.
- Documentação.

1.2.2 Premissas

Para a realização de alguns testes, serão utilizadas algumas agendas com conjuntos diferentes de tarefas, com valores diferentes entre si.

1.2.3 Restrições

- O projeto deve estar concluído até dezembro de 2022.
- A documentação deve seguir as normas da ABNT e o modelo de TCC da UFSC.

- A aplicação deve estar funcionando com a implementação do modelo de priorização.

1.2.4 Critérios de aceitação

A plataforma web deve conseguir priorizar uma lista de tarefas simples e dispor em uma agenda, baseada nos dados informados. E apresentar a ordem de um conjunto de tarefas priorizadas e os elementos da gamificação sendo influenciados pelas entradas informadas.

1.3 MÉTODO DE PESQUISA

Para este trabalho, foi necessário uma pesquisa bibliográfica exploratória em relação ao gerenciamento de tarefas, abordagens de gamificação e desenvolvimento web. Com as informações obtidas, espera-se que seja possível selecionar a ferramenta para o desenvolvimento da aplicação, que supra as necessidades do projeto, seja bem documentada e tenha uma comunidade ativa utilizando.

Além disso, identificar os parâmetros e os pesos para cada um deles, com o objetivo de projetar o modelo de priorização das tarefas. Tal modelo deve conseguir organizar um conjunto de atividades em uma ordem que seja eficiente e produtiva para o usuário realizar.

Com essa ordenação realizada, pode-se encaixar as tarefas em uma agenda informada pelo usuário, de forma a ficar claro quando é necessário executar cada atividade para conseguir concluir os objetivos a tempo.

Será possível incluir na aplicação algumas técnicas de gamificação para tentar aumentar o interesse dos usuários em registrar e executar suas tarefas. Assim, estaria tanto aumentando a produtividade dos mesmos, quanto melhorando o engajamento na plataforma.

Por fim, deve-se realizar testes preliminares para comprovar o funcionamento da priorização de tarefas por meio do modelo proposto. Ou seja, verificar se em condições normais, o modelo priorizaria as tarefas adequadamente e a aplicação conseguiria apresentar essa priorização.

1.4 ORGANIZAÇÃO

No [capítulo 2](#) iniciaremos apresentando alguns conceitos básicos para a compreensão da proposta, além de apresentar algumas soluções similares que conseguimos encontrar.

No [capítulo 3](#) discorreremos sobre alguns trabalhos relacionados com a proposta principal, desde conteúdos sobre modelos de gerenciamento de tarefas até uma dissertação que apresenta um sistema para gerenciamento com gamificação.

No [capítulo 4](#) descrevemos toda a lógica por trás do modelo de priorização de tarefas e realizamos alguns experimentos para analisar o desempenho do deste em comparação com outras formas de priorização.

No [capítulo 5](#) é abordado o desenvolvimento da plataforma web que facilita o acesso à implementação do modelo e permite um experimento qualitativo sobre o desempenho deste.

No [capítulo 6](#) são apresentadas as considerações finais e são observadas algumas possibilidades para futuros trabalhos sobre o tema.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 CONCEITOS

Para garantir que todos os conceitos utilizados no decorrer deste documento fiquem claros em seus sentidos, considerando que em alguns pode haver uma ambiguidade de significado de acordo com o contexto, se faz necessário esclarecer alguns temas.

2.1.1 Tarefa

O termo tarefa possui diferentes significados em diferentes escopos, como quando tratamos de processadores na área de tecnologia de informação ou uma tarefa de casa para um aluno.

No entanto, para o escopo deste trabalho usaremos uma definição mais adequada onde a tarefa seria um “trabalho, ação ou conjunto de ações a serem realizadas para o cumprimento de determinada missão, objetivo ou meta, num determinado prazo” (dicionário CIMM¹). Porém, para simplificar, consideremos que a tarefa é uma ação que possui um objetivo e entrega algum valor, formada ou não por um conjunto de atividades. Além disso, ela possuirá alguns atributos que a caracterizam (que detalharemos mais a frente), como um título, uma descrição própria, um prazo de entrega, uma carga de trabalho demandada para sua execução, entre outros.

2.1.2 Gerenciamento de tarefas

Quando falamos sobre o gerenciamento de tarefas estamos nos referindo ao ato de organizar, planejar, priorizar e definir as tarefas, em uma determinada ordem, e suas características.

Ao pensar de forma prática o que isso significa, pode-se dizer que vai desde gerenciar uma simples lista de afazeres (como ao fazer uma anotação em um papel) até utilizar sistemas mais complexos de gerenciamento, que apresentam mais

¹ Dicionário do Centro de Informação Metal Mecânica, disponível em <<https://www.cimm.com.br/portal/verbetes/exibir/1182-tarefa>>.

funcionalidades e integrações (partindo para uma solução digital). Vamos analisar alguns exemplos no [capítulo 2.2](#) mais a frente.

2.1.3 Gamificação

Apesar dos jogos existirem desde os tempos antigos, o termo gamificação é algo relativamente recente, ainda mais na área de desenvolvimento. Segundo Navarro (2013), a palavra ainda não possui um conceito definitivo e exato, porém pode ser explicada como “a aplicação de elementos, mecanismos, dinâmicas e técnicas de jogos no contexto fora do jogo”. Onde inclui características como a definição de desafios e metas, conquistas como pontos e troféus, entre outros elementos que encontramos constantemente nos jogos.

A autora define algumas características dos jogos que podem ser implementadas em outros ambientes, algumas destas são úteis para este projeto, como por exemplo:

Participação voluntária: a participação no “jogo” proposto deve ser opcional para o usuário. No caso do sistema, deve ser possível utilizar as funcionalidades (ao menos as básicas) sem se envolver com a gamificação.

Meta: o objetivo definido deve ser claro para os envolvidos.

Regras: para alcançar as metas, devem ser seguidas as regras definidas, envolvendo ainda mais o usuário no jogo.

Sistema de resultados: como há uma meta a ser atingida, espera-se um resultado ao cumpri-la ou não. Além disso, *feedbacks* relacionados ao desempenho na evolução e alcance dos objetivos definidos.

Além disso, encontramos um padrão nos sistemas que implementam gamificação, como por exemplo: um personagem com uma quantidade de vida e de experiência, passando de nível conforme adquire mais experiência ao concluir objetivos menores e recebendo conquistas em determinados momentos chave (como um nível específico atingido ou uma quantidade de vezes que concluiu uma tarefa).

2.1.4 Algoritmo

Segundo a Oxford Languages², um algoritmo é um “conjunto das regras e procedimentos lógicos perfeitamente definidos que levam à solução de um problema em um número finito de etapas”. Ou seja, podemos dizer que é uma sequência de instruções e comandos que são realizados de forma sistemática para resolver um problema ou executar uma tarefa.

Note que o significado do termo não está relacionado com tecnologia, apesar de utilizarmos muito para tal. Podemos não perceber, mas os algoritmos estão presentes em praticamente tudo que fazemos diariamente, mas conseguimos representar por meio de algoritmos várias atividades que executamos diariamente.

Todo os passos feitos desde acordar até sair de casa, os passos para seguir uma receita, as instruções de instalação de algum produto ou programa, entre outros casos comuns no dia-a-dia são exemplos de instruções que executamos sequencialmente, algumas vezes sem pensar por já estarmos tão acostumados e não percebendo isso, mas que poderiam ser representados como um algoritmo.

Em relação às tarefas, o esforço que fazemos para organizar e priorizar nossos afazeres diários ou semanais é um algoritmo que executamos de maneira automática, porém será que fazemos isso da melhor maneira possível? Acredito que um computador com um algoritmo bem desenvolvido e treinado poderia nos tornar mais produtivos e efetivos nas tarefas, automatizando esse processo de priorização por exemplo.

2.1.5 Matriz de Eisenhower

A Matriz de Eisenhower procura auxiliar na execução de tarefas, por definição, o “seu objetivo consiste em evitar que tarefas se acumulem e se sobreponham umas às outras, minando o fluxo dos processos” (SILVA, 2020).

A matriz proposta por Eisenhower seria algo como um plano cartesiano, com um eixo sendo a importância da tarefa e o outro sendo a urgência dela. No final, seria possível definir o destino da tarefa conforme o que encontramos na figura abaixo.

² Disponível em <<https://languages.oup.com/>>.

Figura 1: Matriz de Eisenhower.

	URGENTE	NÃO URGENTE
IMPORTANTE	I QUADRANTE "Faça agora"	I QUADRANTE "Agende"
NÃO IMPORTANTE	I QUADRANTE "Delegue"	I QUADRANTE "Elimine"

Fonte: Artigo de Terra, 2022 [18].

Dessa forma, tarefas sem urgência e sem importância deveriam ser eliminadas, enquanto tarefas com muita importância e muita urgência deveriam ser feitas agora, por exemplo.

2.2 SOLUÇÕES SIMILARES

Como já foi abordado, o gerenciamento de tarefas é algo que está presente na rotina das pessoas de forma geral; conseqüentemente, diversas plataformas foram desenvolvidas para auxiliar nesse quesito.

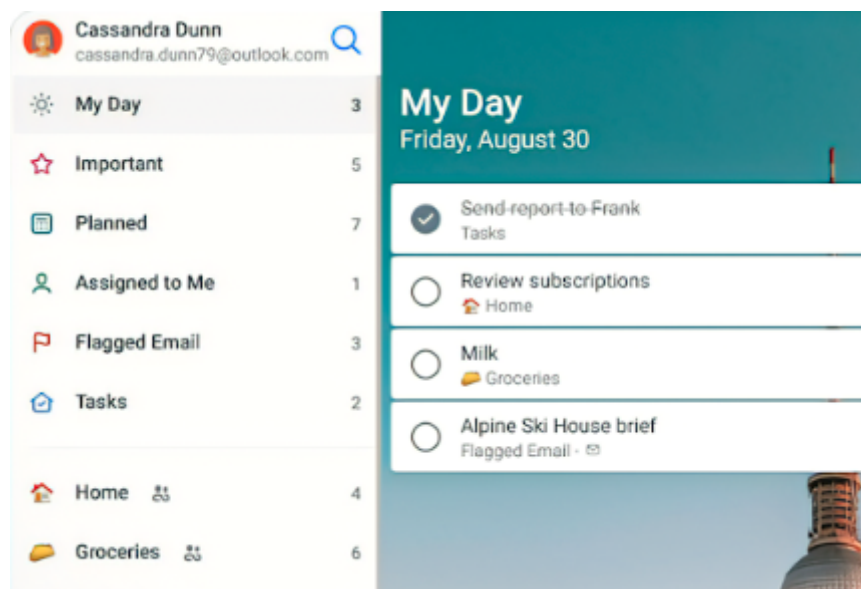
Desde sistemas mais simples, com apenas o gerenciamento de uma lista de tarefas com título e prazo de entrega, até sistemas mais completos, que permitem um detalhamento maior da tarefa e oferecem mais recursos para tal. Vamos abordar os principais para entender o que temos disponível e mais utilizado atualmente.

2.2.1 Microsoft To Do e Google Tarefas

Como um sistema simples de gerenciamento de listas de tarefas é algo relativamente simples de ser desenvolvido (muito comum em exemplo durante aulas de desenvolvimento, inclusive), há uma grande quantidade de exemplos que poderiam ser citados para esta categoria.

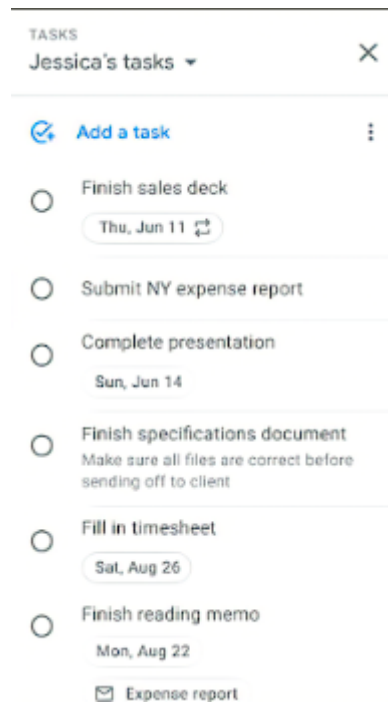
Porém, vamos destacar as soluções de duas grandes empresas de tecnologia (Microsoft e Google), que possuem soluções semelhantes, o Microsoft To Do e o Google Tarefas. Neste caso, de forma geral as principais funcionalidades são muito semelhantes: a criação de tarefas com um título simples e uma data de conclusão e a possibilidade de agrupar estas tarefas em listas diferentes.

Figura 2: Interface do Microsoft To Do.



Fonte: Página do Microsoft To Do na loja de apps da Google.³

³ Disponível em <<https://play.google.com/store/apps/details?id=com.microsoft.todos>>. Acesso em: 26 nov 2022.

Figura 3: Interface do Google Tarefas.

Fonte: Página do Google Tarefas na loja de apps da Google.⁴

2.2.2 Google Agenda

O Google Agenda originalmente é um calendário, para o gerenciamento de eventos na agenda. No entanto, ele permite adicionar outros itens além de eventos, como tarefas e lembretes. E como os sistemas da empresa são integrados, ao adicionar uma tarefa no Google Tarefas com uma data de conclusão, ela aparece no Google Agenda.

⁴ Disponível em <<https://play.google.com/store/apps/details?id=com.google.android.apps.tasks>>. Acesso em 27 nov 2022.

Figura 4: Cadastro de lembrete no Google Agenda.

Adicionar título

Evento Tarefa **Lembrete**

🕒 9 mar. 2022 10:30am

🔄 Não se repete Dia inteiro

Salvar

Fonte: Plataforma do Google Agenda.⁵

Além destas funcionalidades, a ferramenta permite ter outros usuários envolvidos nos eventos, levando a parte colaborativa para a gestão de tarefas. Algo que não se estende para a parte de tarefas e lembretes, de forma que o gerenciamento de tarefas no sentido deste trabalho (listar tarefas, com seus prazos, para então executá-las) não é possível de fazer de forma colaborativa.

Por ser tão simples e genérico, consegue ser uma ferramenta suficiente para diferentes situações e contextos, como por exemplo, para gerenciar as tarefas diárias de uma forma geral e genérica.

2.2.3 Trello

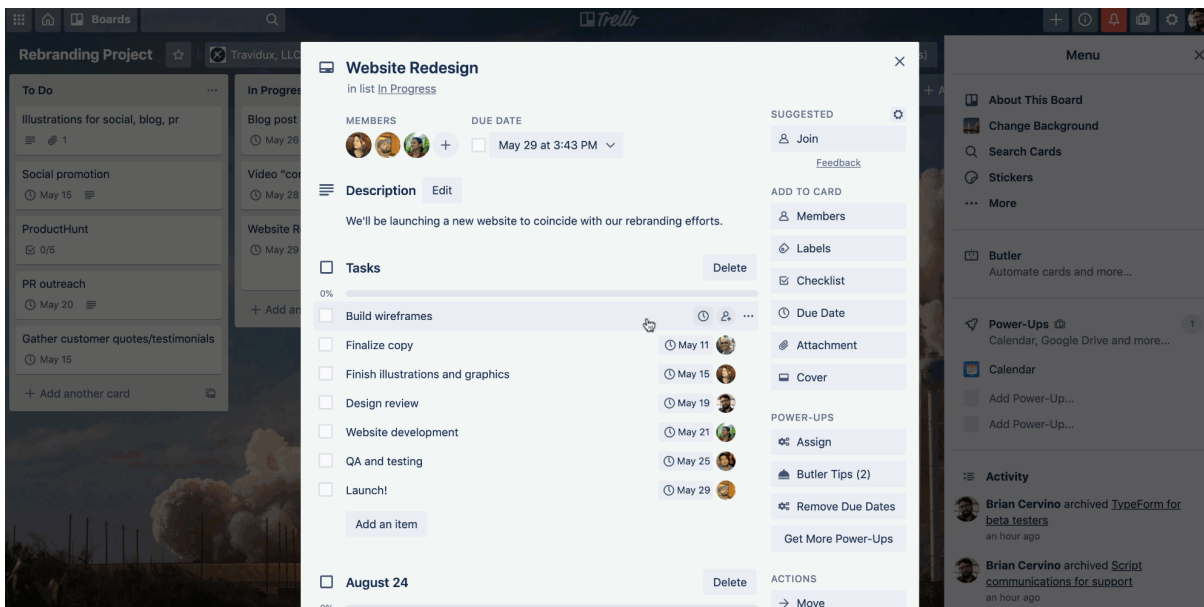
Provavelmente a ferramenta mais utilizada entre desenvolvedores, traz a possibilidade de gerenciar tarefas, com diversas customizações (como anexos, descrições, comentários em cada tarefa, data de conclusão, entre outros), além de integrar com outros sistemas e oferecer um bom suporte para a parte colaborativa.

A figura 4 apresenta como seria uma tarefa na ferramenta note, onde já temos algo bem mais complexo que os exemplos anteriores, que somente armazenavam o título e data de conclusão das tarefas. Nesse caso, já temos um

⁵ Disponível em <calendar.google.com>. Acesso em 20 jan 2022.

espaço para detalhar, checklists, comentários, etiquetas, entre diversas outras opções.

Figura 5: Tarefa no Trello.



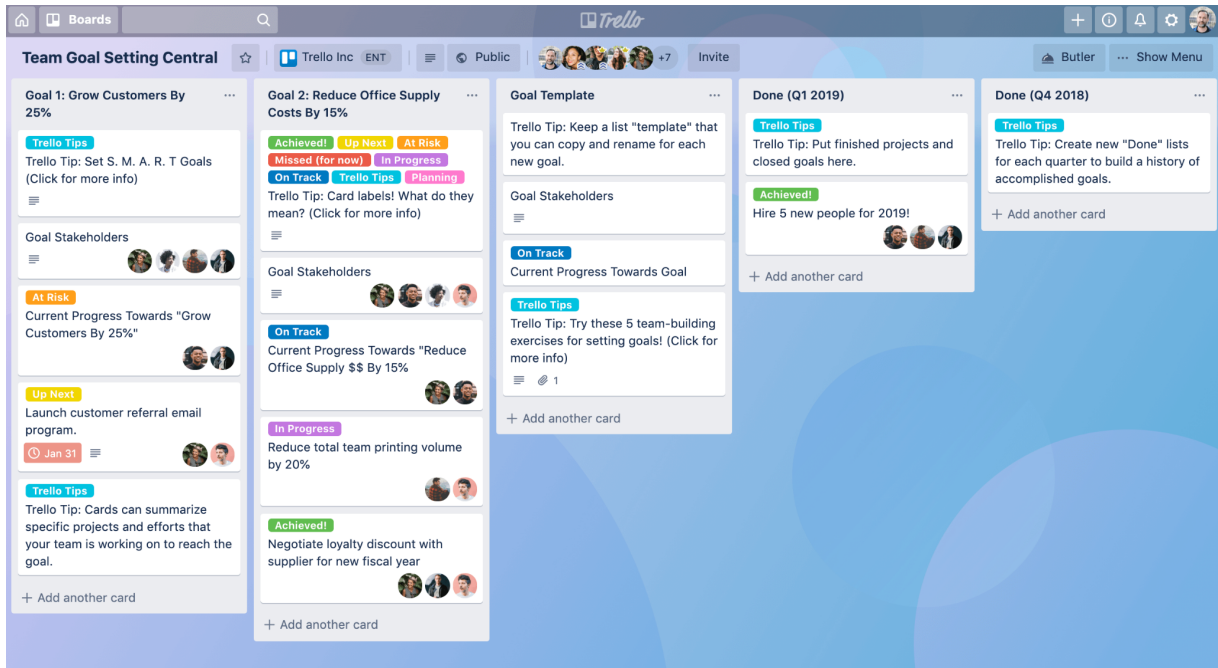
Fonte: Blog do Trello.⁶

Porém, toda a parte de prioridades precisa ser feita manualmente, no máximo com o auxílio de etiquetas que o usuário pode ou não utilizar para indicar o nível de prioridade.

Para isso, a ferramenta distribui as tarefas em colunas (definidas pelo usuário) onde a ordem de apresentação é conforme foram dispostas pelo indivíduo, ou seja, um trabalho de priorização totalmente manual. Somente é possível alterar a ordem de apresentação de uma lista (dentro de uma coluna) por nome ou data de entrega das tarefas, o que não necessariamente reflete na prioridade adequada para a execução destas. Abaixo como seria um quadro na ferramenta.

⁶ Disponível em <<https://blog.trello.com/br/advanced-checklists>>. Acesso em 26 nov 2022.

Figura 6: Quadro de tarefas no Trello.

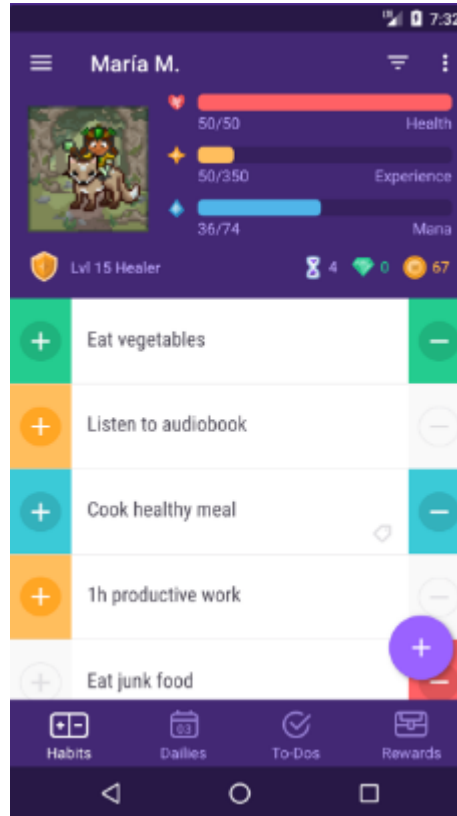


Fonte: Blog do Trello.⁷

2.2.4 Habitica

Dentre as soluções encontradas, esta é a que mais se aproxima da solução proposta. Com o foco em ajudar o usuário a se manter motivado e organizado (como o autor descreve) a ferramenta une o gerenciamento de tarefas (principalmente as diárias) com elementos de gamificação bem desenvolvidos.

⁷ Disponível em <<https://blog.trello.com/br/como-ser-lider-equipe-trabalho>>. Acesso em 27 nov 2022.

Figura 7: Tela do aplicativo Habitica.

Fonte: Site do aplicativo.

Mais complexo que os outros gerenciadores citados, este inclui elementos de gamificação (como metas, recompensas, punições, entre outros) ao gerenciamento de tarefas. Além de tarefas pontuais (feitas apenas uma vez), também trabalha com atividades rotineiras, para o desenvolvimento de hábitos.

Entretanto, a ferramenta não permite realizar as tarefas de forma colaborativa (pelo menos até a versão utilizada nos testes). Dessa forma, toda a parte de execução deve ser realizada pelo próprio usuário ou gerenciando a terceirização sem a ajuda da plataforma.

Além disso, as tarefas não apresentaram algum esquema de priorização para que fosse possível identificar uma ordem pelo qual as tarefas deveriam ser realizadas. Perdendo um certo potencial pelo fato de já ter os principais dados (como as tarefas e seus prazos).

2.3 Desenvolvimento da plataforma

Para o desenvolvimento do sistema que irá implementar o modelo proposto, com o objetivo de simplificar a parte de programação da solução, será feito um *Progressive Web App (PWA)*, que conforme Tandel (2018) afirma, é uma forma de construir e configurar aplicações web que além de entregar uma boa experiência na versão para computadores, entrega uma funcionalidade parecida com a nativa para dispositivos móveis. Ou seja, com apenas um código é possível entregar uma boa solução para mais plataformas.

Esse tipo de sistema vem avançando nos últimos anos e está bem integrado aos principais navegadores, onde consegue interagir com funcionalidades de dispositivos móveis (como notificação, câmera, entre outros) que até então só poderiam ser acessados pelos aplicativos nativos.

E com a popularização deste modelo, vários *frameworks* para desenvolvimento das interfaces já o integraram aos seus ambientes. Desta forma, para conseguir implementá-lo, basta configurar alguns parâmetros simples de como deverá funcionar o sistema.

Por causa da experiência que os desenvolvedores já possuem em desenvolvimento web, o sistema utilizará o framework VueJS, específico para criação de interfaces. O ecossistema das ferramentas que o utilizam já possui diversas bibliotecas que auxiliam na implementação que serão utilizados, como: Vue Router (para gerenciamento das rotas que serão utilizadas no navegador), Vuex (gerenciamento de estado, ou seja, controlar os dados utilizados globalmente no sistema), Vuetify (biblioteca de design, com componentes, utilitários, entre outros) e Vue PWA (configuração automática para rodar PWA).

Para montar o servidor, utilizamos tecnologias baseadas na linguagem de programação Python, que serão melhor detalhadas no capítulo 5.

Toda a lógica do sistema de priorização das tarefas, bem como o gerenciamento delas e das conexões para a colaboração entre os usuários estará implementada no lado do servidor, por motivos de segurança (não ter a lógica exposta no navegador) e para melhoria de desempenho, já que a máquina onde estará hospedado ficará responsável pelos principais processamentos e o navegador do usuário fica apenas responsável pela apresentação das informações.

Numa primeira versão do sistema, os usuários apenas colaboram entre si de forma assíncrona (semelhante ao funcionamento dos serviços de e-mail) e para conseguir utilizar algum sistema de terceiro para melhorar o gerenciamento das tarefas seria necessário fazer isso manualmente por fora da plataforma.

3 TRABALHOS RELACIONADOS

3.1 Social scheduler: a proposal of collaborative personal task management [10]

O artigo busca apresentar uma proposta colaborativa entre um **modelo de aliança** e um **modelo *human-in-the-loop*** (centrado no humano) para o gerenciamento de tarefas pessoais. O modelo de aliança (*alliance*) consiste em compartilhar informações entre pessoas e fomentar a colaboração nas atividades. Já o modelo *human-in-the-loop* apresenta três subsistemas para apoiar as atividades de tomada de decisão, sendo eles: um **visualizador**, um **otimizador** e um **recomendador**. Onde o visualizador busca apresentar os atributos em cada tarefa, como data de entrega, prioridade subjetiva e carga de trabalho, todas definidas pelo usuário. O otimizador gera uma agenda de tarefas com base nas atividades informadas por meio de um cronograma e um algoritmo genético de multi-objetivos. E o recomendador avalia as alternativas de acordo com a hierarquia analítica.

A principal necessidade que o **modelo de gerenciamento colaborativo** procura suprir seria a gestão de recursos (como tempo, ativos e conhecimentos) das pessoas em diferentes ambientes. Porém, os sistemas colaborativos possuem dois problemas principais atualmente: são muito focados em apenas uma organização e o trabalho entre múltiplos sistemas se torna complicado por causa da inconsistência dos dados e problemas de segurança.

Por outro lado, os sistemas de gerência pessoal mais populares (como calendários ou planejadores diários) apresentam soluções muito simples, sem garantir um controle eficiente do tempo, por exemplo.

Um modelo colaborativo pessoal ficaria no meio das duas soluções citadas, possuindo três principais funções:

Modelo de gerenciamento democrático: Apresenta não só um modelo mais eficiente, mas também mais democrático por colocar todas as pessoas como semelhantes (sem hierarquia entre si).

Múltiplos grupos com limites variados: A ideia seria cobrir desde organizações comuns até grupos mais emergentes.

Relações pessoais complicadas e dinâmicas: Para melhorar a qualidade, é necessário considerar as condições para pessoas de diferentes grupos e não considerar um histórico passado como determinante, já que as relações mudam constantemente.

O principal problema do gerenciamento de tarefas pessoais que os autores citam seria que os modelos desenvolvidos para problemas de agenda são basicamente uma ordenação de um conjunto de tarefas que possuem atributos próprios (como prazo, prioridade e carga de trabalho). Porém, para cada indivíduo, as prioridades e valores vistos em cada parte podem ser diferentes, complicando o processo da construção da agenda.

Para o **método de colaboração**, os autores utilizaram uma rede pessoal que contém as relações autorizadas e as relações colaborativas de cada usuário. Onde a relação autorizada é uma solicitação para comunicação e a relação colaborativa é uma requisição para atuar junto em uma tarefa. A rede pessoal é agrupada em subconjuntos pelo servidor, chamados de alianças. Com um método de filtro adequado, é possível fazer com que essas alianças formem grupos emergentes, assumindo que estes possuem unidades de informações compartilhadas.

Cada usuário pode criar tarefas colaborativas que podem ser compartilhadas com outros usuários que fazem parte da sua rede de relações. Para isso, o servidor verifica se o usuário tem alguma conexão com o outro e libera acesso aos dados da tarefa que foram disponibilizados para acesso. Para gerar essa rede de conexões, o servidor pode criar uma matriz utilizando a base de tarefas, permitindo gerenciar quem faz parte da rede de conexões ou não.

Para os usuários que fazem parte dessa rede, é possível visualizar tarefas e a condição de tempo disponível conforme permissões. Com isso, o usuário tem a possibilidade de ter noção prévia da disponibilidade de tempo de outras pessoas para colaborarem em alguma tarefa.

Já o **modelo *human-in-the-loop*** é citado pelos autores como uma extensão do modelo da teoria de tomada de decisão, que propõe três atividades para uma decisão: atividade de inteligência, atividade de design e atividade de escolha.

Baseado nisso, eles definiram os seguintes elementos do modelo:

Visualizador (Suporte para atividade de inteligência): Segundo Ohmukai e Takeda (2003), pode apresentar o cronograma de tarefas em diferentes escalas (como semanal e mensal). Além de estimular que o usuário conheça o estado da tarefa, bem como as principais características dela, como a carga de trabalho e possíveis conflitos de horário entre as tarefas.

Otimizador (Suporte para atividade de design): Conforme os autores explicam, busca gerar diferentes opções de agendas utilizando algoritmos de otimização, como algoritmo genético, especialmente com multi-objetivos (MOGA). As funções deste algoritmo seriam: maximizar o número de tarefas, maximizar a utilização do tempo, maximizar a recompensa baseada em prioridade subjetiva e maximizar a recompensa baseada na prioridade do usuário.

Recomendador (Suporte para atividade de inteligência): Os autores propõem um método baseado em processo de análise hierárquica (AHP) para lidar com dados qualitativos e quantitativos. Porém o número de comparações aumenta conforme o aumento das alternativas e como é necessário refazer essas comparações ao adicionar uma nova, é necessário a utilização de outros métodos, como o método de medição absoluta para evitar os problemas.

Segundo os autores, com o sistema proposto por eles, conseguiram obter as seguintes vantagens:

Procedimento democrático para tomada de decisão: Ao invés do agendamento ser feito por uma pessoa que controla a agenda dos subordinados, cada indivíduo pode reagendar conforme a disponibilidade dos outros.

Identificar e apoiar grupos emergentes: Ao invés de um administrador gerenciar as permissões e acessos, a própria rede de conexões forma os grupos.

Método de filtragem inteligente: As informações das tarefas e dos usuários só é acessada por aqueles com permissão para tal.

Diminuição do custo de entrada de dados: Um usuário consegue atualizar a tarefa de diversos membros, não sendo necessário a redundância de tarefas e atualizações.

Resolver o conflito de tarefas: O modelo *human-in-the-loop* disponibiliza um método simples para resolver os conflitos de tarefas, por meio do otimizador.

Oferece múltiplas perspectivas da agenda e tarefas: Variando entre visão de calendário, cronograma ou tarefas colaborativas.

Os testes deste sistema foram realizados com 12 pessoas por praticamente 3 semanas, que tinham relações por diferentes ambientes (como laboratórios e faculdades). Destes, 10 pessoas relataram que adicionaram mais tarefas no sistema por causa das tarefas colaborativas.

No segundo experimento, a média de tarefas criadas por pessoa foi 17.1, com uma média inicial de 7.3 conflitos por pessoa. Cada usuário avaliou quantitativamente (entre 1 e 7) comparando a condição inicial e após o resultado do algoritmo, obtendo uma avaliação média de 4.5.

Após a melhoria do método, a média de conflitos reduzir para 3.1 por pessoa, além da avaliação média ter ficado em 5.5 após a utilização do otimizador e recomendador.

3.2 WORK RIFT - RPG de Realidade Alternativa para gerenciamento de tarefas [11]

O autor cita a apresentação feita por Jane McGonigal no TED 2010⁸, onde ela discorre sobre a imersão causada pelos jogos, de como as pessoas têm um sentimento de satisfação ao concluir missões dentro dos jogos e como esse sentimento cresce conforme a complexidade das conquistas.

Baseada nisso, ela afirma que seria possível utilizar o foco e o raciocínio estratégico dos jogadores para lidar com desafios da vida real, como a fome, pobreza e conflitos globais. Segundo a apresentadora, são despendidas 3 milhões de horas semanais em jogos, porém de acordo com seus cálculos seriam necessárias 21 bilhões de horas semanais dos jogadores focadas em resolução de problemas reais. Com isso, ela estimula que cada vez mais pessoas se tornem jogadores.

Tanto que McGonigal desenvolveu diversos jogos durante a sua carreira que levassem o usuário a resolver problemas relacionados com o mundo real. A esse modelo de interação foi dado o nome de jogos de realidade alternativa.

Já no contexto de gerenciamento de projetos, o autor cita como o livro sobre o PMBOK destaca o custo e o tempo do projeto como características impactantes e que devem ser bem gerenciadas, para garantir a qualidade e o sucesso do projeto.

⁸ Disponível em <https://www.ted.com/talks/jane_mcgonigal_gaming_can_make_a_better_world>.

Isso pensando em execução de tarefas, onde o autor faz o paralelo entre tarefa e Ordem de Serviço, termo muito utilizado em empresas que possuem prestação de serviços internamente, como um setor de tecnologia prestando suporte aos demais funcionários. E de acordo com a citação do autor do Guia PMBOK, a tarefa ainda pode ser subdividida em unidades menores, denominadas atividades. Já o autor citado Clot (199, apud BARROSO, 2016, p. 27) utiliza os termos gênero e estilo para tarefas e atividades, respectivamente. Além disso, Clot explica o que chama de Inibição, que seria o fato de uma tarefa ser impedida de ser concluída.

Dentro das ferramentas e modelos de gerenciamento de tarefas que serviram de inspiração para o autor (Barroso, 2016), ele destaca algumas como:

Trello é uma ferramenta de gerenciamento de tarefas que, segundo o autor, foi descrita como algo que torna a colaboração em projetos simples e divertida. Utilizando um fluxo simples, ela utiliza elementos visuais (como quadros e cartões) para apresentar o fluxo de tarefas dentro do projeto. Esse sistema tem como base a estratégia denominada **Kanban**, de forma onde cada tarefa possui um “estado” dentro do processo de entrega do projeto. Além disso, a ferramenta disponibiliza diversas outras características para a descrição das tarefas, além do histórico de atividades que pode ser consultado facilmente.

Já **Scrum** é uma metodologia ágil para gerenciamento de projetos, principalmente de desenvolvimento de software. Seu foco é o próprio cliente, onde a entrega de valor é feita por meio de ciclos temporais chamados de Sprints. A ferramenta **Runrun.it** é pensada para a utilização do Scrum, onde as tarefas possuem atributos que podem ser utilizados para controlar o cronograma e estabelecer metas para o próximo Sprint.

Além disso, podem ser citados três papéis importantes no Scrum: Scrum Master (responsável pelo gerenciamento), proprietário do produto (responsável por levantar as demandas do cliente) e a equipe (quem realiza as entregas do produto).

Entrando para o contexto de gamificação, Barroso explica que em cada contexto de uma empresa pode-se ter uma cultura diferente de tarefas para o fluxo de trabalho utilizado, mas isso não significa que os funcionários ficam motivados com o trabalho. Ainda mais considerando que estes métodos adicionam ainda mais burocracia e trabalho, já que é necessário a organização das tarefas.

O autor divide o foco e a concentração em uma tarefa entre fatores extrínseco e intrínseco. Fatores extrínsecos seriam relacionados aos incentivos externos, como

uma recompensa com valor de troca ao alcançar um objetivo. Já os fatores intrínsecos seriam relacionados ao próprio indivíduo, como o prazer por ter concluído uma tarefa, um possível sentimento de felicidade gerado e até mesmo o estado de concentração e ansiedade ao executar a tarefa.

O autor cita que a sustentação do fluxo se deve pelo deleite gerado, de forma que um estudo isolou alguns elementos que contribuem, como: chance de completar uma tarefa, concentração no que é executado, objetivos claros, respostas imediatas às ações, senso de controle das ações, entre outros. Ou seja, o fluxo não é tão focado no sucesso ou fracasso, mas mais na trajetória e sua organização de forma a deixar o usuário imerso na atividade, um efeito chamado de imersão.

Desta forma, o autor cita alguns estudos que indicam que a imersão é geralmente realizada por meio de jogos, onde há realmente um ambiente onde a pessoa é “transportada”. Fora isso, ao participar de um site social, há sensações que se assemelham a estes jogos.

Outra característica interessante encontrada nos jogos digitais seria a autonomia que o usuário tem para realizar modificações. Nos sites sociais, isso pode ser encontrado na própria construção de identidades pelo usuário.

Tendo em vista isso, Barroso explica que a ideia da gamificação seria trazer um modo de pensamento através dos jogos e mecânicas lúdicas aplicadas em diferentes contextos, com o objetivo de aumentar o engajamento e a orientação comportamental do usuário.

Um estudo citado pelo autor define quatro estágios encontrados nos jogos: objetivo pré-lúdio, meios lúdicos, regras constitutivas e atitude lúdica. Ou seja, a presença de regras é a principal forma de levar o comportamento do usuário ao processo desejado. Além disso, pode-se dizer que a criatividade é uma das habilidades mais trabalhadas pela gamificação, já que o usuário é estimulado a encontrar uma solução baseada nas regras e premissas propostas.

Um exemplo prático disso seria relacionado a um caso onde algumas técnicas de gamificação foram aplicadas no trânsito, chamado de *Lottery Speed Camera*, onde o motorista recebia um ticket de loteria caso respeitasse as leis (pago pelas multas daqueles que não respeitam), além de uma alteração no radar de velocidade, estimulando ainda mais o respeito pelo limite, como pode-se ver a seguir:

Figura 8: Gamificação no radar de trânsito.



Fonte: Dissertação de Barroso [11].

Outro projeto realizado utilizando algumas técnicas de gamificação foi o chamado *Urinal Fly*, onde o objetivo era reduzir a urina em locais indesejados de banheiros masculinos em aeroportos. Para isso, foram incluídos adesivos de moscas em lugares estratégicos dos mictórios para fazer com que os homens atingissem elas, o que segundo relatores realmente foi efetivo, como pode-se ver a seguir:

Figura 9: Adesivo de mosca do *Urinal Fly*.



Fonte: Dissertação de Barroso [11].

Em relação a utilização de técnicas de jogos para resolver problemas reais, Barroso entra no conceito de *Serious Games*, explicando que o pesquisador Raph Koster classifica os jogos como algo que ensina por meio da diversão, tendo como consequência a superação de desafios propostos. Ele fala que cada atividade repetida várias vezes treina o usuário e ao aumentar a dificuldade, ele aprimora suas

habilidades. Dessa forma, há diversas possibilidades, desde atividades da área pedagógica até treinamento militar. Essa união de um objetivo sério com um jogo é chamada de *Serious Game*.

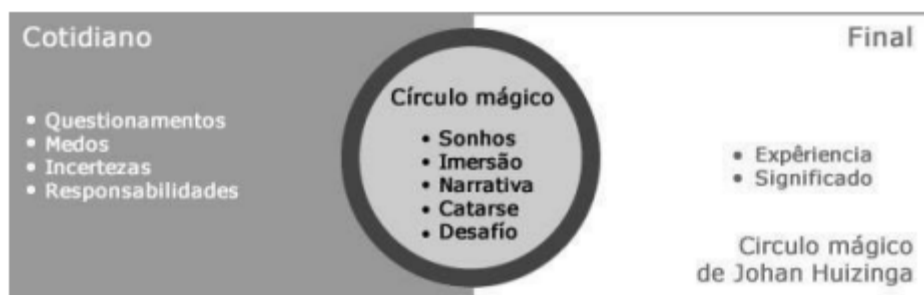
A ideia, nos últimos anos, é de que os jogos cada vez mais consigam interferir na vida cotidiana, de forma a usar cada vez mais as tecnologias para organizar as atividades cotidianas, por exemplo. Misturar o que mais é agradável nos jogos com o que mais é desejado na vida real. Um caso real que podemos ver isso seria por meio do *Quest to Learn*, projeto nos Estados Unidos que busca auxiliar a educação pública por meio de novas formas de assimilar o conteúdo educacional.

Dessa forma, podemos dizer que recompensas por realizar tarefas (como medalhas, pontuações e conquistas) e o incentivo da interação entre usuários fomenta o engajamento das plataformas, como é o caso da *Steam*, uma plataforma de jogos com milhões de usuários.

E não necessariamente a gamificação precisa de um sistema de computador, como é o caso do *Supper Better*, um conjunto de regras criadas e postadas em um blog com o objetivo de auxiliar na recuperação da autora após um acidente. Para isso, ela apenas definiu as regras e postou um texto simples, sem a necessidade de um sistema completo que fizesse tudo automaticamente.

O autor cita um estudo que explica que ao se submeter a uma atividade de entretenimento, o usuário é imerso em um ambiente onde as preocupações ficam dispersas e o foco é o universo de diversão. Dessa forma, é desenhado o círculo mágico para representar o esquema visual do estudo:

Figura 10: Círculo mágico, segundo o autor.



Fonte: Dissertação de Barroso [11].

Partindo do estado inicial (cotidiano) com medos e questionamentos, passando pelo círculo mágico com possibilidades e imersão, até chegar ao final com a experiência adquirida.

Como o produto que ele pretende entregar é relacionado a RPGs, Barroso cita alguns exemplos existentes dessa classe de jogos.

Começando pelo **Chore Wars**, um RPG de realidade alternativa, com o objetivo de apresentar tarefas online que representam tarefas domésticas do cotidiano. Ao invés de jogar com desconhecidos, o usuário joga com pessoas do seu círculo social. Ao iniciar, o indivíduo é levado a criar um avatar, onde já há uma certa conexão com o sistema. Além disso, ele pode formar uma equipe e depois criar as missões no jogo, representando tarefas do mundo real. Essas missões possuem algumas características, como: nome, experiência obtida, descrição, local, pré-requisitos, recompensas, possíveis monstros e status da aventura. Estas tarefas são registradas ao serem concluídas no mundo real, entregando as recompensas conforme definido.

Em seguida ele cita o **Classcraft**, um RPG educacional online, para professores e estudantes jogarem em sala de aula. Neste jogo, os usuários também criam avatares e entram em equipes para a realização de tarefas. Dessa forma, os pontos de vida do jogador podem ser afetados pelo seu comportamento em uma aula. Nesta ferramenta, o professor teria um papel parecido com o ScrumMaster, onde além de transmitir o conhecimento, poderia gerenciar as tarefas executadas pelos alunos.

Por fim, aborda a ferramenta **Habitica**, aplicação online para gerenciar o tempo que busca auxiliar no gerenciamento de tempo. Com uma estrutura de colunas semelhante ao Trello, esta ferramenta apresenta os hábitos que o usuário deseja desenvolver e permite que ele “adicione” ou “subtraia” os pontos de vida de acordo com o seu desempenho na vida real.

Para o desenvolvimento do produto, o autor definiu três camadas de detalhamento: delimitação do objetivo, conhecimento do objetivo e descrição das funcionalidades.

Para iniciar, ele define a parte de Briefing do produto (missão, visão, atividades, mercado-chave, público alvo, valor, vantagem competitiva, entre outros) para então partir para a parte de nomeação e marca deste.

Considerando o público alvo e os dispositivos que possuem (geralmente computadores), verifica-se que é possível utilizar website para permitir o acesso à plataforma. Utilizando as tecnologias padrões do tempo atual, como a comunicação por protocolo HTTP, estruturação por HTML e CSS e ações executadas por JavaScript.

Tendo em vista o que foi descrito na parte das JRAs, o autor definiu alguns pontos que seriam necessários para a plataforma, dentre eles:

Perfil do usuário: Separado em personagem e usuário, um RPG utiliza essas ferramentas para ter referência para a maioria das ações do usuário.

Perfil do personagem: são as características que vão incluir o personagem criado em um grupo específico, podendo apresentar funções, qualidades, defeitos e habilidades diferentes.

Missões: São a forma que os personagens vão interagir com o sistema, enfrentando obstáculos e obtendo recompensas.

Recompensas: Obtidas ao final das missões, ajudam a estimular o usuário a continuar utilizando a ferramenta e completando mais missões.

Por fim, para realizar o controle de entradas e saídas do sistema (e consequentemente o banco de horas de cada usuário no projeto), foi utilizado a tecnologia NFC para identificar e registrar cada entrada do usuário.

Após a conclusão do trabalho, o autor ressaltou que se faz necessária uma validação mais profunda da ferramenta, que por fatores técnicos ficou em estágio inicial de funcionamento. Dessa forma, em trabalhos futuros também é possível que a ferramenta chegue a um estágio mais maduro de funcionamento.

3.3 What a To-Do: Studies of Task Management Towards the Design of a Personal Task List Manager. [2]

O autor inicia citando que a iniciativa considerada pela *Defense Advanced Research Projects Agency* (DARPA) é desenvolver sistemas capazes de ajudar profissionais do governo a gerenciarem melhor suas tarefas de escritório e militares. Os 'assistentes cognitivos' seriam capazes de receber as diretrizes do usuário e guiá-lo para a conclusão das tarefas.

Um exemplo desse tipo de sistema seria sistemas de gerenciamento de lista de tarefas, algo mais simples, que possua: as tarefas diárias, planejar e executar

ações simples, priorizar tarefas, aprender com o usuário, tomar anotações e ideias e responder perguntas com sugestões.

Os autores citam como o mercado para ferramentas de gerenciamento de tarefas está aquecido, aparentando que muitas pessoas se preocupam com essa parte de organização.

Outro ponto que eles ressaltam seria no gerenciamento de informações pessoais, que seria mais voltado para a organização de documentos, arquivos ou notas, facilitando a recuperação eficiente.

Em relação aos materiais encontrados por eles sobre a parte cognitiva, os autores fazem a ressalva de estarem muito focados na parte psicológica e não tanto sobre gerenciamento e planejamento no contexto de trabalho. Sendo assim, a análise de cognição distribuída se assemelha mais com o que buscam no artigo.

Para auxiliar na visão da situação atual, os autores realizam uma pesquisa com os funcionários para listar onde eles realizam a gestão das tarefas atualmente.

Tabela 1 - Número de recursos e tarefas por pessoa, recursos online estão sombreados.

Per Person	Ave	Median	%
No. of resources used to represent to-dos	11.25	11.5	
No. of active to-dos	74.4	65	
In email	26.6	20	35.8
In online calendar	8.6	9	11.6
In paper list or paper notepad	7.2	0	9.7
In online folders	3.5	1.5	4.7
In online special purpose to-do list	3	0	4.0
In PDA calendar and list combined	2	0	2.7
In daytimer/bound notebook/planner	1.6	0	2.2

Fonte: Belloti, Dalal, Good, Flynn, Bobrow e Ducheneaut, 2004 [2].

Segundo os autores, as anotações das tarefas foram feitas pelos entrevistados com o mínimo de esforço, geralmente sem descrever a tarefa, somente algo superficial. Pelo que notaram, o título geralmente não é algo claro como “Enviar e-mail para fulano”, mas algo tão simples que só é efetivo por um curto espaço de tempo (onde a pessoa se lembra ao que se refere ainda). Além disso, das anotações observadas, somente 14% delas estavam em algum formato de lista, segundo a informação apresentada no artigo, o que geralmente daria uma percepção de mais trabalho a ser feito.

Com a pesquisa que os autores realizaram, conseguiram perceber que cerca de metade das tarefas já estão em ambiente online, mesmo sem o suporte de um gerenciador específico. Os principais seriam os e-mails e calendários eletrônicos, que são fracos para esse objetivo.

Tendo em vista isso, os autores citam algumas características importantes para um gerenciador de tarefas com listas: diversas formas de visualizar e manipular as tarefas, estar disponível em um lugar no “caminho” do dia-a-dia do usuário, constantemente acessível para entradas rápidas de tarefas, permitir entradas mais atômicas (sem tanta descrição formal), formas de lidar com tarefas obsoletas de baixa importância (este último que pode ser abordado no meu trabalho).

Em um estudo mais profundo, os autores notaram que parte das tarefas está a uma semana ou mais ali, mesmo a maior parte das estimativas apresentadas pelos usuários estarem entre 10 minutos e 4 horas de conclusão para a tarefa.

Pelo observado, tarefas sem lembretes costumavam demorar mais a serem concluídas, além de geralmente não serem classificadas como tão importantes. O que não necessariamente representa uma correlação direta, algo que os próprios autores citam. Outro ponto destacado que não influencia no final seria o fato de ter ou não um prazo para entrega.

No estudo apresentado, apenas 3% das tarefas foram abandonadas sem algum motivo plausível, o que os autores destacam como fator positivo em relação à priorização de tarefas dos envolvidos. Mesmo assim, os resultados apontaram que as tarefas que não são concluídas em até duas semanas se tornam improváveis de serem cumpridas (considerando que o estudo teve um prazo de duração também).

A conclusão dessa pesquisa, segundo os autores, foi que as pessoas não possuem um desafio em relação à priorização das tarefas, mesmo que eles tenham demonstrado preocupação com isso no início da pesquisa.

Com o estudo feito pelos autores, algumas estratégias para o gerenciamento de tarefa foram notados, geralmente relacionados com as preferências pessoais dos envolvidos e de acordo com suas demandas, conforme eles citaram:

Vistas de tarefas: Uma forma mais abrangente de visualizar as tarefas, de forma a apresentar tudo em uma página, tornando mais visível.

Lista de prioridades informais: Antes de iniciar o trabalho, alguns participantes já listavam as tarefas que pretendiam executar.

Monitoramento de estado: Em alguns casos, havia uma grande quantidade de entidades que precisavam ser monitoradas em relação ao seu estado atual.

Gerenciamento de tempo: Na medida que o tempo se torna mais escasso, é mais importante ainda gerenciá-lo adequadamente.

Alguns pontos não são totalmente relacionados com o gerenciamento de tarefas mas podem impactar em resultados:

Relações sociais (efeito de rede): algumas tarefas podem ser fortemente influenciadas por ajuda externa.

Trabalhando fora da mesa: nem todas as pessoas entrevistadas permaneciam em um escritório, por exemplo, precisando acessar as tarefas “remotamente”.

Ritmo e rotina: algumas tarefas são mais facilmente encaixadas em uma rotina, como responder os e-mails logo pela manhã.

Com esse estudo, os autores notaram que não necessariamente os problemas no gerenciamento de tarefas estão relacionados com uma priorização mal feita, mas sim no esforço que ela requer. Dessa forma, a ferramenta apresentada pelos autores (TaskVista) busca reduzir o esforço em relação a gestão.

O objetivo deles seria apresentar as tarefas utilizando formato de lista, de forma leve e com visualizações intuitivas, podendo trabalhar de forma abstrata e com as características observadas nas formas de anotação atual dos usuários.

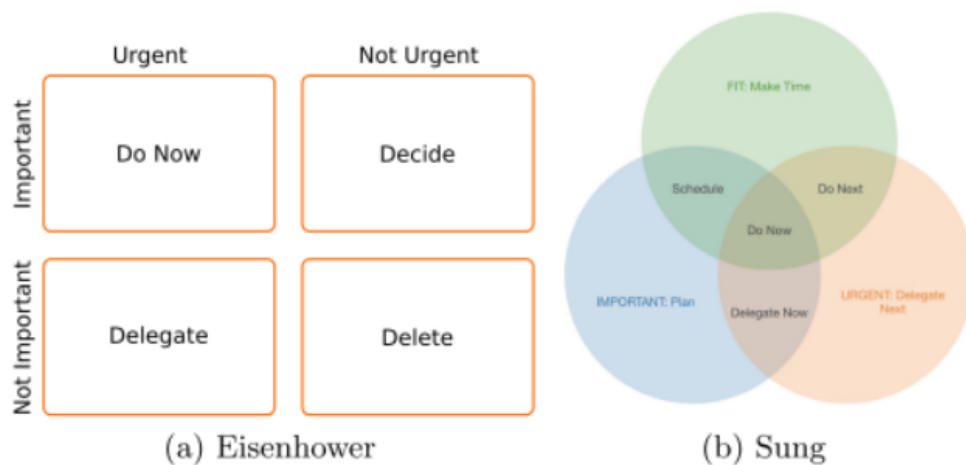
3.4 The Sung Diagram: Revitalizing the Eisenhower Matrix. [12]

Os autores iniciam introduzindo o conceito da Matriz de Eisenhower, que apresenta quatro quadrantes diferentes delimitados pelos eixos de importância e urgência, de forma a ter um quadrante importante/urgente, importante/não urgente, não importante/urgente e não importante/não urgente. Sendo que isso definiria as ações que deveriam ser tomadas para cada tarefa, ou seja, fazer agora, decidir, delegar e apagar, respectivamente.

Porém, eles destacam uma falha nesse método, dizendo que ele não considera os ajustes que podem ser feitos. Para exemplificar, se você irá fazer uma mudança em dois dias, arrumar as coisas para a mudança seria uma tarefa urgente e importante, porém poderia contratar uma empresa de mudanças ao invés de executar a tarefa, como o modelo original sugeriria.

Sendo assim, propõem um novo modelo, com algumas adaptações, que chamam de Diagrama de Sung:

Figura 11: Comparação entre Matriz de Eisenhower e Diagrama de Sung.



Fonte: Artigo de Sung [12]

Para chegar nessa proposta, adicionaram uma nova variável para “ajuste”, dizendo se a entidade (pessoa, empresa, entre outros) é necessária para executar a tarefa. E como agora seriam três variáveis binárias, o diagrama de Venn é mais adequado para a apresentação do que o plano cartesiano.

Sendo assim, apresentaram as três dimensões:

Importância: Que possui significado ou valor, que traga algum benefício para alguém ou algum grupo.

Urgência: É a necessidade de atenção imediata da tarefa. No caso do diagrama de Sung, as tarefas urgentes não podem ter algo bloqueando, ou seja, pode começar imediatamente.

Ajuste: Seria o atributo que os autores julgaram estar faltando na matriz original. Ainda dividem ele em duas partes: se o indivíduo tem a capacidade de

concluir a tarefa (se não precisa de um treinamento, por exemplo); e se há algum objetivo pessoal em realizar a tarefa.

Para ter uma execução do modelo, realizaram a implementação utilizando JavaScript e React, onde cada tarefa possui três campos que podem ser marcados, sendo eles: importância, urgência e “adequação”. Para com isso, apresentar o diagrama de Venn do modelo.

3.5 Considerações

O trabalho analisado na [subseção 3.1](#) apresenta uma boa visão geral do gerenciamento de tarefas e como seria construir um modelo voltado para o desenvolvimento de uma solução. Este aborda os três pilares principais: visualização, otimização e recomendação, que são uma boa base para a solução proposta. Além disso, começa a introduzir o conceito de colaboração, apesar de ainda ficar superficial em comparação com a proposta que realizamos.

Na dissertação apresentada na [subseção 3.2](#) o autor apresenta uma solução interessante para o gerenciamento de tarefas com gamificação, porém, não aborda a parte da colaboração entre indivíduos para a conclusão de uma tarefa além de não apresentar um sistema de priorização que apresente a ordem recomendada de execução.

Já na [subseção 3.3](#) o autor realiza um estudo sobre o gerenciamento de tarefas com um grupo de pessoas, trazendo uma visão de como ocorre em ambientes reais na época do estudo.

Por fim, foi interessante trazer na [subseção 3.4](#) o modelo do Diagrama de Sung que adiciona uma variável a Matriz Eisenhower (fazendo a matriz virar um diagrama de Venn) para aprimorar a priorização de tarefas e cobrir uma falha destacada. O ponto discutido faz bastante sentido, porém na prática pode não surtir tanto efeito quando no escopo de um sistema colaborativo, já que a tarefa pode ser compartilhada com outro usuário para este realizar a execução.

Para uma melhor compreensão, podemos colocar os trabalhos relacionados citados na tabela comparativa abaixo, trazendo um visão melhor do posicionamento do trabalho atual:

Tabela 2 - Comparação dos trabalhos relacionados.

Autor	Gerenciador de tarefas	Gamificação	Colaborativo	Sistema	Diagrama de Sung	Matriz de Eisenhower
BELLOTTI, 2004 [2]	X					
OHMUKAI, 2003 [10]	X		X			
Barroso, 2016 [11]	X	X		X		
SUNG, 2020 [12]	X				X	
Guilherme, 2022	X	X	X	X		X

Fonte: Elaborada pelo autor com base nas pesquisas.

Como podemos observar, o trabalho proposto compila diferentes características de cada artigo. Apenas o artigo de Sung (2020) apresenta um ponto não incluído no modelo proposto por ser uma outra alternativa, já que seria uma extensão da matriz de Eisenhower.

4 Modelo de Gerenciamento de Tarefas

4.1 Conceito geral

Podemos resumir o modelo de gerenciamento de tarefas em uma expressão matemática, representando a relação entre os atributos de uma tarefa (as variáveis da expressão) como um valor que pode ser utilizado na priorização destas. Ou seja, calcular o que podemos chamar de “peso” de cada tarefa, de acordo com o modelo, para então otimizar o conjunto de tarefas em uma ordem mais eficiente de execução. Com isso, ordenar a lista de tarefas por estes pesos e definir o que seria uma ordem mais eficiente de execução destas.

Este modelo pode ser implementado por meio de um algoritmo que nos permite automatizar o cálculo dos pesos e a priorização das tarefas, sem necessidade de verificar uma a uma.

Durante todo o desenvolvimento deste trabalho, foi feita uma implementação incremental do modelo. Sendo assim, partimos de uma versão mais “básica” e aprimoramos para tentar obter o melhor resultado possível e resultando em diferentes versões do modelo, como vamos observar a seguir.

Com isso, para uma primeira versão de um modelo mais básico, consideramos somente a urgência e a importância para calcular o peso, da seguinte forma:

$$p = 2.u + i$$

Onde:

p = Peso da tarefa (pontuação utilizada para a priorização);

u = Urgência;

i = Importância;

Nesse modelo, a urgência fica definida com uma prioridade maior que a importância das tarefas. Foi feito dessa forma considerando que quando duas tarefas possuem valores semelhantes para estes atributos, aquela com uma urgência maior deveria ser realizada antes. Ou seja, uma tarefa com uma urgência

maior tem um peso maior do que uma tarefa com uma urgência menor e um peso semelhante. Consideramos isso para que tarefas mais urgentes tenham um peso maior, justamente pelo fato de precisar ser entregue antes (considerando o próprio conceito de *urgência*).

Para uma análise mais avançada, é necessário considerar alguns atributos a mais, vamos utilizar carga de trabalho e prazo necessário para entrega (que começa a nos levar também para o envolvimento de uma agenda completa), ambos em horas.

Com isso, é possível considerar duas novas variáveis no modelo e aprimorá-lo para uma versão um pouco mais avançada:

$$p = (2 \cdot u + i) / (c / t)$$

Sendo:

c = Carga de trabalho (em horas);

t = Prazo para entrega (em horas);

Nessa expressão, verificamos a proporção do tempo necessário para executar a tarefa e o tempo disponível até o prazo de entrega. Depois disso, verificamos como ficaria a razão do peso da tarefa em relação a essa proporção do tempo necessário para executá-la.

A relação entre o tempo disponível e o tempo necessário para executar a tarefa (uma estimativa) deve ser menor que 1 para que a tarefa seja considerada possível de ser concluída, portanto, se esse valor for maior que 1 podemos dizer que ela já está atrasada ou que pelo menos a chance dela ser entregue antes do prazo é menor que as demais.

Com isso, tarefas que estão com um prazo “mais apertado” possuem um peso maior, priorizando elas para serem executadas antes (no caso do surgimento de algum imprevisto, já que possui mais tempo para algo dar errado, aumentaria o espaço para ajustes).

Desta forma, já temos a ideia macro do modelo de priorização, aplicando o conceito da matriz de Eisenhower na priorização das tarefas em relação a ordem de execução destas. Além disso, podemos considerar pesos diferentes em cada variável utilizada na expressão matemática para tentar melhorar os resultados. Ou

até considerar algumas condições a mais, como por exemplo, para tarefas com valores semelhantes utilizar a razão da carga e do prazo como um “desempate” para definir quais seriam priorizadas.

Certamente há variáveis que não foram abordadas, como em qualquer modelo, e que podem influenciar na priorização de tarefas para cada contexto e para cada indivíduo diferente. Por exemplo, poderíamos ainda utilizar um atributo na tarefa para indicar se ela pode ser terceirizada para outra pessoa e com isso ela possuiria uma carga de trabalho menor (apenas o trabalho de passar a tarefa para um outro indivíduo), como vimos no [Sung Diagram](#) (SUNG, et al., 2020).

4.1.1 Gamificação

O peso resultante no modelo serve como entrada para os elementos de gamificação que podem ser construídos, como um fator de influência em medidores de desempenho do indivíduo.

Por exemplo, podemos considerar que o indivíduo possui uma “energia” (que muitos jogos também chamam de “vida”) que aumenta ou diminui de acordo com a carga de trabalho de cada tarefa que é entregue antes ou depois do prazo. Ele também possui uma “experiência” (que muitos jogos utilizam para indicar o “nível” do usuário), que é alterada conforme a entrega das tarefas e de acordo com o peso destas.

Com esses indicadores de desempenho, podemos relacionar o indivíduo com outros e apresentar um *ranking* daqueles com melhor desempenho em um período de tempo (última semana, último mês, este ano, entre outros). Como por exemplo, aqueles que obtiveram mais energia ou experiência, ou até que concluíram mais tarefas ou tarefas com pesos maiores.

De qualquer forma, o modelo entrega os valores necessários para a aplicação de elementos de gamificação, como vimos.

4.2 Forma de implementação

Para automatizar o cálculo de uma expressão matemática (para não ser necessário calcular manualmente para cada situação) utilizamos um algoritmo desenvolvido com o objetivo de aplicar o modelo.

Esse algoritmo recebe um conjunto de tarefas, realiza o cálculo do peso destas e resulta em uma lista ordenada conforme a avaliação obtida. Algo extremamente simples e que nos remete ao conceito de sistema de informação com uma entrada para um processamento que gera uma saída, segundo as funções de um sistema de informação conforme Gouveia e Ranito (2004).

Com essa lista ordenada e uma parametrização simples indicando a quantidade de horas disponíveis por dia para a execução das tarefas (padrão de 8 horas diárias), é possível realizar a distribuição das tarefas nos dias, formando uma agenda para o indivíduo.

Caso antes de criar essa agenda exista uma agenda pré-definida com eventos existentes já (considerados como espaços de tempo indisponíveis), é possível encaixar as tarefas conforme o restante do tempo disponível.

Porém, para iniciar a avaliação do nosso modelo, vamos desenvolver o algoritmo necessário para aplicá-lo (como gostaríamos de comparar com o resultado de outros modelos, vamos implementá-los também) e calcular algumas métricas em cima do resultado obtido, para conseguir avaliar o real efeito da utilização do modelo de priorização de tarefas proposto.

4.3 Métricas de avaliação

Em um caso real de execução de um conjunto de tarefas, um indivíduo pode identificar diversas variáveis que influenciam na priorização das tarefas ou na indicação se uma priorização seria melhor que outra.

No entanto, para o escopo dos experimentos que vamos precisar nos limitar à variáveis quantitativas e que possam ser calculadas baseadas nos atributos existentes das tarefas.

Sendo assim, para os experimentos realizados, vamos considerar duas métricas principais:

- Tempo livre: O total de horas entre a entrega teórica da tarefa e o prazo de entrega que ela possuía, representado em horas. Para tarefas consideradas atrasadas, esse valor é negativo;
- Entrega no prazo: O percentual de tarefas entregues antes do prazo no conjunto de tarefas. Como vamos realizar todos os testes em conjuntos de

tarefas (já que não faria sentido comparar uma tarefa apenas), gostaríamos de saber quantas tarefas a mais ou a menos poderiam ser entregues dentro do prazo, teoricamente;

Com essas métricas, já é possível avaliar o resultado do modelo de priorização em um conjunto de tarefas e realizar alguns experimentos de forma automática para analisar o desempenho dos modelos e medir a variação média das métricas. Outra medida que podemos observar, ao testar em mais de um conjunto de tarefas, é a quantidade de conjuntos em que o modelo se sobressai aos demais em ambas as métricas quantitativas.

Além disso, é possível utilizar uma métrica qualitativa para a avaliação de um modelo, como por exemplo “o indivíduo acredita que teria um desempenho pior, igual ou melhor utilizando essa priorização?”. Para isso, seria necessário o envolvimento de mais indivíduos e o desenvolvimento de um sistema para que possam visualizar o resultado e avaliá-los. Sendo assim, o indivíduo indicaria qual seria o modelo com melhor desempenho considerando as variáveis que ele mesmo define, de forma específica para cada caso.

4.4 Experimentos realizados

Para os experimentos que vamos realizar, vamos considerar quatro modelos, sendo eles:

- Sem modelo: Considerando apenas a urgência das tarefas;
- Modelo básico: Considerando o modelo com a relação apenas entre urgência e importância;
- Modelo avançado: Considerando mais atributos, como a carga de trabalho e o prazo de entrega;
- Modelo alternativo: Durante o desenvolvimento dos experimentos, conseguimos estender o modelo avançado para uma versão aprimorada, colocando pesos diferentes e calculando a urgência com base na carga de trabalho e no prazo de entrega.

Para conseguir aplicar os modelos precisamos de conjuntos de tarefas a serem priorizadas, com alguns atributos como título (para visualização e identificação de quem executa os testes), importância, urgência, carga de trabalho e prazo de entrega.

Com o intuito de avaliar diferentes cenários, vamos executar três experimentos que variam de acordo com a origem das tarefas:

- Tarefas reais: Durante alguns dias, foram anotadas as tarefas realizadas em ambiente de trabalho para a construção de conjuntos reais, com o objetivo de representar um cenário real. Por motivos de sigilo, o título foi adaptado para ser mais genérico, mas mantendo os dados realmente relevantes. Sendo assim, podemos fazer uma comparação direta da agenda realizada com o resultado da sugestão dos modelos.
- Tarefas aleatórias: Como não seria possível levantar muitos conjuntos de tarefas reais, foram geradas algumas tarefas com os valores aleatórios para simular um cenário diferente e sem influência de quem estaria levantando as tarefas reais, algo mais imparcial.
- Lote de conjuntos de tarefas aleatórias: Após avaliar o resultado em conjuntos isolados, podemos realizar um experimento com a avaliação de uma quantidade maior de conjuntos de tarefas, geradas aleatoriamente, de forma agregada. Com isso, conseguimos avaliar a diferença média de cada modelo para a priorização sem modelo (a ordem originalmente levantada para o conjunto de tarefas) em uma escala muito maior, obtendo uma medida de variação do desempenho média.

Estes experimentos variam conforme a complexidade de sua realização e, por consequência, melhorando a qualidade das medidas obtidas para realizar a avaliação dos resultados do desempenho de cada modelo.

Com isso, conseguimos obter resultados para diferentes modelos em diferentes cenários e comparar as métricas quantitativas para identificar se houve ou não uma melhora na priorização de tarefas proposta. Ou seja, para cada experimento, observamos cada métrica em cada modelo e comparamos a métrica obtida com uma priorização sem modelos, nos trazendo uma medida de quanto aquele modelo melhora ou piora a métrica para aquele conjunto de tarefas.

Além destes três experimentos, seria importante testar o modelo de forma qualitativa por meio de avaliações de usuários reais com conjuntos de tarefas reais. Porém, por falta de tempo hábil, não seria possível fazer esse tipo de experimento, nos levando a focar nos experimentos levantados com dados mais reduzidos ou com dados aleatórios.

4.5 Análise dos resultados

Após executar os experimentos citados anteriormente, obtivemos resultados quantitativos do desempenho de cada modelo para analisar o resultado conforme as métricas obtidas em cada um e para cenários diferentes.

Conseguimos então comparar as métricas, por serem quantitativas, possibilitando uma análise da melhora obtida entre diferentes modelos e identificando quanto realmente o uso do modelo de priorização de tarefas proposto neste trabalho pode melhorar a eficiência de um indivíduo na realização de suas tarefas.

Porém, ainda resta a análise qualitativa, onde o dono da lista de tarefas iria realmente identificar se a agenda obtida realmente faz sentido para sua realidade e traz benefícios para ele. Sendo assim, seria uma análise mais pontual onde cada usuário pode identificar critérios diferentes que adequam-se melhor ao seu caso. Este experimento, por precisar de uma interação com usuários, necessita do desenvolvimento de um sistema (descrito na seção 5).

4.5.1 Tarefas reais

Para a realização dos experimentos, foram registradas as tarefas realizadas no ambiente corporativo durante dois períodos de uma semana cada. No primeiro conjunto de dados [Apêndice A], todos os modelos aplicados tiveram 100% de sucesso nas tarefas (todas entregues à tempo). Porém o modelo avançado teve um ganho no tempo livre, tendo um ganho de 2% de tempo livre em relação a priorização sem modelo e o modelo avançado demonstrou 6% de melhora. Como podemos observar a seguir:

Tabela 3 - Resultado do primeiro experimento, métrica de tempo livre.

Modelo	Tempo livre (h)
Sem modelo	1253h
Básico	1269h
Avançado	1279h
Alternativo	1330h

Fonte: Elaborado pelo autor (2022).

Neste caso, essa diferença do que chamamos de tempo livre não possui um efeito específico no resultado final da execução das tarefas. Porém, conseguimos citar dois benefícios que isso poderia apresentar:

Primeiramente, mais tempo entre a realização de uma tarefa e o prazo de entrega desta permitiria um espaço maior para melhorias e ajustes necessários após a realização da tarefa. Por exemplo, uma tarefa para a escrita de um relatório foi finalizada com 24h de antecedência, isso significa que durante estas 24h seria possível revisar, pedir para um terceiro revisar, ajustar e melhorar o resultado, entre outras atividades neste sentido. No entanto, caso tenha sido finalizada com 1h ou menos de antecedência, basicamente não abre espaço para melhorias. Fora que, como o início da tarefa foi definido pensando no tempo que levaria e com o prazo de entrega, qualquer erro ou falha que atrase ou aumente o tempo necessário para a realização da tarefa iria atrasar a entrega desta.

O segundo ponto, e mais subjetivo, seria relacionado a pressão para a realização de uma tarefa com o prazo para a entrega mais apertado. Ter esse tempo a mais traria benefícios para o período de execução, mesmo não afetando o resultado final da entrega, propriamente.

No segundo conjunto de dados [Apêndice B] foi observada uma agenda mais cheia, onde muitas tarefas não foram entregues no prazo. Sendo assim, houve uma variação maior na taxa de sucesso, que no modelo avançado chegou a ser 87% superior do que a priorização sem modelos. Enquanto o modelo alternativo obteve o dobro de sucesso ao ser aplicado.

Tabela 4 - Resultado do primeiro experimento, métrica de taxa de entrega no prazo.

Modelo	Entrega no prazo (%)
Sem modelo	27,6%
Básico	24,1%
Avançado	51,7%
Alternativo	55,2%

Fonte: Elaborado pelo autor (2022).

Com este resultado, percebemos que a versão final e mais completa (modelo alternativo) consegue priorizar as tarefas de uma forma que mais tarefas sejam entregues dentro do prazo de entrega, em comparação aos outros modelos ou até em comparação a uma priorização sem modelo (simulando o que um indivíduo faria normalmente sem utilizar um modelo para priorização).

O resultado dessa parte do primeiro experimento já demonstra o potencial do modelo proposto, de conseguir aumentar a eficiência na execução de uma lista de tarefas. No entanto, como os conjuntos utilizados foram levantados manualmente, os resultados podem ser considerados parciais. Por causa disso, os outros experimentos foram realizados com dados aleatórios.

4.5.2 Tarefas aleatórias

Neste experimento, utilizamos um conjunto de tarefas com dados aleatórios [Apêndice C]. Ou seja, atributos como urgência, importância, carga de trabalho e prazo de entrega não foram definidos anteriormente, foram gerados no momento de execução por algoritmos aleatórios (ou pseudo-aleatórios, conforme o dilema conhecido da computação).

Dessa forma, cada atributo utilizado no método para calcular o peso de cada tarefa foi definido aleatoriamente e a ordem teórica de execução das tarefas na priorização sem modelo foi a ordem de criação das tarefas aleatórias. Para os demais modelos a ordem teórica de execução das tarefas, assim como a data de conclusão de cada uma (utilizada para calcular a métrica de tempo livre), foi obtida considerando a ordem priorizada das tarefas conforme seus pesos, ou seja,

considerou-se que uma tarefa seria executada após a outra conforme foram priorizadas pelo modelo.

Durante este experimento, os modelos avançado e alternativo apresentaram resultados melhores nas métricas observadas, como podemos verificar no exemplo a seguir:

Tabela 5 - Resultado do segundo experimento.

Modelo	Entrega no prazo (%)	Tempo livre (h)
Sem modelo	62,5%	247h
Básico	52,5%	43h
Avançado	67,5%	374h
Alternativo	62,5%	477h

Fonte: Elaborado pelo autor (2022).

Neste caso, a entrega no prazo do modelo alternativo não variou em relação a priorização sem modelo, porém o tempo livre obtido foi 93,11% maior. Enquanto isso, o ganho de tempo livre do modelo avançado foi um pouco menor (51,41%) porém com uma taxa de entrega no prazo um pouco maior, com um aumento de 8% ou 5 pontos percentuais.

Curiosamente, durante o desenvolvimento dos experimentos, foi possível notar essa características dos modelos, de que o modelo avançado geralmente apresentava uma taxa de tarefas entregues no prazo maior, enquanto o modelo alternativo apresentava um tempo livre maior. Porém, na maioria dos casos, o modelo alternativo apresentou resultados melhores que os demais em ambas as métricas (como vamos observar no terceiro experimento). Podemos, então, considerar que este é o modelo com melhores resultados.

4.5.3 Lote de conjuntos de tarefas aleatórias

Os experimentos realizados anteriormente cobriam apenas um conjunto por vez, abrindo espaço para afirmar que o resultado obtido foi bom por causa dos dados utilizados. Para contornar isso, a ideia deste experimento é executá-lo em

maior escala, verificando o desempenho em um lote com vários conjuntos de tarefas, geradas aleatoriamente da mesma forma que o experimento anterior.

Para tal, vamos observar o desempenho de cada modelo para cada um dos 100.000 conjuntos de 40 tarefas cada, verificando as métricas utilizadas anteriormente e verificando qual a variação média dos modelos em comparação a uma priorização sem modelos. Ou seja, basicamente observar qual seria a variação média das métricas caso o indivíduo passe a usar um dos modelos de priorização.

A seguir, podemos observar a métrica de tempo livre e o desempenho de cada modelo em relação a ela:

Tabela 6 - Resultado do terceiro experimento, medidas da métrica de tempo livre.

Modelo	Casos com maior tempo livre	Varição em relação ao padrão (sem modelo)
Sem modelo	2.250	N/A
Básico	7	55,8%
Avançado	28	17,6%
Alternativo	97.769	442,8%

Fonte: Elaborado pelo autor (2022).

O modelo alternativo se demonstrou com os melhores resultados no tempo livre em mais de 97% dos casos, com uma variação média de 442,8% a mais de tempo livre do que uma priorização sem modelo. Um resultado surpreendente, que nos demonstra que o resultado obtido nos outros experimentos na real foram abaixo da média geral.

A seguir, podemos observar a métrica da taxa de entrega no prazo e o desempenho de cada modelo em relação a ela:

Tabela 7 - Resultado do terceiro experimento, medidas da métrica de taxa de entrega no prazo.

Modelo	Casos com melhor taxa de entrega no prazo	Varição em relação ao padrão (sem modelo)
Sem modelo	12.150	N/A
Básico	7.119	0,7%
Avançado	26.956	4,7%
Alternativo	73.626	15,06%

Fonte: Elaborado pelo autor (2022).

Neste caso, conseguimos observar que o modelo alternativo foi superior aos demais em mais de 73% dos casos, com uma média de 15% a mais de tarefas sendo entregues dentro do prazo. Ou seja, de fato ao utilizar o modelo de priorização, na média, houve um ganho de eficiência na entrega das tarefas, com mais tarefas sendo entregues dentro do prazo.

Como resultado final, observamos as ocorrências em que cada modelo se saiu melhor que os demais em ambas as métricas citadas anteriormente, como podemos ver a seguir:

Tabela 8 - Resultado do terceiro experimento, agregado das métricas.

Modelo	Casos com melhor resultado em ambas as métricas
Sem modelo	1.518
Básico	5
Avançado	27
Alternativo	72.741

Fonte: Elaborado pelo autor (2022).

Como observado, o modelo alternativo se saiu melhor que os demais em ambas as métricas em mais de 72% dos casos. Sendo assim, acredito que há espaço para citar que o modelo proposto apresenta uma melhoria na execução de um conjunto de tarefas, com um aumento médio da eficiência.

É interessante ressaltar que essas análises consideraram as duas métricas quantitativas que foram possíveis de calcular com os valores existentes. Uma

análise mais a fundo em um conjunto de tarefas reais para um indivíduo pode ter resultados diferentes de qual seria o melhor modelo por envolver outras variáveis mais específicas para cada caso ou que sejam qualitativas (ex: a preferência do indivíduo de executar uma tarefa antes da outra, uma tarefa depender da conclusão de outra tarefa ou até de terceiros, entre outras variáveis que poderiam influenciar em uma análise mais qualitativa).

5 Desenvolvimento do sistema web

Na seção anterior, analisamos os resultados das métricas quantitativas e as medidas de variações de desempenho obtidas com modelo para diferentes conjuntos de tarefas. No entanto, para conseguir realizar um experimento com métricas qualitativas, vamos precisar de indivíduos avaliando o resultado da priorização de cada modelo.

Para isso, a ideia é desenvolver um sistema web para disponibilizar de maneira fácil tanto a visualização das priorizações quanto para avaliar os resultados e indicar aquele que teria o melhor desempenho segundo a avaliação do usuário. Levando o modelo proposto para uma situação real onde o usuário pode priorizar suas tarefas e executá-las com as melhorias vistas nos experimentos anteriores.

Sendo assim, a plataforma precisa permitir que o usuário tenha a capacidade de cadastrar um conjunto de tarefas, visualizar o resultado da priorização de cada modelo e conseguir avaliar o resultado, pelo menos.

Nas próximas seções, vamos descrever como é a estrutura necessária para este sistema, abordando a parte técnica e conceitual da sua arquitetura.

5.1 Tecnologias utilizadas

Até então, detalhamos e descrevemos a lógica por trás de um modelo de priorização de tarefas com o intuito de ser mais eficiente e produtivo no dia-a-dia. Agora iniciaremos a produção do sistema que deve implementar e facilitar o acesso de usuários comuns ao modelo.

Buscando entregar a melhor solução no tempo disponível, vale ressaltar que parte do processo de decisão das tecnologias foi influenciada pelas experiências prévias nas tecnologias e conhecimento já existente (que economizaria tempo de aprendizado no desenvolvimento).

5.1.1 Servidor

O servidor da aplicação será responsável por todo o processamento das tarefas e aplicações do modelo de priorização. Dessa forma, linguagens como

JavaScript perdem um pouco da prioridade por não lidar bem com números de ponto flutuante (segundo Kaczorowski, 2022), fazendo mais sentido utilizar linguagens como Python ou PHP.

Como boa parte dos processos são aplicações de operações matemáticas, a linguagem Python se sobressai pela existência de bibliotecas voltadas para tais atividades, como o pandas (segundo a própria documentação da biblioteca [14]). Além disso, há bibliotecas que auxiliam no desenvolvimento web e uma comunidade grande que auxilia na resolução de problemas, como é descrito por Kaczorowski (2022), que permite uma prototipação mais rápida.

5.1.2 Plataforma web

Atualmente, há diversos frameworks disponíveis para o desenvolvimento frontend, que são ferramentas para auxiliar o programador e abstrair funções e comportamentos comuns entre projetos, ainda mais se comparado com a situação de alguns anos atrás.

Como a solução a ser desenvolvida não é muito complexa (em relação à interfaces das telas), alguns dos critérios para a tecnologia desejada são justamente o tamanho, a velocidade (desempenho) e complexidade desta.

Dentre as principais (mais utilizadas) do contexto atual, encontra-se: React, Vue e Angular. Dentre estas, React e Vue são mais recomendados para projetos menores e possuem um tempo para prototipação menor, segundo o SSA Group (2021). Por ser de conhecimento do estudante, optou-se por VueJS.

5.2 Modelagem

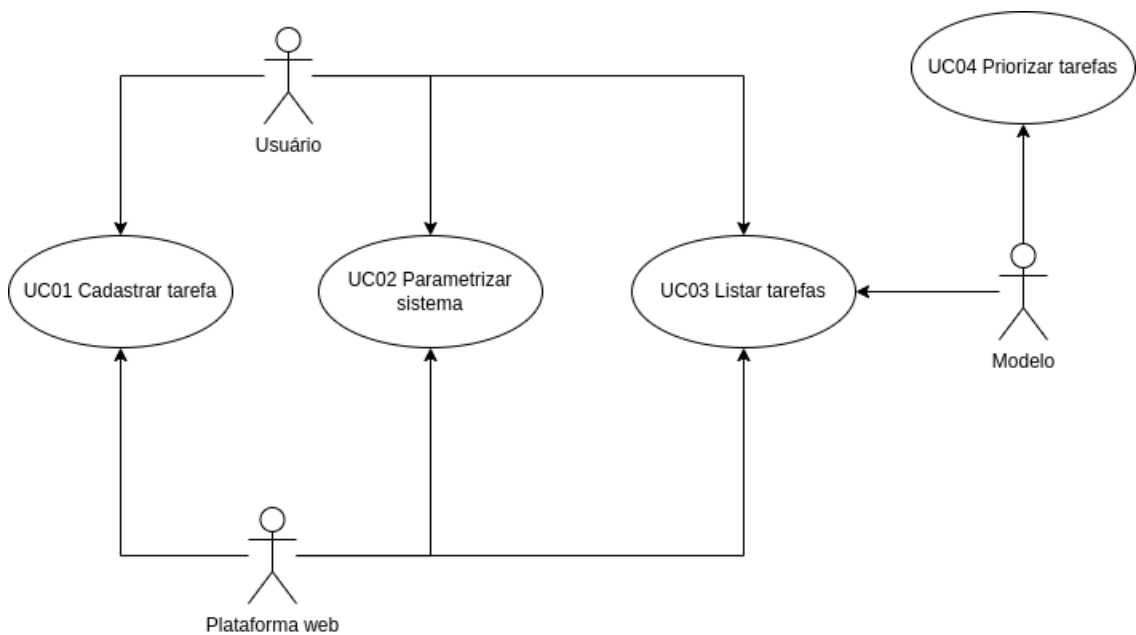
Como o intuito do sistema web é apenas tornar acessível o modelo de priorização proposto, a estrutura dele é relativamente simples, com poucas classes e funções.

Para simplificar o entendimento da estrutura do sistema, podemos utilizar alguns diagramas para representar visualmente, como para caso de uso, diagrama de atividade e de classes.

5.2.1 Casos de uso

Para tentar deixar mais claro a estrutura desenhada, começamos apresentando o diagrama de casos de uso geral da plataforma. Neste caso, a implementação do método proposto para priorização fica centralizado no ator Modelo:

Figura 12: Diagrama de casos de uso.



Fonte: Elaborado pelo autor (2022).

UC01 Cadastrar tarefa

Descrição: Quando o usuário deseja cadastrar uma nova tarefa na plataforma web.

Ator principal: Usuário.

Pré-condições: O usuário está logado na plataforma.

Fluxo básico:

1. O usuário navega até a tela de cadastro de tarefas.
2. O usuário preenche os campos adequadamente, conforme seu próprio critério dos valores que deve informar para cada tarefa.
3. O usuário envia a tarefa para a plataforma.
4. A plataforma valida a tarefa recebida.
5. Tarefa cadastrada com sucesso, o usuário pode visualizá-la nas listagens.

UC02 Parametrizar sistema

Descrição: O usuário deseja alterar os parâmetros da plataforma web, adaptando-se melhor a sua necessidade, alterando o seu nome apresentado na plataforma e o modo do design (modo escuro ou claro).

Ator principal: Usuário.

Pré-condições: O usuário está logado na plataforma.

Fluxo básico:

1. O usuário navega até a tela de configuração da plataforma web.
2. O usuário analisa as listagens de tarefas nos diferentes modelos.
3. O usuário indica qual modelo priorizou de maneira mais efetiva suas tarefas.
4. Caso deseje, o usuário pode alterar as configurações de design da plataforma. Caso não deseje, segue para o passo 5.
5. O usuário salva as alterações.
6. A plataforma registra as preferências do usuário.
7. A plataforma passa a utilizar o modelo solicitado para a listagem de tarefas.

Fluxo alternativo: Ausência de tarefas.

1. Caso o usuário não tenha tarefas, o passo 2 seria uma descrição da lógica utilizada em cada modelo.
2. O usuário indica aquele que tem a lógica que mais lhe agrada.
3. Segue para o passo 4.

UC03 Listar tarefas

Descrição: O usuário deseja visualizar as suas tarefas conforme o modelo de priorização selecionado.

Ator principal: Usuário.

Pré-condição: O usuário deve estar logado na plataforma.

Fluxo básico:

1. O usuário navega até a tela de listagem de tarefas.
2. A plataforma entrega a lista priorizada conforme as preferências do usuário.
3. O usuário analisa a lista entregue.
4. O usuário pode começar a executar as tarefas conforme desejar.

Fluxo alternativo: Ausência de tarefas.

1. Caso o usuário não tenha tarefas, a plataforma apresenta a opção de cadastrar uma nova tarefa.

2. Caso o usuário cadastre uma tarefa, segue para o passo 2.
3. Caso o usuário não cadastre nenhuma tarefa, encerra o fluxo.

UC04 Priorizar tarefas

Descrição: Adicionar o peso para cada tarefa conforme os diferentes modelos.

Ator principal: Modelo.

Pré-condição: A existência de tarefas na base de dados.

Fluxo básico:

1. O modelo lista todas as tarefas ativas.
2. Para cada tarefa, o modelo calcula o peso de acordo com cada modelo de priorização.
3. O peso de cada tarefa em cada modelo fica salvo para facilitar buscar posteriores.

Fluxo alternativo: Tarefa sem dados completos.

1. Caso alguma tarefa não tenha os dados completos, o modelo ignora ela e segue para o passo 2 até finalizar.

Estes casos de uso demonstram que a ideia de todo o funcionamento fica ao redor da análise de uma listagem de tarefas (como é o objetivo da plataforma, para facilitar um experimento qualitativo com o modelo), para isso seria necessário cadastrar as tarefas utilizadas e configurar a plataforma para se adequar ao usuário, por exemplo.

5.2.2 Diagrama de atividades

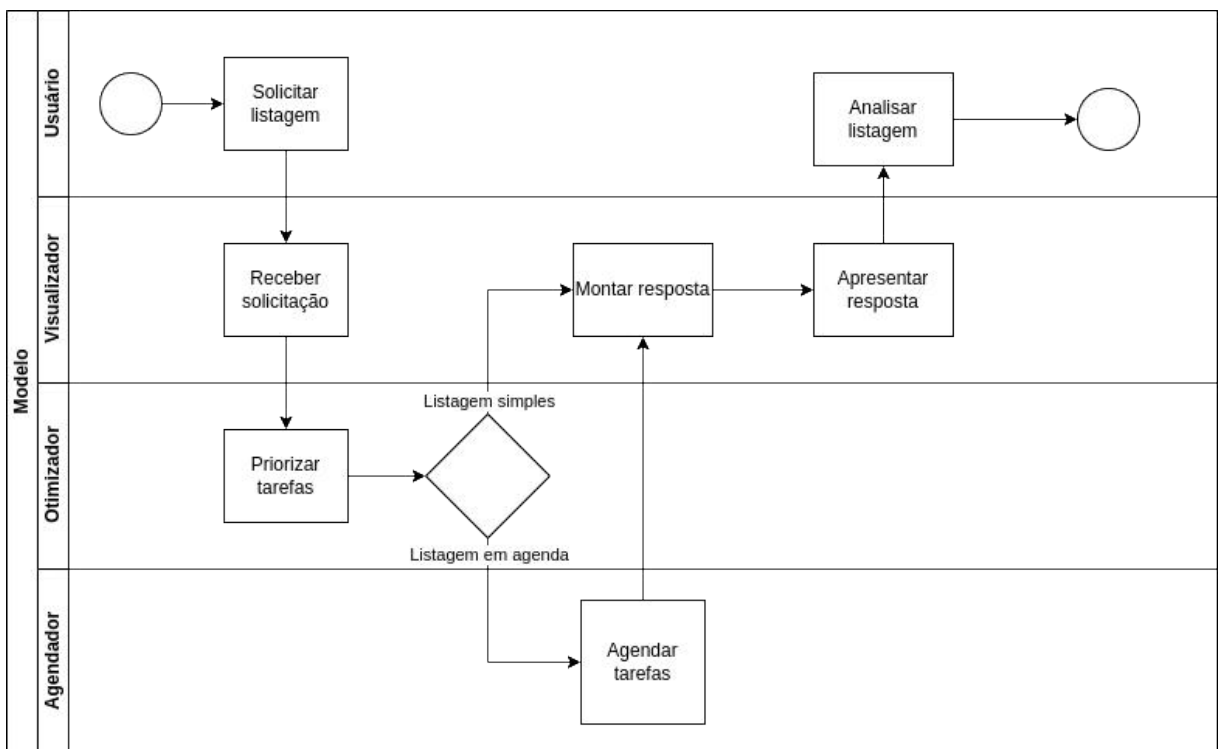
Os casos de uso que vimos anteriormente passam uma visão geral das funções principais do sistema, mas para ficar mais claro, podemos descrever um diagrama de atividades para a principal função do sistema: a listagem das tarefas de forma priorizada. Porém agora com uma visualização mais detalhada da implementação do modelo e as partes envolvidas.

Estendendo a ideia de Ohmukai e Takeda (2003), é possível separar o modelo de priorização em três principais partes: Visualizador, Otimizador e Agendador.

O visualizador é responsável por apresentar as tarefas de acordo com o formato necessário. O otimizador para priorizar as tarefas de acordo com o modelo que deve ser utilizado em cada situação (conforme parametrização do usuário por exemplo). E o agendador deve distribuir as tarefas em uma agenda de fato, com a separação dos horários para cada tarefa.

A seguir, podemos observar de uma forma mais simplificada como seria o principal fluxo de interação entre essas partes e o usuário:

Figura 13: Diagrama de atividades.



Fonte: Elaborado pelo autor (2022).

Conforme o diagrama, a atividade começa com uma solicitação do usuário enviada ao Visualizador, que por sua vez, envia a requisição para o Otimizador, juntamente com os parâmetros de priorização selecionados pelo usuário (o modelo desejado). Com esses dados, o otimizador busca as tarefas do usuário e prioriza elas conforme o modelo desejado.

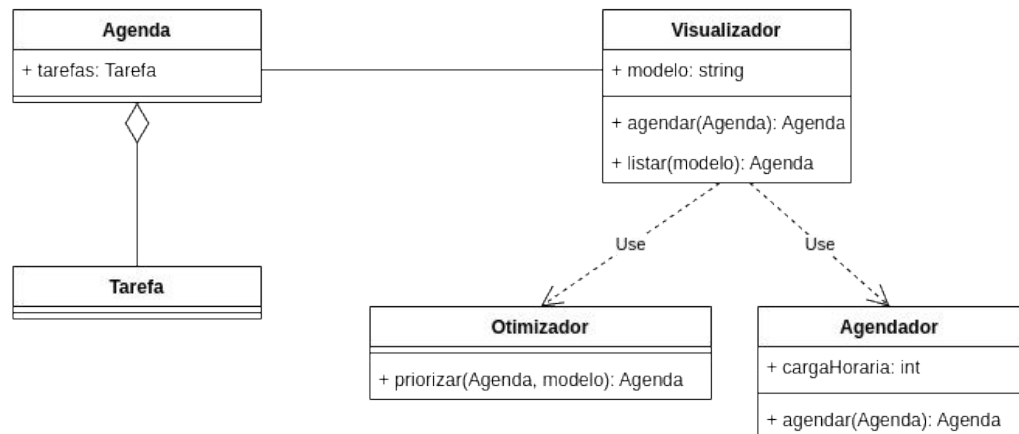
Caso a solicitação seja para apresentar as tarefas em um formato de agenda, o otimizador envia para o Agendador distribuir essas tarefas.

Com a listagem pronta, seja ela em agenda ou não, o visualizador recebe novamente e prepara a resposta adequada, apresentando-a ao usuário que enfim pode analisar a resposta obtida.

5.2.3 Diagrama de classes

Agora que já entendemos melhor como é o funcionamento do sistema por trás da plataforma e como funciona a análise de uma lista de tarefas, fica mais clara a descrição das principais classes envolvidas:

Figura 14: Diagrama de classes.



Fonte: Elaborado pelo autor (2022).

Sendo assim, temos as Agendas compostas por Tarefas. E do outro lado, o Visualizador que se utiliza o Otimizador e o Agendador para preparar as listagens das tarefas conforme cada situação.

5.3 Elementos de gamificação

Como foi descrito anteriormente, o modelo classifica as tarefas de acordo com alguns valores que podem ser utilizados para a implementação de alguns elementos de gamificação no sistema.

Para estimular a conclusão das tarefas dentro do prazo, podemos criar alguns indicadores para representar o desempenho do usuário dentro da plataforma.

Com isso, conseguimos apresentar um estímulo para uma constância do usuário no acesso e utilização da plataforma, que também aumentaria a sua eficiência na conclusão das tarefas.

Para isso, podemos utilizar duas pontuações que aumentam ou diminuem conforme o usuário executa ou deixa de executar uma tarefa dentro do prazo.

A primeira, que chamamos de experiência, estaria relacionada com o peso de cada tarefa e delimita um nível para o usuário. Este valor é acrescentado ou diminuído pelo peso de cada tarefa, em uma paridade de um para um. Ou seja, uma tarefa com peso calculado de 10 e concluída antes do prazo de entrega acrescenta 10 “pontos” na experiência do usuário.

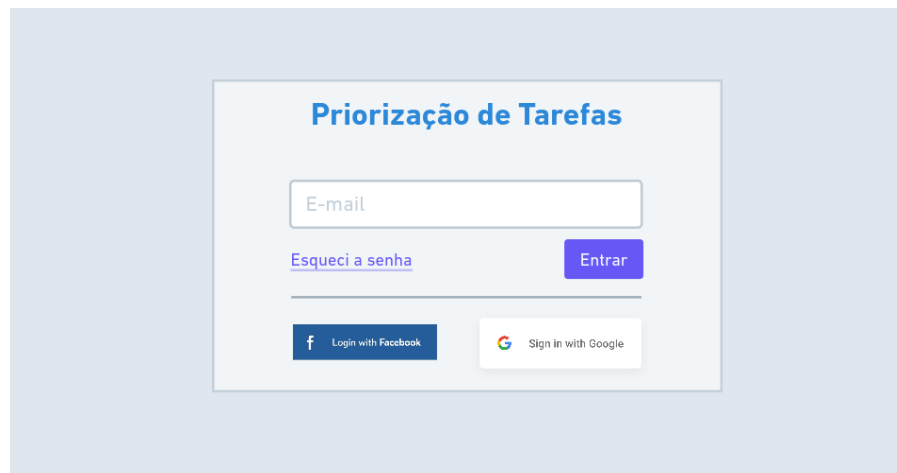
A segunda, que chamamos de energia, está relacionada com a carga de trabalho de cada tarefa e pode estar relacionada também com algumas bonificações (avisar que ele ganhou muita energia no dia ou semana e merece um descanso, por exemplo). Neste caso, alteramos o valor de energia conforme a carga de trabalho em horas de cada tarefa, em uma paridade de um para um, de forma semelhante a experiência.

Com esses indicadores, apresentamos um *ranking* listando os usuários com melhor desempenho, por exemplo. Visando estimular uma competição entre os indivíduos e abrindo espaço para bonificação para os melhores colocados.

5.4 Estudo da interface

Com os elementos do sistema definidos, iniciamos a descrição das telas que a plataforma web precisa. Nesse momento, a ideia é ter um wireframe para conseguir visualizar como a plataforma deveria funcionar, auxiliando no entendimento do sistema e o seu funcionamento.

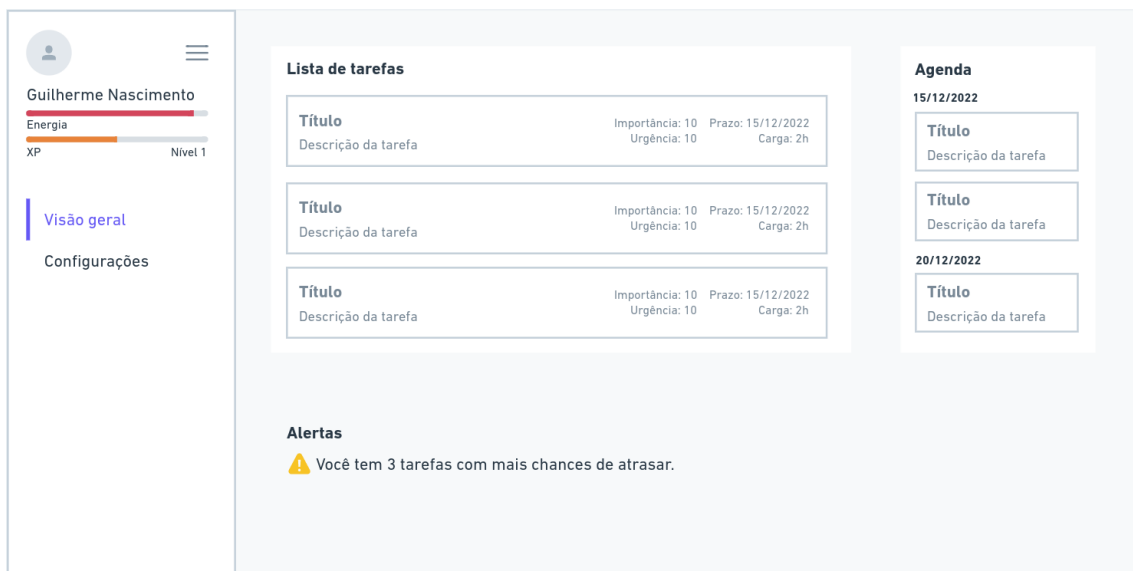
Figura 15: Wireframe - Tela de login.



Fonte: Elaborado pelo autor (2022).

A primeira interface é a mais comum dentro da internet, a tela de login para credenciar o usuário para entrar na plataforma. No nosso caso, login por meio de terceiros (como Google e Facebook) auxiliam a diminuir a barreira de entrada de novos usuários.

Figura 16: Wireframe - Tela inicial.



Fonte: Elaborado pelo autor (2022).

Após entrar na plataforma, o usuário tem uma visão geral das principais informações que ele inseriu. Dentre elas: a lista de tarefas priorizadas conforme o modelo parametrizado (em um primeiro momento poderia ser sem modelo, por

padrão); a lista de tarefas por meio de uma agenda; alertas para recomendações e avisos principais; um menu lateral para navegação; e os elementos de gamificação (descritos na subseção anterior) próximo à identificação do usuário.

Figura 17: Wireframe - Cadastro de tarefas.

The wireframe shows a modal window titled "Cadastro de tarefas" with a close button (X) in the top right corner. The form contains the following elements:

- A text input field labeled "Título".
- A larger text area labeled "Descrição e detalhamento da tarefa".
- Four input fields for task parameters: "Importância:" (10 pontos), "Urgência:" (10 pontos), "Prazo de entrega:" (15/12/2022), and "Carga de trabalho:" (2 horas).
- Two buttons at the bottom right: "Cancelar" and "Salvar".

Fonte: Elaborado pelo autor (2022).

Para o cadastro de novas tarefas pelo usuário, precisamos dos campos básicos de identificação (título e descrição) e os parâmetros para os modelos de priorização (importância, urgência, prazo de entrega e carga de trabalho).

Figura 18: Wireframe - Configuração da plataforma.

Configurar plataforma

Nome de usuário:

Modo: *Light* *Dark*

Salvar

Fonte: Elaborado pelo autor (2022).

Como falamos na subseção [5.2.1 Casos de uso](#), a ideia é que o usuário consiga alterar algumas características da plataforma. Essa parte realmente é mais estética, mas é interessante ter a possibilidade de alterar algumas características para a plataforma ser mais amigável ao usuário.

Figura 19: Wireframe - Comparação entre modelos.

Qual modelo se adequa melhor e parece mais efetivo?

Sem modelo
 Modelo básico
 Modelo avançado
 Modelo alternativo

Título 1 Descrição da tarefa	Título 2 Descrição da tarefa	Título 4 Descrição da tarefa	Título 2 Descrição da tarefa
Título 2 Descrição da tarefa	Título 1 Descrição da tarefa	Título 2 Descrição da tarefa	Título 4 Descrição da tarefa
Título 3 Descrição da tarefa	Título 3 Descrição da tarefa	Título 3 Descrição da tarefa	Título 3 Descrição da tarefa
Título 4 Descrição da tarefa	Título 4 Descrição da tarefa	Título 1 Descrição da tarefa	Título 1 Descrição da tarefa
Título 5 Descrição da tarefa	Título 6 Descrição da tarefa	Título 5 Descrição da tarefa	Título 6 Descrição da tarefa

Salvar

Fonte: Elaborado pelo autor (2022).

Para a tela comparando os modelos de priorização, o usuário pode analisar como ficaria cada listagem e qual delas parece realmente mais efetiva. Essa análise é mais qualitativa e envolve diversas variáveis mais específicas que nenhum modelo de priorização conseguiria estimar, por ser algo pontual para o caso do indivíduo e mudar conforme as suas preferências.

Figura 20: Wireframe - Comparação qualitativa entre modelos.

Como seria seu desempenho usando a priorização sem modelo?

Melhor Igual Pior

Como seria seu desempenho usando o modelo básico?

Melhor Igual Pior

Como seria seu desempenho usando modelo avançado?

Melhor Igual Pior

Como seria seu desempenho usando modelo alternativo?

Melhor Igual Pior

Salvar

Fonte: Elaborado pelo autor (2022).

Essa seria a tela com maior valor para este trabalho, onde o usuário poderia de fato avaliar o desempenho individual de cada modelo. Este é o quarto experimento, uma avaliação qualitativa dos modelos de priorização, realizado pelos próprios usuários da plataforma.

5.5 Aplicação do modelo de priorização de tarefas

De forma geral, o sistema é feito utilizando código que é semelhante a qualquer outro sistema web, com o cadastro, leitura, edição e remoção de tarefas. Algo que muitas ferramentas poderiam até gerar automaticamente, então podemos abstrair essa parte geral da análise do código.

A implementação do modelo de priorização de tarefas é a parte principal do código e por isso vamos focar na análise deste.

Como vimos, a lógica do modelo nada mais é que a aplicação de uma fórmula matemática utilizando os atributos das tarefas como variáveis. Portanto, a implementação do modelo nada mais é que um algoritmo para aplicar esta fórmula.

Relembrando que, para efeitos de comparação, utilizamos quatro modelos diferentes, variando a lógica e os atributos utilizados. Sendo assim, podemos analisar a implementação de cada modelo. Além disso, a implementação foi feita utilizando a linguagem Python com a biblioteca Pandas, mas descreveremos os pontos mais importantes do código.

Os trechos a seguir são responsáveis por definir o peso em cada tarefa. Fora disso, outras implementações ficaram responsáveis por ordenar e entregar essas tarefas ao usuário.

5.5.1 Sem modelo

O que estamos chamando de sem modelo nada mais é que a própria ordem definida pelo indivíduo para a execução das tarefas, então não alteramos a ordem original da execução das tarefas. Sendo assim, não seria necessário a implementação de um algoritmo de fato para uma priorização nesse estilo.

Porém, no sistema web e um pouco diferente dos experimentos realizados, consideramos um atributo da tarefa (como a urgência) para possibilitar que o usuário indique a ordem que pretende executar as tarefas. Essa diferença se dá pois os experimentos foram realizados com dados já agrupados e ordenados, e com tarefas já realizadas.

Dessa forma, o desenvolvimento é mais simples (menos complexo por não precisar gerenciar a indicação de ordem das tarefas pelo usuário) e mesmo assim cobre a necessidade de indicar uma priorização de tarefas de forma mais “simples”, sem um modelo mais avançado. Já que dessa forma, caso o usuário queira mudar a ordem da listagem, basta ele indicar um valor diferente para o campo indicado (por exemplo, aumentar ou diminuir a urgência de uma tarefa para alterar a posição dela na listagem).

Por exemplo, como podemos ver na imagem abaixo, atribuímos o valor da urgência da tarefa ao peso e todo o restante da lógica do sistema pode continuar igualmente aos demais modelos.

Figura 21: Código da implementação da priorização sem modelo, porém reordenada.

A screenshot of a code editor with a dark background and light text. The code is a Python function named 'sem_modelo' that takes 'self', 'df_old', and 'arg' as parameters. It copies 'df_old' into 'tasks', updates the 'peso' column with the values from 'arg', and returns 'tasks'. The code is numbered 1 to 4.

```
1 def sem_modelo(self, df_old, arg="Urgência"):
2     tasks = df_old.copy()
3     tasks['peso'] = tasks.loc[:, arg]
4     return tasks
```

Fonte: Elaborado pelo autor (2022).

Nesse caso, a implementação mais simples considera o valor da urgência como sendo o peso da tarefa. Então, a fórmula que expressaria essa priorização seria:

$$p = u$$

Onde:

p = Peso da tarefa;

u = Urgência;

5.5.2 Modelo básico

Para a implementação da priorização com esse modelo, alteramos o valor da urgência para a razão entre a carga e o prazo para entrega (ambos em horas). Fora isso, buscamos aplicar a fórmula que descrevemos inicialmente relacionando urgência e importância.

Figura 22: Código da implementação do modelo básico.

```

1 def basico(self, df_old):
2     df = df_old.copy()
3     tasks = df.loc[:, COLUMNS].copy()
4     tasks.loc[:, 'Urgência'] = tasks['Carga'] / tasks['Prazo Entrega']
5     tasks['peso'] = 2 * tasks.loc[:, 'Urgência'] + tasks.loc[:, 'Importância']
6     return tasks

```

Fonte: Elaborado pelo autor (2022).

Nessa implementação, o peso seria obtido relacionando a urgência ajustada com a importância da tarefa. Logo, a fórmula seria aproximadamente:

$$p = 2.(c / t) + i$$

Sendo:

p = Peso da tarefa;

i = Importância;

c = Carga de trabalho (em horas);

t = Prazo para entrega (em horas);

Estes valores foram obtidos conforme os testes realizados e obtiveram o melhor desempenho neste método.

5.5.3 Modelo avançado

Contudo, durante o desenvolvimento e os testes realizados, foi possível observar que colocando pesos diferentes nas variáveis os resultados poderiam ser melhores. Além disso, para as tarefas que teoricamente podem ser feitas antes do prazo de entrega (considerando a carga), adicionamos essa razão ao peso para influenciar na ordem da priorização final. Ou seja, tarefas com um “prazo mais apertado” seriam priorizadas, seguindo o que descrevemos anteriormente.

Com isso conseguimos testar, de forma mais manual, alguns pesos diferentes para cada variável e chegar no resultado final observado na implementação a seguir:

Figura 23: Código da implementação do modelo avançado.

```

1 def avancado(self, df_old):
2     df = df_old.copy()
3     tasks = df.loc[:, COLUMNS].copy()
4
5     # tarefas com carga_prazo > 1 não serão entregues antes do prazo
6     tasks['carga_prazo'] = tasks['Carga'] / tasks['Prazo Entrega'] * 15
7
8     # se dividir, vai mostrar quantos pontos de peso tem em cada unidade de prazo
9     tasks['peso'] = (2 * tasks['Urgência'] + tasks['Importância']) / tasks['carga_prazo']
10    # Se a carga_prazo for maior que 1, a tarefa está teoricamente atrasada
11    for i, row in tasks.iterrows():
12        if (row['carga_prazo'] <= 1):
13            tasks.loc[i, 'peso'] = row['peso'] + row['carga_prazo'] * 100
14    return tasks

```

Fonte: Elaborado pelo autor (2022).

Com isso, conseguimos obter resultados melhores (conforme observamos nas análises dos experimentos) mas ainda seguindo a lógica do modelo básico. Então, a fórmula que melhor expressaria este modelo na sua versão final seria:

$$p = (2.u + i) / (c / t . 15)$$

Sendo:

p = Peso da tarefa;

u = Urgência;

i = Importância;

c = Carga de trabalho (em horas);

t = Prazo para entrega (em horas);

Além disso, no final adicionamos a razão da carga e prazo ao peso final da tarefa:

$$\text{Se } (c / t . 15) \leq 1, \text{ então: } p = p + (c / t . 1500)$$

5.5.4 Modelo alternativo

Por fim, implementamos uma variação do modelo avançado que calcula o valor da urgência baseado na carga e no prazo e alterando um pouco os pesos das variáveis, como podemos observar:

Figura 24: Código da implementação do modelo alternativo.

```

1 def alternativo(self, df_old):
2     df = df_old.copy()
3     tasks = df.loc[:, COLUMNS].copy()
4
5     tasks['carga_prazo'] = (tasks['Carga'] / tasks['Prazo Entrega']) * 15
6
7     # Considerar mais urgente aquilo que possui menos tempo para ser concluído
8     tasks['Urgência'] = tasks['carga_prazo']
9
10    tasks['peso'] = (4 * tasks['Urgência'] + tasks['Importância']) / tasks['carga_prazo']
11    for i, row in tasks.iterrows():
12        if (row['carga_prazo'] <= 1):
13            tasks.loc[i, 'peso'] = row['peso'] + row['carga_prazo'] * 10
14    return tasks

```

Fonte: Elaborado pelo autor (2022).

Nesse modelo, a fórmula ficaria semelhante ao modelo avançado, como podemos ver abaixo:

$$p = (4 \cdot (c/t \cdot 15) + i) / (c/t \cdot 15)$$

Se $(c/t \cdot 15) \leq 1$, então: $p = p + (c/t \cdot 150)$

Sendo:

p = Peso da tarefa;

u = Urgência;

i = Importância;

c = Carga de trabalho (em horas);

t = Prazo para entrega (em horas);

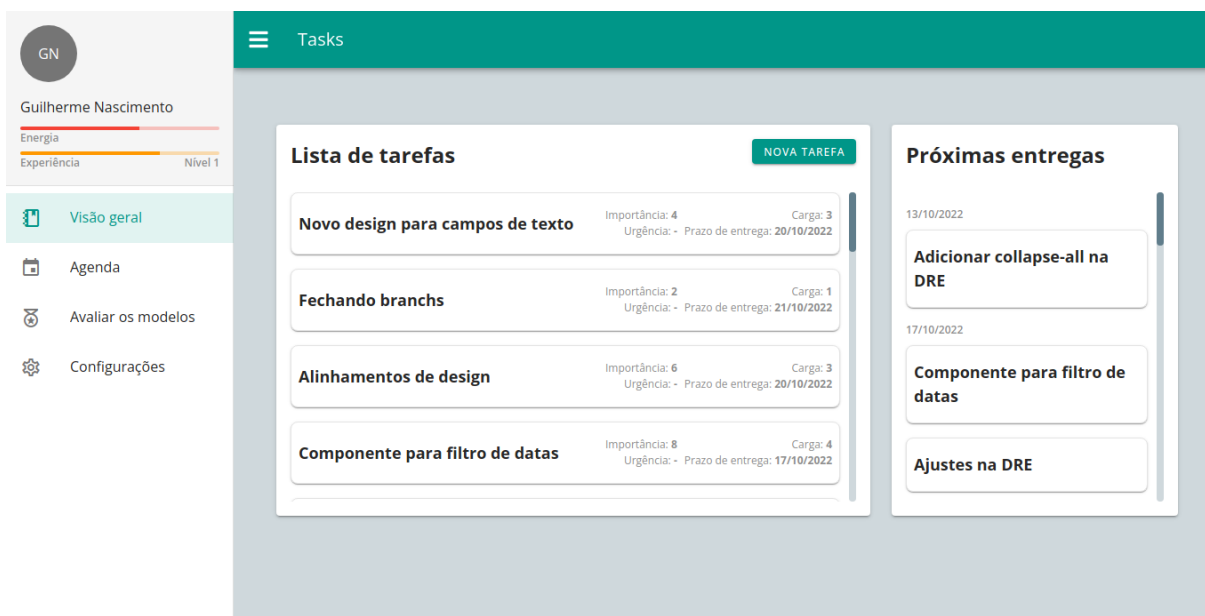
E como observamos anteriormente, esse modelo chegou nos melhores resultados dos experimentos. Durante a implementação utilizamos diferentes pesos para as variáveis e lógicas para o modelo, no entanto, esse foi o melhor resultado obtido.

5.6 Resultado

Conforme a descrição apresentada sobre o sistema web, podemos analisar as telas como resultado do desenvolvimento.

Primeiramente, temos a visão geral das tarefas do usuário. Podemos dizer que aqui ele pode encontrar as principais informações para o uso da plataforma. Dentre elas: os elementos de gamificação no canto superior esquerdo, a lista de tarefas já priorizada pelo modelo e uma apresentação de quais seriam as próximas entregas que ele teria (nesse caso, apenas agrupando as tarefas pelo prazo de entrega).

Figura 25: Tela de visão geral do sistema web.

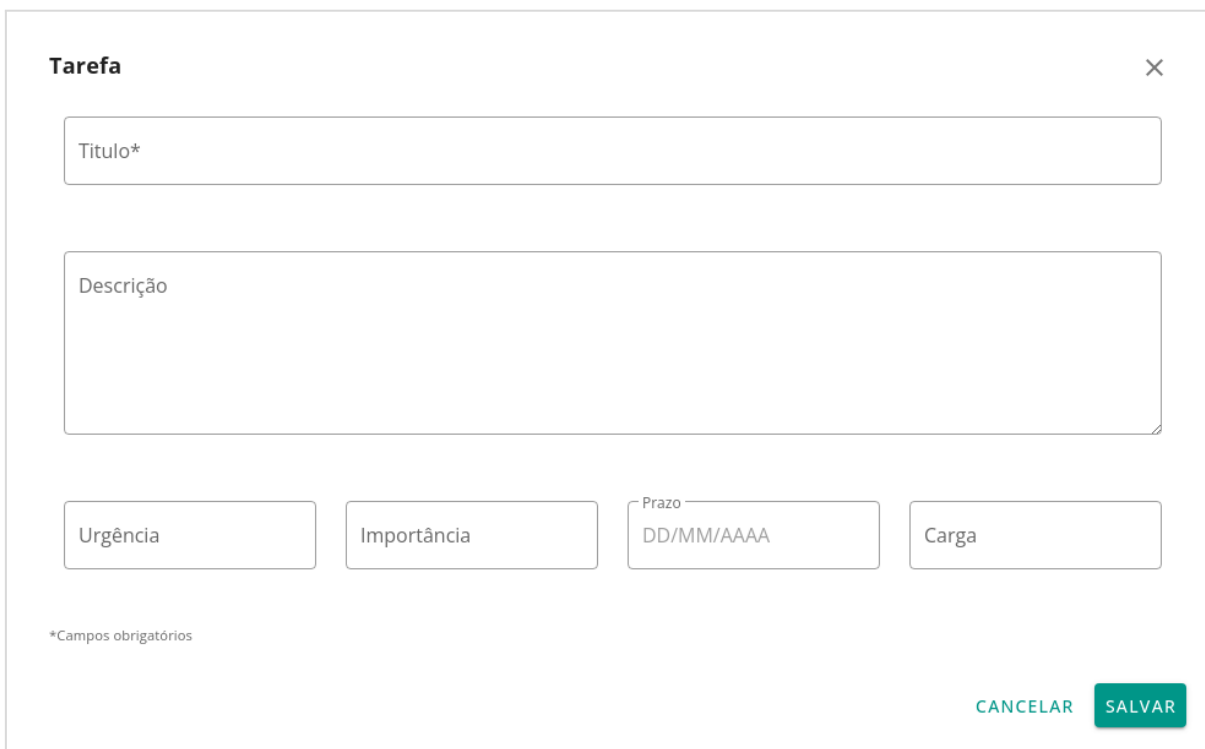


Fonte: Elaborado pelo autor (2022).

Aqui o usuário também teria acesso ao menu de navegação para as demais telas e ao cadastro de novas tarefas, que como podemos observar na figura abaixo, permite que além de preencher os atributos necessários para o modelo de

priorização, também permite que o usuário identifique a tarefa por meio de um título e adicione uma descrição para no futuro conseguir lembrar com mais detalhes o que precisaria ser feito em tal tarefa.

Figura 26: Tela para cadastro de tarefas.



A imagem mostra uma interface de usuário para o cadastro de tarefas. O formulário é intitulado "Tarefa" e possui um ícone de fechamento (X) no canto superior direito. Os campos de entrada são:

- Titulo***: Campo de texto obrigatório.
- Descrição**: Campo de texto maior para detalhes.
- Urgência**: Campo de seleção.
- Importância**: Campo de seleção.
- Prazo**: Campo de data com máscara DD/MM/AAAA.
- Carga**: Campo de seleção.

Na base do formulário, há o texto "*Campos obrigatórios" e dois botões: "CANCELAR" (desativado) e "SALVAR" (ativo).

Fonte: Elaborado pelo autor (2022).

Os atributos de urgência, importância e carga são subjetivos conforme o usuário acredita que deve preencher. A única restrição que precisa ser seguida aqui seria manter o mesmo padrão para o restante das tarefas. Ou seja, tarefas com um esforço semelhante precisam ter uma carga de trabalho semelhante, e assim por diante.

Além da configuração das tarefas, o usuário também tem a possibilidade de ajustar um pouco a plataforma para ficar da forma que lhe agrade mais, por meio da tela observada abaixo:

Figura 27: Tela para configurar a plataforma.

Fonte: Elaborado pelo autor (2022).

Para o experimento de avaliação qualitativa dos modelos, há a possibilidade do usuário utilizar a tela a seguir para informar qual modelo mais lhe agrada:

Figura 28: Tela para comparar os modelos.

Fonte: Elaborado pelo autor (2022).

E deve utilizar a segunda parte da tela para responder às questões sobre cada modelo, informando qual o ponto de vista de cada modelo apresentado, conforme seus próprios critérios pessoais.

Figura 29: Tela para avaliar os modelos.

Como seria o seu desempenho usando a priorização sem modelo?

Melhor Igual Pior

Como seria o seu desempenho usando o modelo básico?

Melhor Igual Pior

Como seria o seu desempenho usando o modelo avançado?

Melhor Igual Pior

Como seria o seu desempenho usando o modelo alternativo?

Melhor Igual Pior

SALVAR

Fonte: Elaborado pelo autor (2022).

Por fim, a plataforma também oferece uma visualização da lista de tarefas priorizadas em forma de uma agenda, ou seja, separando quais tarefas deveriam ser entregues em quais dias para seguir o que foi proposto pelo modelo. Nesse caso, a agenda não está apresentando por prazo de entrega, mas por ordem de execução.

Figura 30: Tela para visualizar as tarefas em forma de agenda de execução.

Agenda

As tarefas abaixo estão na ordem priorizada pelo modelo selecionado e separadas por dia em que devem ser entregues (não relacionado ao prazo de entrega).

28/11/2022

Novo design para campos de texto	Importância: 4 Urgência: -	Carga: 3 Prazo de entrega: 20/10/2022
Fechando branches	Importância: 2 Urgência: -	Carga: 1 Prazo de entrega: 21/10/2022
Alinhamentos de design	Importância: 6 Urgência: -	Carga: 3 Prazo de entrega: 20/10/2022

29/11/2022

Componente para filtro de datas	Importância: 8 Urgência: -	Carga: 4 Prazo de entrega: 17/10/2022
	Importância: 10	Carga: 4

Fonte: Elaborado pelo autor (2022).

Como utilizamos o sistema web somente rodando em servidor local, não priorizamos o desenvolvimento da tela de login e as interações entre diferentes usuários.

Porém, conseguimos observar que os principais pontos para a utilização do mesmo e para a avaliação qualitativa dos modelos estão presentes. Dessa forma, basta o indivíduo cadastrar algumas tarefas e conseguir analisar o desempenho de cada modelo de priorização.

6 CONSIDERAÇÕES FINAIS

Seguindo a proposta inicial do trabalho, foi apresentada a fundamentação teórica que serviu para o desenvolvimento da solução proposta. Definimos o significado dos termos tarefa, gerenciamento de tarefas e gamificação no escopo deste trabalho, para evitar dificuldades no entendimento por causa de palavras com mais de um significado.

Para o escopo que utilizamos, a tarefa seria uma ação a ser executada para alcançar algum objetivo que entrega algum valor, enquanto gerenciamento de tarefas seria a forma de organizar estas atividades, conforme uma lógica de priorização e de detalhamento que possibilite a execução das mesmas.

Além disso, vimos que a gamificação é trazer os elementos de jogos para fora do ambiente do jogo, no nosso caso, para o gerenciamento de tarefas, utilizando de características como metas claras e objetivas, sistema de recompensas, lógica de evolução, entre outros elementos.

Como era de se esperar para um problema tão comum atualmente, há diversas plataformas que auxiliam no gerenciamento de tarefas. Desde soluções mais simples para o gerenciamento de uma lista simples de tarefas (como Microsoft To Do e Google Tarefas), passando por soluções um pouco mais robustas por terem mais funcionalidades (como Google Agenda e Trello), até ferramentas mais complexas que possuem similaridades com a solução proposta (como o Habitica), apesar de não entregarem as funções que propomos.

Ao pesquisar por trabalhos semelhantes, encontramos algumas propostas interessantes para o gerenciamento de tarefas além de uma dissertação feita com o desenvolvimento de uma solução com gamificação para o gerenciamento de tarefas, que não foi concluída e não incluía um sistema de priorização nem de colaboração.

Entrando para a nossa proposta de modelo de priorização de tarefas, conseguimos descrever o conceito geral da ideia por trás deste. Demonstrando por meio de fórmulas matemáticas, por acreditar ser uma linguagem universal, foi possível detalhar como seria a lógica do modelo. A que melhor representaria a ideia seria:

$$p = (2.u + i) / (c / t)$$

Para conseguir avaliar o modelo proposto, executamos três experimentos para avaliar duas métricas diferentes em cada modelo em situações diferentes. Em cada um deles, conseguimos medir a eficiência e a melhora no desempenho em relação a uma priorização sem modelos.

Como resultado dos experimentos, observamos que o modelo mais completo que desenvolvemos (chamado aqui de modelo alternativo) apresentou resultados de melhora em todas as métricas levantadas. Em média (para 100.000 casos avaliados), conseguiu apresentar uma melhora de mais de 442% no tempo livre na execução das tarefas e mais de 15% de melhora na taxa de entrega das tarefas dentro do prazo. Isso mostra que utilizar esse modelo para priorizar tarefas realmente apresenta ganhos para o indivíduo.

Para conseguir fazer um experimento qualitativo, desenvolvemos um sistema web com o objetivo de facilitar o acesso ao resultado de cada modelo e permitir a avaliação destes. Após descrever em detalhes a estrutura da plataforma, demonstramos com código como seria a implementação de cada modelo citado durante o trabalho.

Por falta de tempo hábil, algumas funções não foram feitas e fica aberto para a extensão em trabalhos futuros. Como por exemplo: o agendamento das tarefas, a integração com o calendário já existente do usuário, entre outros.

6.1 Trabalhos futuros

O resultado final deste trabalho deixa aberto caminhos diferentes para possíveis trabalhos futuros, como por exemplo:

É possível seguir para mais estudos sobre o tema e aprimorar o modelo de priorização. Por exemplo, utilizar outras fórmulas para calcular o peso de cada tarefa (como a distância de Manhattan) ou adicionar mais variáveis para definir o peso de cada tarefa.

Também há espaço para melhorias no sistema web, como: a parte de gamificação que pode evoluir mais, incluindo mais elementos; a parte de colaboração entre indivíduos, com mais interações entre os usuários em tempo real; e integrações com sistemas externos (como a agenda do usuário).

Além disso, a parte de agendamento deveria considerar que não é tão adequado para o usuário executar uma mesma tarefa em dias diferentes, então faria

mais sentido buscar uma tarefa menor para completar o horário que ficou “disponível” e executar a outra tarefa de forma ininterrupta no dia seguinte.

Em relação aos experimentos, seria possível adicionar uma métrica para considerar a construção da agenda. Onde seria considerado o “status” obtido no resultado, como por exemplo: uma agenda ocupada (mais de 90% de horário preenchido), uma agenda normal (por volta de 80% de horário preenchido) ou uma agenda ociosa (menos de 50% do horário preenchido).

Para a parte de gamificação, seria interessante analisar a implementação de mais elementos, como por exemplo: desafios e missões específicas, como concluir um número determinado de tarefas em um período de tempo; ou até uma forma de permitir que o usuário aprimore o avatar dele com elementos visuais em troca de alguma bonificação que obteve ao concluir tarefas.

REFERÊNCIAS

- [1] RODRIGUES, Alexsandro Sutil. **O GERENCIAMENTO DO TEMPO APLICADO NA PRODUTIVIDADE, QUALIDADE DE VIDA E DESEMPENHO: ANÁLISE DE PUBLICAÇÕES NACIONAIS RESULTANTES DA BASE DE DADOS DO PORTAL DE PERIÓDICOS CAPES E GOOGLE ACADÊMICO**. Dourados, 2018. 59 p. Projeto de pesquisa (Bacharel em Administração) - Faculdade de Administração, Ciências Contábeis e Ciências Econômicas da Universidade Federal da Grande Dourados. Disponível em <https://repositorio.ufgd.edu.br/jspui/bitstream/prefix/2075/1/AlexsandroSutilRodrigues.pdf>.
- [2] BELLOTTI, Victoria; DALAL, Brinda; GOOD, Nathaniel; FLYNN, Peter; BOBROW, Daniel G.; DUCHENEAUT, Nicolas. **What a To-Do**: Studies of Task Management Towards the Design of a Personal Task List Manager. Vienna, 2004. 8 p. Paper. Disponível em <https://dl.acm.org/doi/pdf/10.1145/985692.985785>.
- [3] I. Ohmukai; H. Takeda; M. Miki. **A proposal of the person-centered approach for personal task management**. Orlando, 2003. Symposium on Applications and the Internet. Disponível em <https://ieeexplore.ieee.org/abstract/document/1183055>.
- [4] NAVARRO, Gabrielle. **Gamificação**: a transformação do conceito do termo jogo no contexto da pós-modernidade. São Paulo, 2013. 26 p. Trabalho de conclusão de curso (Mídia, Informação e Cultura) - CELACC/ECA. Disponível em <http://paineira.usp.br/celacc/sites/default/files/media/tcc/578-1589-1-PB.pdf>.
- [5] COSTA, Amanda C. S.; MARCHIORI, Patricia Z. **Gamificação, elementos de jogos e estratégia**: uma matriz de referência. Ribeirão Preto, 2015. 22 p. Disponível em <https://www.revistas.usp.br/incid/article/view/89912/103928>.
- [6] TANDEL, Sayali S.; JAMADAR, Abhishek. **Impact of Progressive Web Apps on Web App Development**. Maharashtra, 2018. 6 p. Disponível em https://www.researchgate.net/profile/Sayali-Tandel-2/publication/330834334_Impact

_of_Progressive_Web_Apps_on_Web_App_Development/links/5c5605d3a6fdccd6b5dde018/Impact-of-Progressive-Web-Apps-on-Web-App-Development.pdf>.

[7] CARDIERI, Giulia de A.; ZAINA, Luciana M. **Analyzing User Experience in Mobile Web, Native and Progressive Web Applications: A User and HCI Specialist Perspectives.** Sorocaba, 2018. 11 p. Universidade Federal de São Carlos. Disponível em <<https://dl.acm.org/doi/pdf/10.1145/3274192.3274201>>.

[8] SILVA, Hortência T. M. **A IMPORTÂNCIA DE METODOLOGIAS DE PRIORIZAÇÃO NO PROCESSO DE GESTÃO DE PORTFÓLIO DE PROJETOS:** um estudo de caso. São Luís, 2020. 26 p. Trabalho de conclusão de curso (Bacharel em Administração) - Universidade Federal do Maranhão (UFMA). Disponível em <<https://monografias.ufma.br/jspui/bitstream/123456789/4550/1/HORTENCIA-SILVA.pdf>>.

[9] GONÇALVES CALDEIRA BRANT LOSEKANN, R.; CARDOSO MOURÃO, H. **DESAFIOS DO TELETRABALHO NA PANDEMIA COVID-19: QUANDO O HOME VIRA OFFICE.** Caderno de Administração, v. 28, p. 71-75, 5 jun. 2020. Disponível em <<https://periodicos.uem.br/ojs/index.php/CadAdm/article/view/53637/751375150139>>

[10] OHMUKAI, Ikki; TAKEDA, Hideaki. **Social scheduler: a proposal of collaborative personal task management.** Tokyo, 2003. 5 p. Artigo. Disponível em <<http://www-kasm.nii.ac.jp/papers/takeda/03/ohmukai03wi.pdf>>.

[11] BARROSO, Jordão França. **WORK RIFT - RPG de Realidade Alternativa para gerenciamento de tarefas.** Goiânia, 2016. 173 p. Dissertação. Disponível em <<https://repositorio.bc.ufg.br/tede/handle/tede/6799>>.

[12] SUNG, Christine; OSTER, Troy; MILLMAN, David; FASY, Brittany T.; BURGUESS, Mac; BRATTERUD, Hannah; **The Sung Diagram: Revitalizing the Eisenhower Matrix.** Livro Diagrammatic Representation and Inference. p. 498-502, Agosto 2020. Disponível em

<https://www.researchgate.net/publication/343686873_The_Sung_Diagram_Revitalizing_the_Eisenhower_Matrix>.

[13] KACZOROWSKI, Miłosz. **The Best Programming Languages to Use In Your Backend In 2022.** Ideamotive, 2022. Disponível em <<https://www.ideamotive.co/blog/best-programming-languages-to-use-in-your-backend>>.

[14] Pandas. Biblioteca para análise de dados e ferramenta de manipulação. Disponível em <<https://pandas.pydata.org/>>.

[15] SSA Group. **Factors to consider when choosing a frontend framework for web application.** 2021. Disponível em <<https://www.ssa.group/blog/choosing-a-frontend-framework/>>.

[16] GOUVEIA, Luís Borges; RANITO, João. **SISTEMAS DE INFORMAÇÃO DE APOIO À GESTÃO.** Porto, 2004. 96 p. Manual. Seção 1.2. Disponível em <https://bdigital.ufp.pt/bitstream/10284/264/1/Manual_VII.pdf>.

[17] RAGAN, Sean M. **Fun Theory Award winner rewards safe drivers with lottery tickets.** Makezine, 2010. Disponível em <<https://makezine.com/article/science/energy/fun-theory-award-winner-rewards-saf/>>. Acesso em 26 nov 2022.

[18] TERRA, Mateus. **Como gastar menos tempo ao priorizar tarefas em escritórios de advocacia.** 2022. Artigo. Disponível em <<https://www.aurum.com.br/blog/priorizar-tarefas-escritorio-de-advocacia/>>.

APÊNDICES

APÊNDICE A - Conjunto de tarefas reais - semana 1

Título	Importância	Urgência	Data entrega	Carga
Planejamento com equipe	6	2	17/10/2022	1
Componente para filtro de datas	8	6	17/10/2022	4
Adicionar collapse-all na DRE	10	10	13/10/2022	1
Verificando homologação	10	8	17/10/2022	1
Ajustes na DRE	6	6	17/10/2022	2
Ajustes nos dados financeiros	5	5	18/10/2022	2
Preparando versão para release	10	10	18/10/2022	1
Liberando versão	10	10	18/10/2022	1
Campos com % na DRE	8	6	19/10/2022	2
Expandir/minimizar tudo na DRE	6	4	19/10/2022	1
Barra de representatividade	8	4	19/10/2022	2
Modal de boas vindas	10	10	20/10/2022	2
Integrar filtros de datas	10	10	20/10/2022	1
Novo design para campos de texto	4	2	20/10/2022	3
Ajustes na gestão de carteiras	10	2	20/10/2022	1
Alinhamentos de design	6	4	20/10/2022	3
Ajustes no partnership profissional	4	1	20/10/2022	1
Mostrar eventos de come-cotas	10	10	20/10/2022	1
Leitura cumbuca	10	10	21/10/2022	4
Rota para evolução do saldo	8	4	21/10/2022	1
Fechando branches	2	4	21/10/2022	1

APÊNDICE B - Conjunto de tarefas reais - semana 2

Título	Importância	Urgência	Data entrega	Carga
Atualizar titulo da tabela DFC	1	10	10/10/2022	1
Atualizar email de cobrança internet	2	3	10/10/2022	1
Novo componente para filtro de datas	5	2	10/10/2022	2
Integrar tela do profissional	8	9	10/10/2022	4
Ajustar visão geral financeira	10	10	10/10/2022	2
Cabeçalho fixo para DRE	5	4	11/10/2022	1
Ajustes de design na DRE	4	4	10/10/2022	1
Integrar DRE do profissional	8	10	11/10/2022	2
Integrar pagina de produto	6	10	11/10/2022	2
Integrar pagina de filial	6	10	11/10/2022	2
Integrar dre filial e produto	5	9	11/10/2022	1
Separar tarefa para tarefas do dia	1	1	12/10/2022	1
Atualizar modal de release	5	9	10/10/2022	2
Ordem dos menus navbar	1	1	11/10/2022	1
Ordem colunas resultado profissional	1	1	11/10/2022	1
Ajustar texto para a representatividade	1	1	11/10/2022	1
Priorizar tarefas do sprint	5	5	10/10/2022	1
Modal de novidades	4	8	11/10/2022	2
Erro ao salvar contrato	9	9	11/10/2022	1
Mudar ordem do navbar	4	6	12/10/2022	1
Analisar nova versão para produção	6	10	12/10/2022	2
Documentar hygen	4	3	12/10/2022	1
Avaliar PR	2	2	12/10/2022	1
Definição de cor em whitelabel	4	1	12/10/2022	2
Ajustes gerais no módulo financeiro	10	4	13/10/2022	6
Feedback onboarding	10	2	14/10/2022	1
Entrevista consultoria	2	2	14/10/2022	1
Componente para filtro de datas	6	2	17/10/2022	4
Ler Arquitetura limpa	4	8	21/10/2022	8

APÊNDICE C - Conjunto de tarefas com dados aleatórios

Nesse conjunto de dados, o valor para o prazo de entrega já é considerando quantas horas estão disponíveis do início do conjunto até o prazo de entrega e o valor para urgência é uma razão entre a carga e o prazo de entrega.

Titulo	Importância	Urgência	Carga	Prazo Entrega
0	8	1,5625	1	64
1	0	4,166666667	2	48
2	2	1,5625	1	64
3	1	5	2	40
4	6	7,142857143	4	56
5	5	1,388888889	1	72
6	8	7,5	3	40
7	8	10	4	40
8	7	8,333333333	4	48
9	8	4,166666667	2	48
10	4	2,5	1	40
11	3	2,083333333	1	48
12	0	4,6875	3	64
13	1	2,777777778	2	72
14	6	1,785714286	1	56
15	3	2,083333333	1	48
16	1	1,785714286	1	56
17	9	5	2	40
18	7	7,5	3	40
19	9	8,333333333	4	48
20	3	3,571428571	2	56
21	0	7,142857143	4	56
22	8	1,388888889	1	72
23	0	1,5625	1	64
24	1	5,357142857	3	56
25	1	10	4	40
26	0	5	2	40
27	3	10	4	40
28	0	10	4	40

29	8	5,555555556	4	72
30	6	3,125	2	64
31	9	10	4	40
32	5	2,083333333	1	48
33	3	2,5	1	40
34	0	2,083333333	1	48
35	0	2,777777778	2	72
36	4	2,083333333	1	48
37	0	6,25	3	48
38	5	5,555555556	4	72
39	0	2,5	1	40

APÊNDICE D - Códigos-fonte

Para simplificar este relatório, seguem os links para os repositórios com o código-fonte utilizado.

Para o servidor: <https://github.com/guiliznas/tasks-backend-py>

Para a plataforma web (frontend): <https://github.com/guiliznas/tasks-frontend>

APÊNDICE E - ARTIGO

Sistema para gerenciamento de tarefas com gamificação

Guilherme de Liz do Nascimento

Universidade Federal de Santa Catarina

guilherme.liz@grad.ufsc.br

***Abstract.** As the complexity of society advances, the demand for people's productivity also increases, but the available time to complete these tasks is still the same. To assist in the completion of various activities within a certain deadline, this article presents a task management method that prioritizes them according to their characteristics (such as urgency and importance), based on the Eisenhower matrix, but adapted to gamification techniques, and fits them into a schedule with available time slots. To test the efficiency of the model, in addition to quantitative experiments, we used a web platform with some gamification elements.*

***Resumo.** Com o avanço da complexidade da sociedade a demanda por produtividade das pessoas também aumenta, porém o tempo disponível para o cumprimento dessas tarefas ainda é o mesmo. Para auxiliar no cumprimento de diversas atividades em determinado prazo, este artigo apresenta um método de gerenciamento de tarefas que as priorize conforme suas características (como urgência e importância), baseado na matriz de Eisenhower, porém adaptado à técnicas de gamificação, e encaixe-as em uma agenda com espaços de tempo disponíveis. Para testar a eficiência do modelo, além de experimentos quantitativos, utilizamos uma plataforma web com alguns elementos de gamificação.*

1. Introdução

No nosso dia-a-dia executamos diversas tarefas, tanto no âmbito profissional quanto pessoal e familiar, e com o avanço das tecnologias em geral podemos perceber essas tarefas mudando e aumentando em quantidade e complexidade.

Isso nos leva a ter uma necessidade de organizar da melhor forma possível o recurso escasso e precioso que temos: o tempo, para obter uma boa produtividade. Segundo os autores (Ohmukai; Takeda; Miki, 2003), a complexidade em gerir os recursos pessoais aumenta juntamente com o avanço da complexidade da nossa sociedade.

Como a execução de tarefas pode ser algo um pouco menos atrativo para as pessoas, a aplicação utilizará algumas técnicas de gamificação. Navarro (2013) explica o objetivo da gamificação, aumentar o engajamento dos usuários ao utilizar a aplicação:

Percebe-se que o objetivo principal da gamificação é criar envolvimento entre o indivíduo e determinada situação, aumentando o interesse, o engajamento e a eficiência na realização de uma tarefa específica, buscando mudar o comportamento desse indivíduo.

Como o objetivo seria justamente aumentar a produtividade dos indivíduos, conseguir ter um maior engajamento está em linha com o objetivo principal.

1.1 Objetivo

Em resumo, o objetivo consiste em propor um método, representado por uma expressão matemática, para gerenciamento de tarefas baseado na Matriz de Eisenhower adaptado à gamificação e testá-lo mediante sua implementação em um sistema web.

2. Conceitos básicos

2.1 Tarefa

Apesar de possuir diferentes definições em diferentes escopos, para este trabalho usaremos uma definição onde a tarefa seria um “trabalho, ação ou conjunto de ações a serem realizadas para o cumprimento de determinada missão, objetivo ou meta, num determinado prazo” (dicionário CIMM). Para simplificar, vamos considerar que a tarefa é uma ação que possui um objetivo e uma entrega de valor. Além disso, ela possui alguns atributos que a caracterizam, como um título, uma descrição própria, um prazo de entrega, uma carga de trabalho demandada para sua execução, entre outros.

2.2 Matriz de Eisenhower

A Matriz de Eisenhower procura auxiliar na execução de tarefas, por definição, o “seu objetivo consiste em evitar que tarefas se acumulem e se sobreponham umas às outras, minando o fluxo dos processos” (SILVA, 2020).

A matriz proposta por Eisenhower seria algo como um plano cartesiano, com um eixo sendo a importância da tarefa e o outro sendo a urgência dela. No final, seria possível definir o destino da tarefa conforme o que encontramos na figura abaixo.

	URGENTE	NÃO URGENTE
IMPORTANTE	I QUADRANTE "Faça agora"	I QUADRANTE "Agende"
NÃO IMPORTANTE	I QUADRANTE "Delegue"	I QUADRANTE "Elimine"

Figura 1. Matriz de Eisenhower (TERRA, 2022)

Dessa forma, tarefas sem urgência e sem importância deveriam ser eliminadas, enquanto tarefas com muita importância e muita urgência deveriam ser feitas agora, por exemplo.

2.3 Gamificação

Apesar dos jogos existirem desde os tempos antigos, o termo gamificação é algo relativamente recente, ainda mais na área de desenvolvimento. Segundo Navarro (2013), a palavra ainda não possui um conceito definitivo e exato, porém pode ser explicada como “a aplicação de elementos, mecanismos, dinâmicas e técnicas de jogos no contexto fora do jogo”. Onde inclui características como a definição de desafios e metas, conquistas como pontos e troféus, entre outros elementos que encontramos constantemente nos jogos.

3. Trabalhos relacionados

3.1 Social scheduler: a proposal of collaborative personal task management

O autor apresenta uma estrutura para apoiar um sistema de tomada de decisões, baseada em três partes:

Visualizador, que fica responsável pela apresentação das tarefas, seja por meio de uma lista simples, de uma agenda em forma de calendário, ou qualquer outro formato desejado.

Otimizador, que seria a parte responsável por priorizar as tarefas, indicando a melhor ordem de execução das tarefas.

Recomendador, ou Agendador, propõe uma forma de executar as tarefas dentro de uma agenda, aplicando a priorização feita pelo otimizador em uma ordem real de execução considerando o tempo disponível em cada dia, por exemplo.

Além disso, é apresentado o modelo colaborativo chamado de *alliance* (aliança) que consiste em compartilhar as informações entre as pessoas envolvidas e fomentar a colaboração dos indivíduos nas atividades.

3.2 The Sung Diagram: Revitalizing the Eisenhower Matrix

Os autores identificam o que acreditam ser uma falha na matriz de Eisenhower (a falta de consideração de ajustes que podem ser feitos, como tarefas que podem ser terceirizadas) e apresentam uma extensão para ela.

Então, adicionam uma variável a mais para ser considerada que chamam de “ajuste” (*fit*) que indica se aquela tarefa precisa ser realizada pelo indivíduo ou se pode ser delegada à outro. Com isso, temos o diagrama de Sung apresentado na figura 2.

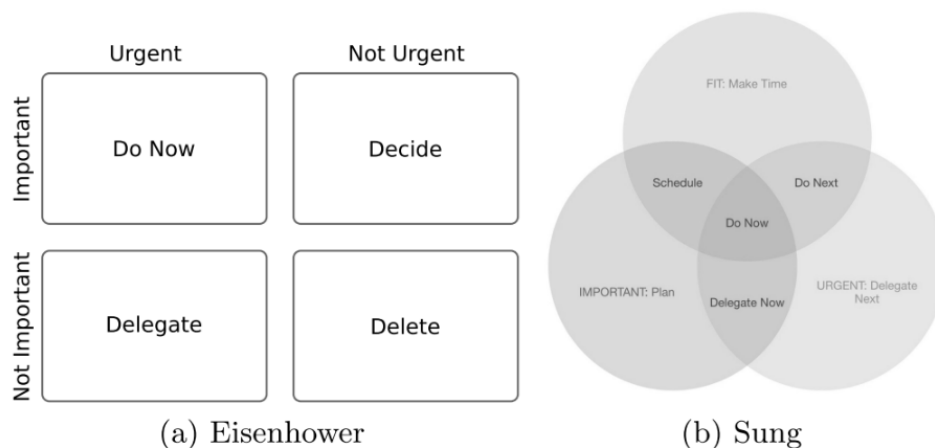


Figura 2. Comparar Matriz de Eisenhower e Diagrama de Sung (SUNG, 2020)

3.3 WORK RIFT - RPG de Realidade Alternativa para gerenciamento de tarefas

Para o trabalho proposto por Barroso, foram aplicadas técnicas de gamificação para a execução de atividades em sala de aula, apresentando como isso funcionaria em um sistema. Porém, ele deixa de fora a colaboração entre usuários (por ser algo focado em estudantes, não deveria existir a colaboração mesmo) e não utiliza um modelo para priorização de tarefas.

3.4 Considerações

Cada trabalho apresentado apresenta uma característica que isoladamente já cumpre um papel específico, porém em sua maioria não unem os diferentes pontos.

Na nossa proposta, uniremos as principais características de diferentes trabalhos para atingir o objetivo proposto. Apenas o artigo de Sung (2020) apresenta um ponto não incluído no modelo proposto por ser uma outra alternativa, já que seria uma extensão da matriz de Eisenhower, porém que poderia ser adicionado em trabalhos futuros.

4. Modelo

Podemos resumir o modelo de gerenciamento de tarefas em uma expressão matemática, representando a relação entre os atributos de uma tarefa (as variáveis da expressão) como um valor que pode ser utilizado na priorização destas. Ou seja, calcular o que podemos chamar de “peso” de cada tarefa, de acordo com o modelo, para então otimizar o conjunto de tarefas em uma ordem mais eficiente de execução. Com isso, ordenar a lista de tarefas por estes pesos e definir o que seria uma ordem mais eficiente de execução destas.

Este modelo pode ser implementado por meio de um algoritmo que nos permite automatizar o cálculo dos pesos e a priorização das tarefas, sem necessidade de verificar uma a uma.

Durante todo o desenvolvimento deste trabalho, foi feita uma implementação incremental do modelo. Sendo assim, partimos de uma versão mais “básica” e aprimoramos para tentar obter o melhor resultado possível e resultando em diferentes versões do modelo, como vamos observar a seguir.

Com isso, para uma primeira versão de um modelo mais básico, consideramos somente a urgência e a importância para calcular o peso, da seguinte forma:

$$p = 2.u + i$$

Onde:

p = Peso da tarefa (pontuação utilizada para a priorização);

u = Urgência;

i = Importância;

Nesse modelo, a urgência fica definida com uma prioridade maior que a importância das tarefas. Foi feito dessa forma considerando que quando duas tarefas possuem valores semelhantes para estes atributos, aquela com uma urgência maior deveria ser realizada antes. Ou seja, uma tarefa com uma urgência maior tem um peso maior do que uma tarefa com uma urgência menor e um peso semelhante. Consideramos isso para que tarefas mais urgentes

tenham um peso maior, justamente pelo fato de precisar ser entregue antes (considerando o próprio conceito de urgência).

Para uma análise mais avançada, é necessário considerar alguns atributos a mais, vamos utilizar carga de trabalho e prazo necessário para entrega (que começa a nos levar também para o envolvimento de uma agenda completa), ambos em horas.

Com isso, é possível considerar duas novas variáveis no modelo e aprimorá-lo para uma versão um pouco mais avançada:

$$p = (2 \cdot u + i) / (c / t)$$

Sendo:

c = Carga de trabalho (em horas);

t = Prazo para entrega (em horas);

Nessa expressão, verificamos a proporção do tempo necessário para executar a tarefa e o tempo disponível até o prazo de entrega. Depois disso, verificamos como ficaria a razão do peso da tarefa em relação a essa proporção do tempo necessário para executá-la.

A relação entre o tempo disponível e o tempo necessário para executar a tarefa (uma estimativa) deve ser menor que 1 para que a tarefa seja considerada possível de ser concluída, portanto, se esse valor for maior que 1 podemos dizer que ela já está atrasada ou que pelo menos a chance dela ser entregue antes do prazo é menor que as demais.

Com isso, tarefas que estão com um prazo “mais apertado” possuem um peso maior, priorizando elas para serem executadas antes (no caso do surgimento de algum imprevisto, já que possui mais tempo para algo dar errado, aumentaria o espaço para ajustes).

Desta forma, já temos a ideia macro do modelo de priorização, aplicando o conceito da matriz de Eisenhower na priorização das tarefas em relação a ordem de execução destas. Além disso, podemos considerar pesos diferentes em cada variável utilizada na expressão matemática para tentar melhorar os resultados. Ou até considerar algumas condições a mais, como por exemplo, para tarefas com valores semelhantes utilizar a razão da carga e do prazo como um “desempate” para definir quais seriam priorizadas.

4.1 Gamificação

O peso e demais atributos utilizados no modelo servem como entrada para os elementos de gamificação que podem ser construídos, como um fator de influência em medidores de desempenho do indivíduo.

Por exemplo, podemos considerar que o indivíduo possui uma “energia” (que muitos jogos também chamam de “vida”) que aumenta ou diminui de acordo com a carga de trabalho de cada tarefa que é entregue antes ou depois do prazo, considerando uma paridade de um para um. Ou seja, para cada hora na carga de trabalho seria descontado um ponto de energia.

Ele também possui uma “experiência” (que muitos jogos utilizam para indicar o “nível” do usuário), que é alterada conforme a entrega das tarefas e de acordo com o peso destas, também considerando uma paridade de um para um. Ou seja, para cada um ponto no peso da tarefa concluída, seria adicionado um ponto de

4.2 Forma de implementação

Para automatizar o cálculo de uma expressão matemática (para não ser necessário calcular manualmente para cada situação) utilizamos um algoritmo desenvolvido com o objetivo de aplicar o modelo.

Esse algoritmo recebe um conjunto de tarefas, realiza o cálculo do peso destas e resulta em uma lista ordenada conforme a avaliação obtida. Algo extremamente simples e que nos remete ao conceito de sistema de informação com uma entrada para um processamento que gera uma saída, segundo as funções de um sistema de informação conforme Gouveia e Ranito (2004).

Com essa lista ordenada e uma parametrização simples indicando a quantidade de horas disponíveis por dia para a execução das tarefas (padrão de 8 horas diárias), é possível realizar a distribuição das tarefas nos dias, formando uma agenda para o indivíduo.

4.3 Métricas de avaliação

Em um caso real de execução de um conjunto de tarefas, um indivíduo pode identificar diversas variáveis que influenciam na priorização das tarefas ou na indicação se uma priorização seria melhor que outra.

No entanto, para o escopo dos experimentos que vamos precisar nos limitar à variáveis quantitativas e que possam ser calculadas baseadas nos atributos existentes das tarefas.

Sendo assim, para os experimentos realizados, vamos considerar duas métricas principais: Tempo livre (o total de horas entre a conclusão teórica da tarefa e o prazo de entrega) e Entrega no prazo (percentual de tarefas que foram concluídas antes do prazo de entrega).

Com essas métricas, já é possível avaliar o resultado do modelo de priorização em um conjunto de tarefas e realizar alguns experimentos de forma automática para analisar o desempenho dos modelos e medir a variação média das métricas. Outra medida que podemos

observar, ao testar em mais de um conjunto de tarefas, é a quantidade de ocorrências em que o modelo se sobressai aos demais em ambas as métricas quantitativas.

Além disso, é possível utilizar uma métrica qualitativa para a avaliação de um modelo, como por exemplo “o indivíduo acredita que teria um desempenho pior, igual ou melhor utilizando essa priorização?”. Para isso, seria necessário o envolvimento de mais indivíduos e o desenvolvimento de um sistema para que possam visualizar o resultado e avaliá-los. Sendo assim, o indivíduo indicaria qual seria o modelo com melhor desempenho considerando as variáveis que ele mesmo define, de forma específica para cada caso.

4.4 Experimentos

Para a realização dos experimentos, calculamos as métricas para quatro formas de priorização das tarefas para avaliar o desempenho do método proposto, sendo elas:

- **Sem modelo:** Considerando apenas a urgência das tarefas;
- **Modelo básico:** Considerando o modelo com a relação apenas entre urgência e importância;
- **Modelo avançado:** Considerando mais atributos, como a carga de trabalho e o prazo de entrega;
- **Modelo alternativo:** Durante o desenvolvimento dos experimentos, conseguimos estender o modelo avançado para uma versão aprimorada, colocando pesos diferentes e calculado a urgência com base na carga de trabalho e no prazo de entrega.

No primeiro experimento, usamos agendas reais levantadas pelo próprio autor durante períodos de uma semana. No primeiro conjunto de tarefas, todos os modelos obtiveram uma entrega dentro do prazo de 100% das tarefas, porém, houve uma melhora de aproximadamente 6% do modelo alternativo em relação a uma execução sem modelo de priorização:

Tabela 1. Resultado do primeiro experimento, métrica de tempo livre

Modelo	Tempo livre (h)
Sem modelo	1253h
Básico	1269h
Avançado	1279h
Alternativo	1330h

No segundo conjunto de tarefas avaliado, o modelo proposto para priorização chegou a entregar quase o dobro de tarefas dentro do prazo de entrega:

Tabela 4. Resultado do primeiro experimento, métrica de entrega no prazo

Modelo	Entrega no prazo (%)
Sem modelo	27,6%
Básico	24,1%
Avançado	51,7%
Alternativo	55,2%

Dessa forma, já podemos observar que o modelo proposto apresenta uma melhora no desempenho na execução das tarefas. Porém, para evitar um pouco a parcialidade, executamos um experimento semelhante com um conjunto de tarefas geradas de forma aleatória. Ou seja, os atributos utilizados na fórmula para o cálculo do peso de cada tarefa foram definidos aleatoriamente em tempo de execução, para evitar a influência do próprio autor na observação das tarefas, e a ordem de execução das tarefas foi considerada a mesma ordem de criação delas para a priorização sem modelo, para os demais foi considerada a ordem obtida na priorização dos pesos calculados das tarefas conforme cada modelo.

Com isso, foi possível avaliar o desempenho de cada modelo novamente, mas agora considerando as tarefas aleatórias, onde foi possível observar um aumento de aproximadamente 93% na métrica de tempo livre, apesar de não haver uma variação relevante na taxa de entrega dentro do prazo:

Tabela 5. Resultado do segundo experimento.

Modelo	Entrega no prazo (%)	Tempo livre (h)
Sem modelo	62,5%	247h
Básico	52,5%	43h
Avançado	67,5%	374h
Alternativo	62,5%	477h

Por fim, para avaliar a melhora média no desempenho, foram realizados os mesmos testes porém em uma escala maior com 100.000 conjuntos de tarefas aleatórias (geradas de forma aleatória igual ao experimento anterior). Neste caso, o objetivo seria avaliar a média de

variação entre a métrica obtida em um modelo para a mesma métrica em uma execução de priorização, por exemplo: qual a variação na taxa de entrega de tarefas dentro do prazo para o modelo alternativo?

Em relação ao tempo livre, foram obtidos os seguintes resultados, com o modelo alternativo apresentando o melhor desempenho em 97% dos casos:

Tabela 6. Resultado do terceiro experimento, medidas da métrica de tempo livre

Modelo	Casos com maior tempo livre	Varição em relação ao padrão (sem modelo)
Sem modelo	2.250	N/A
Básico	7	55,8%
Avançado	28	17,6%
Alternativo	97.769	442,8%

Já em relação a taxa de entrega de tarefas dentro do prazo, o modelo alternativo se saiu melhor em 73% dos casos, com uma melhora média de 15%:

Tabela 7. Resultado do terceiro experimento, medidas da métrica de taxa de entrega no prazo

Modelo	Casos com melhor taxa de entrega no prazo	Varição em relação ao padrão (sem modelo)
Sem modelo	12.150	N/A
Básico	7.119	0,7%
Avançado	26.956	4,7%
Alternativo	73.626	15,06%

De forma geral, foi observado que o modelo alternativo apresentou o melhor resultado em ambas as variáveis em 72% dos casos:

Tabela 8. Resultado do terceiro experimento, resumo das métricas

Modelo	Casos com melhor resultado em ambas as métricas
Sem modelo	1.518

Básico	5
Avançado	27
Alternativo	72.741

5. Desenvolvimento da plataforma web

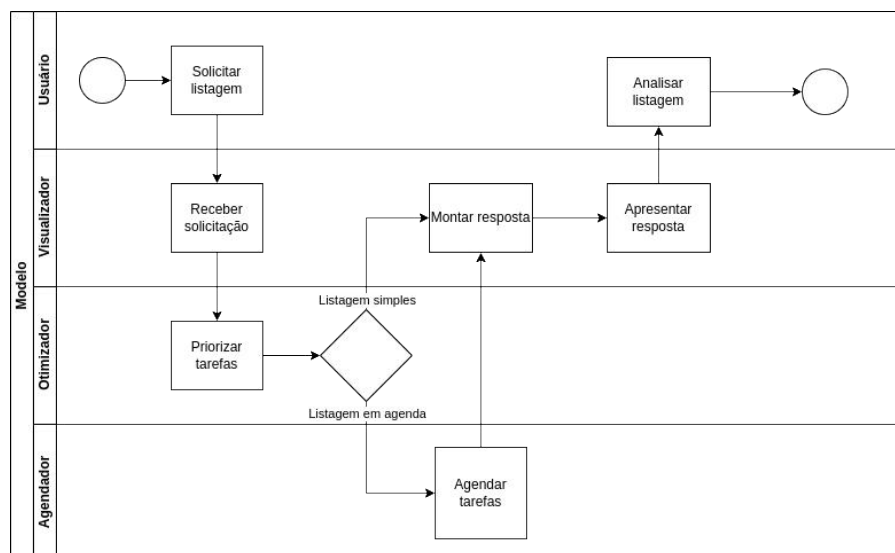
Para realizar um experimento qualitativo, é necessário disponibilizar o acesso ao modelo pelos indivíduos. Neste caso, a opção escolhida foi uma plataforma web. Para o servidor, foi utilizada a linguagem Python, por apresentar ferramentas adequadas para a situação (como Flask e Pandas). E para a parte da web, foi utilizado o *framework* VueJS, por ser mais simples (atendendo aos requisitos) e ter um desempenho bom.

5.1 Estrutura

Para atender os requisitos necessários para o experimento qualitativo, foi utilizado uma estrutura simples de cliente-servidor. Fora isso, seguimos a ideia apresentada na subseção 3.1, de utilizar um Visualizador, Otimizador e Agendador como parte principal da priorização das tarefas.

Sendo assim, de forma resumida, podemos apresentar as principais funções do sistema no diagrama de atividades abaixo:

Figura 3. Diagrama de atividades



Conforme o diagrama, a atividade começa com uma solicitação do usuário enviada ao Visualizador, que por sua vez, envia a requisição para o Otimizador, juntamente com os parâmetros de priorização selecionados pelo usuário (o modelo desejado). Com esses dados, o otimizador busca as tarefas do usuário e prioriza elas conforme o modelo desejado. Caso a

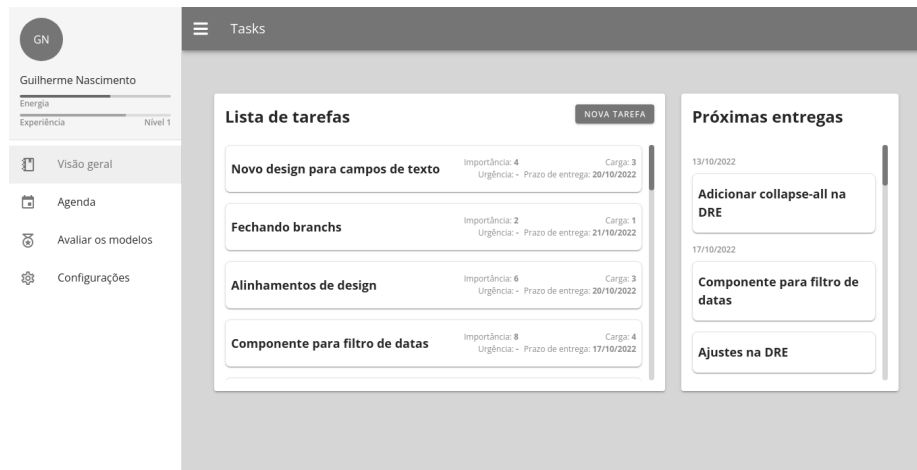
solicitação seja para apresentar as tarefas em um formato de agenda, o otimizador envia para o Agendador distribuir essas tarefas.

Com a listagem pronta, seja ela em agenda ou não, o visualizador recebe novamente e prepara a resposta adequada, apresentando-a ao usuário que enfim pode analisar a resposta obtida.

5.2 Resultado

Após o desenvolvimento, conseguimos apresentar para o usuário de forma geral as tarefas priorizadas conforme o modelo proposto:

Figura 4. Tela de visão geral do sistema web



Sendo assim, já seria possível acompanhar as tarefas que precisam ser executadas. Além disso, apresentamos também uma sugestão em forma de agenda, ou seja, quais tarefas deveriam ser feitas em cada dia:

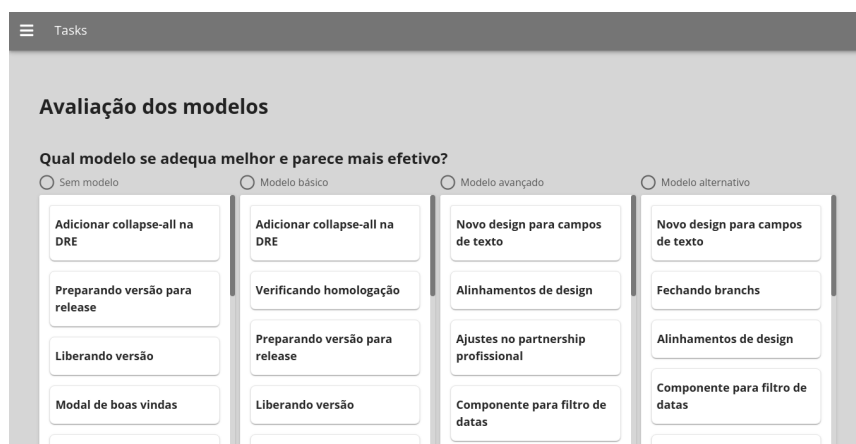
Figura 5. Tela para visualizar as tarefas em forma de agenda de execução



Com esta listagem, já seria possível uma avaliação qualitativa do modelo, mas também disponibilizamos uma forma de comparar a priorização de cada modelo para avaliar

aquele que apresenta o melhor desempenho. Dessa forma, cada usuário terá suas próprias variáveis que considera na hora de priorizar tarefas mas consegue avaliar a priorização do modelo proposto e verificar se está adequado a sua própria situação:

Figura 6. Tela para comparar os modelos



6. Considerações finais

Seguindo a proposta inicial do trabalho, foi apresentada a fundamentação teórica que serviu para o desenvolvimento da solução proposta. Definimos o significado dos termos tarefa, matriz de Eisenhower e gamificação no escopo deste trabalho, para evitar dificuldades no entendimento por causa de palavras com mais de um significado.

Entrando para a nossa proposta de modelo de priorização de tarefas, conseguimos descrever o conceito geral da ideia por trás deste. Demonstrando por meio de fórmulas matemáticas, por acreditar ser uma linguagem universal, foi possível detalhar como seria a lógica do modelo. A que melhor representaria a ideia seria:

$$p = (2 \cdot u + i) / (c / t)$$

Para conseguir avaliar o modelo proposto, executamos três experimentos para avaliar duas métricas diferentes em cada modelo em situações diferentes. Em cada um deles, conseguimos medir a eficiência e a melhora no desempenho em relação a uma priorização sem modelos.

Como resultado dos experimentos, observamos que o modelo mais completo que desenvolvemos (chamado aqui de modelo alternativo) apresentou resultados de melhora em todas as métricas levantadas. Em média (para 100.000 casos avaliados), conseguiu apresentar uma melhora de mais de 442% no tempo livre na execução das tarefas e mais de 15% de

melhora na taxa de entrega das tarefas dentro do prazo. Isso mostra que utilizar esse modelo para priorizar tarefas realmente apresenta ganhos para o indivíduo.

Para conseguir fazer um experimento qualitativo, desenvolvemos um sistema web com o objetivo de facilitar o acesso ao resultado de cada modelo e permitir a avaliação destes. Após descrever em detalhes a estrutura da plataforma, demonstramos com código como seria a implementação de cada modelo citado durante o trabalho.

Por falta de tempo hábil, algumas funções não foram feitas e fica aberto para a extensão em trabalhos futuros. Como por exemplo: o agendamento das tarefas, a integração com o calendário já existente do usuário, entre outros.

6.1 Trabalhos futuros

O resultado final deste trabalho deixa aberto caminhos diferentes para possíveis trabalhos futuros, como por exemplo:

A realização dos experimentos qualitativos com a plataforma web, para obter mais uma métrica de avaliação para o modelo proposto.

É possível seguir para mais estudos sobre o tema e aprimorar o modelo de priorização. Por exemplo, utilizar outras fórmulas para calcular o peso de cada tarefa (como a distância de Manhattan) ou adicionar mais variáveis para definir o peso de cada tarefa, como a variável proposta no diagrama Sung de ajuste das tarefas.

Também há espaço para melhorias no sistema web, como: a parte de gamificação que pode evoluir mais, incluindo mais elementos; a parte de colaboração entre indivíduos, com mais interações entre os usuários em tempo real; e integrações com sistemas externos (como a agenda do usuário).

Referências

I. Ohmukai; H. Takeda; M. Miki (2003). **A proposal of the person-centered approach for personal task management**. Symposium on Applications and the Internet. Disponível em <<https://ieeexplore.ieee.org/abstract/document/1183055>>.

NAVARRO, Gabrielle. **Gamificação**: a transformação do conceito do termo jogo no contexto da pós-modernidade (2013). 26 p. Trabalho de conclusão de curso (Mídia, Informação e Cultura) - CELACC/ECA. Disponível em <<http://paineira.usp.br/celacc/sites/default/files/media/tcc/578-1589-1-PB.pdf>>.

SILVA, Hortência T. M (2020). **A IMPORTÂNCIA DE METODOLOGIAS DE PRIORIZAÇÃO NO PROCESSO DE GESTÃO DE PORTFÓLIO DE PROJETOS:**

um estudo de caso. 26 p. Trabalho de conclusão de curso (Bacharel em Administração) - Universidade Federal do Maranhão (UFMA). Disponível em <<https://monografias.ufma.br/jspui/bitstream/123456789/4550/1/HORTENCIA-SILVA.pdf>>.

TERRA, Mateus (2022). **Como gastar menos tempo ao priorizar tarefas em escritórios de advocacia.** Disponível em <<https://www.aurum.com.br/blog/priorizar-tarefas-escritorio-de-advocacia/>>.

OHMUKAI, Ikki; TAKEDA, Hideaki (2003). **Social scheduler: a proposal of collaborative personal task management.** 5 p. Artigo. Disponível em <<http://www-kasm.nii.ac.jp/papers/takeda/03/ohmukai03wi.pdf>>.

SUNG, Christine; OSTER, Troy; MILLMAN, David; FASY, Brittany T.; BURGUESS, Mac; BRATTERUD, Hannah (2020). **The Sung Diagram: Revitalizing the Eisenhower Matrix.** Livro Diagrammatic Representation and Inference. p. 498-502. Disponível em <https://www.researchgate.net/publication/343686873_The_Sung_Diagram_Revitalizing_the_Eisenhower_Matrix>.

BARROSO, Jordão França (2016). **WORK RIFT - RPG de Realidade Alternativa para gerenciamento de tarefas.** 173 p. Dissertação. Disponível em <<https://repositorio.bc.ufg.br/tede/handle/tede/6799>>.

GOUVEIA, Luís Borges; RANITO, João (2004). **SISTEMAS DE INFORMAÇÃO DE APOIO À GESTÃO.** 96 p. Manual. Seção 1.2. Disponível em <https://bdigital.ufp.pt/bitstream/10284/264/1/Manual_VII.pdf>.