

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA
PARAÍBA**

**UM SISTEMA COLABORATIVO DE GESTÃO E AUXÍLIO A
EQUIPES CRIATIVAS NO CONTEXTO DE DESIGN E
DESENVOLVIMENTO DE JOGOS INDIE**

THIAGO YURE VIEIRA DE ALMEIDA

Cajazeiras - Paraíba, Agosto de 2023

THIAGO YURE VIEIRA DE ALMEIDA

**UM SISTEMA COLABORATIVO DE GESTÃO E AUXÍLIO A EQUIPES
CRIATIVAS NO CONTEXTO DE DESIGN E DESENVOLVIMENTO DE
JOGOS INDIE**

Trabalho apresentado junto ao programa de **Graduação em Análise e Desenvolvimento de Sistemas** do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, como requisito parcial à obtenção da nota na **disciplina de TCC II**.

Orientador:

Prof. Me. Diogo Dantas Moreira.

Cajazeiras - Paraíba, Agosto de 2023

IFPB / Campus Cajazeiras
Coordenação de Biblioteca
Biblioteca Prof. Ribamar da Silva
Catalogação na fonte: Cícero Luciano Félix CRB-15/750

A447s Almeida, Thiago Yure Vieira de.
Um sistema colaborativo de gestão e auxílio a equipes criativas no contexto de design e desenvolvimento de jogos indie / Thiago Yure Vieira de Almeida.– 2023.

44f. : il.

Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) - Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Cajazeiras, 2023.

Orientador(a): Prof. Me. Diogo Dantas Moreira.

1. Desenvolvimento de sistemas. 2. Desenvolvimento de jogos online. 3. Gestão de equipes. 4. sistema de colaboração. I. Instituto Federal de Educação, Ciência e Tecnologia da Paraíba. II. Título.



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

THIAGO YURE VIEIRA DE ALMEIDA

**UM SISTEMA COLABORATIVO DE GESTÃO E AUXÍLIO A EQUIPES CRIATIVAS NO
CONTEXTO DE DESIGN E DESENVOLVIMENTO DE JOGOS INDIE**

Trabalho de Conclusão de Curso apresentado junto ao
Curso Superior de Tecnologia em Análise e
Desenvolvimento de Sistemas do Instituto Federal de
Educação, Ciência e Tecnologia da Paraíba - Campus
Cajazeiras, como requisito à obtenção do título de
Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador

Prof. MSc. Diogo Dantas Moreira.

Aprovada em: **29 de Agosto de 2023.**

Prof. MSc. Diogo Dantas Moreira - Orientador

Prof. Dr. Fabio Gomes de Andrade - Avaliador

IFPB - Campus Cajazeiras

Prof. MSc. Paulo Ewerton Gomes Fragoso - Avaliador

IFPB - Campus Cajazeiras

Documento assinado eletronicamente por:

- **Diogo Dantas Moreira**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 01/09/2023 07:50:31.
- **Paulo Ewerton Gomes Fragoso**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 01/09/2023 09:08:45.
- **Fabio Gomes de Andrade**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 04/09/2023 08:22:18.

Este documento foi emitido pelo SUAP em 30/08/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 468477

Verificador: 17df5d57c2

Código de Autenticação:



Rua José Antônio da Silva, 300, Jardim Oásis, CAJAZEIRAS / PB, CEP 58.900-000

<http://ifpb.edu.br> - (83) 3532-4100

Dedico este trabalho a minha mãe Lúcia e ao meu irmão Adonay, que sempre me apoiaram durante todo o desenvolvimento deste trabalho e durante todo o curso. Também dedico ao meu amigo e orientador Diogo Dantas, por sua dedicação, competência e paciência.

AGRADECIMENTOS

A Deus pela vida, sabedoria, força e proteção. Ao professor Diogo Dantas Moreira pela orientação, dedicação e amizade. A minha família por ser a base da minha vida e por sempre me apoiar em tudo. Aos meus amigos por me darem suporte e carinho em todos os momentos. A todos os professores do curso que sempre me apoiaram e me ajudaram a chegar ao meu objetivo. A todos os funcionários do IFPB campus Cajazeiras pelos serviços prestados e pela hospitalidade com que fui tratado. Aos colegas que tive durante os anos que estive na graduação.

*“It’s not wether you were right or wrong, but
how much faith you were willing to have that
decides the future.”*

Hideo Kojima, Metal Gear Solid 2: Sons of
Liberty

RESUMO

O desenvolvimento de jogos indie vem se popularizando cada vez mais conforme a sociedade avança, tanto dentro da própria indústria de jogos, quanto na de entretenimento no geral. Devido a esse aumento de visibilidade e importância, lacunas no processo de design e gerenciamento ficaram mais evidentes, principalmente no que diz respeito a gerência das equipes que compõem o projeto. Considerando esse cenário, é possível perceber que um sistema colaborativo focado na gestão de equipes em projetos criativos, como é o caso dos jogos indie, se torna uma adição quase indispensável. O trabalho em questão busca propor o desenvolvimento de um sistema colaborativo que atue na organização e apoio a projetos de jogos indie, com enfoque principal no gerenciamento e manutenção da equipe encarregada de desenvolver a ideia e o design do mesmo. Utilizando como metodologia o Design Science Research, o trabalho espera conseguir desenvolver a pesquisa de forma iterativa e incremental, de modo que proporcione resultados analíticos e concretos que possam de alguma forma tentar melhorar o contexto apresentado.

Palavras-chave: Design de jogos *indie*, Gestão de equipes criativas, Sistemas colaborativos.

ABSTRACT

The development of indie games has become increasingly popular as society advances, both within the gaming industry itself and in the entertainment industry in general. Due to this increase in visibility and importance, gaps in the design and management process became more evident, especially with regard to the management of the teams that make up the project. Considering this scenario, it is possible to see that a collaborative system focused on managing teams in creative projects, as is the case with indie games, becomes an almost indispensable addition. The work in question seeks to propose the development of a collaborative system that acts in the organization and support of indie game projects, with a main focus on managing and maintaining the team in charge of developing the idea and design of the same. Using Design Science Research as a methodology, the work hopes to be able to develop the research in an iterative and incremental way, so that it provides analytical and concrete results that can somehow try to improve the context presented.

Keywords: *Indie* game design, Creative team management, Collaborative systems.

LISTA DE FIGURAS

Figura 2.1 – Ciclo de Design	10
Figura 5.1 – Visão geral dos requisitos do sistema	22
Figura 5.2 – Arquitetura do sistema Windie	24
Figura 5.3 – Tela do quadro de tarefas	26
Figura 5.4 – Tela de perfil do time	27
Figura 5.5 – Avaliação e recomendação de usuários	28
Figura 5.6 – Tela de demonstração do algoritmo de recomendação	29
Figura 5.7 – Fluxo de processamento da imagem no sistema	30
Figura 5.8 – Fluxo de funcionamento da biblioteca Socket.IO	31
Figura A.1 – Formulário de envio de imagem no cadastro de usuário	35
Figura B.1 – Tela de pesquisa e convite de colaboradores	36
Figura C.1 – Tela da sala de reuniões	37

SUMÁRIO

1	INTRODUÇÃO	6
1.1	Problemática	7
1.2	Objetivos	9
1.3	Contribuições do trabalho	9
1.4	Organização do trabalho	9
2	METODOLOGIA	10
3	REFERENCIAL TEÓRICO	12
3.1	Histórico do desenvolvimento de jogos	12
3.2	Design de Jogos	14
3.3	Mercado de jogos digitais	14
3.4	Desenvolvimento de jogos Indie	15
3.5	Sistemas Colaborativos	16
3.6	Gestão de equipes criativas	16
3.7	WebSocket	17
3.8	Sistemas de Recomendação	17
4	FERRAMENTAS SIMILARES	19
4.1	Trello	19
4.2	KanbanFlow	20
4.3	Notion	20
4.4	Slack	20
5	IMPLEMENTAÇÃO	22
5.1	Apresentação do sistema Windie	22
5.2	Requisitos	22
5.3	Arquitetura	23

5.4	Requisitos implementados	25
5.4.1	RF1 Gerenciamento de fases e tarefas	25
5.4.2	RF2 Gerenciamento de equipes criativas	25
5.4.3	RF3, RF4 e RF5 Avaliação e recomendação de usuários e projetos	26
5.4.4	RF6. Integração com ferramentas existentes	27
5.4.5	RF7. Gerenciamento de imagem do perfil de usuários e projetos	28
5.4.6	RF8. Sala de reuniões	29
5.4.7	Outros requisitos	30
5.5	Comparação com outras ferramentas	30
5.6	Dificuldades encontradas	30
6	CONSIDERAÇÕES FINAIS	32
	REFERÊNCIAS	33
	APÊNDICE A – IMAGEM DO FORMULÁRIO DE ENVIO DE IMAGEM NO CADASTRO DE USUÁRIO	35
	APÊNDICE B – IMAGEM DA TELA DE PESQUISA E CONVITE DE COLABORADORES	36
	APÊNDICE C – IMAGEM DA TELA DA SALA DE REUNIÕES	37

1 INTRODUÇÃO

Sendo um dos filhos do desenvolvimento de software que mais cresce comercialmente e popularmente nos últimos tempos, o desenvolvimento de jogos digitais carrega consigo genes ruins herdados de seu pai. E como toda nova camada que é adicionada a algo já existente, ele possui suas próprias peculiaridades, tanto positivas quanto negativas.

O Odyssey, criado pela Magnavox em 1972, foi considerado o primeiro console para jogos pessoal da história (LOGUIDICE; BARTON, 2014). Embora de maneira ainda limitada, começava a ganhar os holofotes de forma discreta um novo ramo de mercado e possibilidades para qual a programação poderia ser utilizada, entretenimento. Mais precisamente na forma de jogos, inicialmente simples, mas que com o tempo e os avanços tecnológicos se tornariam cada vez mais complexos e diversos.

A importância da indústria vem crescendo substancialmente e tomando o espaço de mídias mais tradicionais no mercado. Segundo a WePC (2020) os jogos são de longe a indústria de entretenimento mais lucrativa, valendo US\$145,7 bilhões em 2019, em comparação com US\$ 42,5 bilhões em receitas de bilheteria e US\$ 20,2 bilhões em Música.

É de se esperar que com tamanha importância econômica, o desenvolvimento de jogos e suas características únicas também ganhasse uma atenção merecida. No entanto, como já foi citado, os problemas que foram herdados e os novos criados não deixam de incomodar e de existir.

Se focarmos nas grandes empresas que desenvolvem esse tipo de atividade, talvez elas consigam contornar a maioria desses problemas e ter uma forma de desenvolver mais robusta. Mas mesmo assim nem sempre o resultado é satisfatório ou preenche as expectativas iniciais. Um exemplo recente disso foi o lançamento do aguardado *Cyberpunk 2077*¹.

Apesar das expectativas e especulações sobre as inovações visuais e de jogabilidade, o jogo pecou em diversos pontos, durante o desenvolvimento e até após o jogo sair para o público. A empresa responsável, a CD Projekt, adiou cada vez mais o lançamento para no fim ceder a pressão e entregar algo que possuía claramente problemas de design, programação, excesso de *bugs* (denominação dada aos erros que impedem o programa de se comportar como o esperado), tanto visuais, como de programação.

Fica claro que mesmo nesse cenário mais robusto, montar uma equipe capaz de

¹ **Cyberpunk 2077**. Disponível em: <<https://www.cyberpunk.net/us/pt-br/>>

dar vida a um projeto criativo simples ou mesmo ambicioso pode ser um desafio complexo. Pois ter um time que ao mesmo tempo que agregue, resolva e supere os obstáculos depende de muitos fatores. Podem ser tanto gerenciais, relacionados ao design do jogo, marketing, a divisão das equipes, aos recursos investidos, quanto podem estar ligados ao material humano. Nesse quesito podemos citar a experiência dos envolvidos, a motivação dos mesmos, o entendimento da ideia, o gerenciamento dessas pessoas, etc.

Isso se torna mais evidente quando nos concentramos no desenvolvimento de jogos independentes (chamaremos de “jogos *indie*”, termo amplamente utilizado na indústria, desse ponto em diante), que na sua maioria possuem mínimo ou nenhum recurso, pouca experiência dos colaboradores, menos disponibilidade de tempo, etc. Essa realidade acaba gerando situações em que, na maioria dos casos, poucas pessoas acumulam muitas funções e por consequência, despesas e estresses.

Pois diferente das grandes empresas que desenvolvem jogos, no desenvolvimento indie esse processo está muito mais ligado a um desejo e engajamento pessoal daqueles que decidem ir por esse caminho. Sendo assim é fácil imaginar que o processo não seja de fácil organização e muito menos exista uma garantia de sucesso no fim de tudo.

1.1 PROBLEMÁTICA

Segundo a [VGInsights \(2020\)](#) em levantamento feito sobre a receita arrecadada por jogos *indie* dentro da Steam, uma das maiores plataformas de streaming de jogos da atualidade,² mostra que mais da metade dos jogos *indie* não consegue atingir uma arrecadação maior do \$4.000,00. Segundo eles isso indica que esses jogos não conseguem sequer cobrir os custos e o tempo investido para desenvolvê-los.

Essa é uma realidade muito comum nesse universo mais independente. Falta de recursos (humanos, intelectuais e físicos), pouca experiência, incerteza, ambiente instável sempre são pontos recorrentes e que ajudam nesse resultado.

[Ćurčić \(2018\)](#), que já trabalhou como programador, game designer, gerente de projeto e produtor em vários jogos indie durante sua carreira, escreveu um artigo para o site Marble IT onde descreveu alguns problemas vivenciados por ele e provavelmente pela maioria dos envolvidos nesse tipo de projeto.

Terminar o projeto é um deles. Ele comenta que a maioria dos projetos sofre por não conseguir encontrar um fim, na maioria das vezes pelo simples motivo de que várias ideias são adicionadas em tempo de desenvolvimento, aumentando o número de bugs,

² **Steam**. Disponível em: <<https://store.steampowered.com/>>

novas coisas a serem desenvolvidas, etc. Segundo ele isso se deve a falta de planejamento prévio, falta de clareza da ideia e falta de análise minuciosa de cada nova ideia.

O segundo problema citado por ele seria cobrir todas as etapas do desenvolvimento de jogos em si. Ele se refere aqui a um problema supracitado, que muitas pessoas acabam acumulando funções e na maior parte das vezes essas não são especialistas naquilo ou não sabem nada sobre. Nesse caso tem que desprender tempo aprendendo novos assuntos, softwares e formas de fazer algo. E mesmo que isso possa significar mais controle dos idealizadores sobre todas as áreas do projeto, isso pode significar acabar usando de forma ineficiente seu tempo e atenção.

E por fim ele cita o problema de publicação e *marketing* do jogo em si. Quase sempre ligado à falta de experiência, conhecimento, tempo para empregar na análise de estatística e mercado e uma forma de destacar o jogo no meio da multidão. Nesses casos, segundo ele, as soluções mais empregadas são ou delegar essa tarefa a algum companhia especializada ou utilizar-se de blogs de desenvolvimento, fóruns, etc.

É sabido que existem casos de sucesso comercial e de popularidade, como é o caso de *Cuphead*³, *Undertale*⁴, *Limbo*⁵, *Braid*, entre outros. No entanto do outro lado existem jogos pouco mencionados ou de pouca relevância comercial. Um exemplo de fracasso comercial e criativo foi o jogo *Mighty No.9*, que foi desenvolvido com estratégia de financiamento coletivo e idealizado por Keiji Inafune (ilustrador e co-designer do personagem Megaman).

Sem contar os inúmeros projetos que nem tiveram a oportunidade de ver a luz do dia e tentar sua chance no mercado, fruto dos problemas que cercam essa indústria. E mesmo os exemplos de sucesso com certeza passaram por problemas dessa natureza, que provavelmente atrapalharam o seu andamento.

Considerando esse cenário, é possível enxergar que apesar de ser um mercado promissor e com vários casos de sucesso, a indústria de jogos indie possui uma realidade de desenvolvimento com alguns problemas claros. Embora alguns deles derivam da filosofia de desenvolvimento empregada nesses projetos, todos eles podem ser mitigados ou resolvidos. Logo, existe uma carência na indústria dos jogos, principalmente de jogos independentes, por uma solução que amenize ou solucione esses problemas específicos.

³ **Cuphead**. Disponível em: <<http://www.cupheadgame.com/>>

⁴ **Undertale**. Disponível em: <<https://undertale.com/>>

⁵ **Limbo**. Disponível em: <<https://playdead.com/games/limbo/>>

1.2 OBJETIVOS

Portanto, pensando em tudo que foi exposto até aqui, esse trabalho tem como objetivo geral desenvolver uma plataforma de apoio ao processo de desenvolvimento de jogos indie. Tanto na parte de criação e gerenciamento da equipe, quanto no auxílio a todas as etapas do desenvolvimento. Para atingi-lo, alguns objetivos específicos estão elencados abaixo:

- Produzir um levantamento das ferramentas utilizadas por desenvolvedores de jogos;
- Desenvolver um comparativo com as ferramentas encontradas para identificar pontos fortes e fracos;
- Propor uma ferramenta baseada nos requisitos e pontos de melhorias a partir do comparativo;
- Realizar uma avaliação da ferramenta no desenvolvimento de um jogo exemplo.

1.3 CONTRIBUIÇÕES DO TRABALHO

A contribuição deste trabalho para o contexto apresentado está na criação de uma plataforma de gestão e colaboração que facilitará a interação e organização entre indivíduos engajados no desenvolvimento de projetos de jogos indie. Com isso espera-se que os problemas e fracassos relacionados a tais fatores sejam minimizados ou até mesmo sanados proporcionando ao contexto supracitado um ambiente mais ameno e organizado para que os problemas existentes não impeçam a evolução estável desse ramo do mercado.

1.4 ORGANIZAÇÃO DO TRABALHO

O capítulo 2 trata-se de uma explicação sobre a metodologia utilizada no trabalho. Já no capítulo 3 expomos o referencial teórico utilizado para o desenvolvimento e andamento do trabalho em questão. A seguir temos o capítulo 4, que se empenha em exemplificar algumas ferramentas semelhantes a que está sendo proposta. No capítulo 5, os detalhes e pormenores da implementação da proposta. Por último no capítulo 6, as considerações finais.

2 METODOLOGIA

Considerando o caráter científico e a intenção do presente trabalho de produzir uma solução prática para um problema real em um contexto específico, se mostrou necessário o uso de uma estrutura de pesquisa que seguisse essa mesma premissa. Portanto, para tal, foi escolhida pelos autores a metodologia Design Science, como proposto em [Wieringa \(2014\)](#).

Pautada na produção de artefatos e ou recomendações como saída para o seu desenvolvimento, ela se fundamenta em aproximar o mundo e o conhecimento real, dentro do contexto, da sua realidade desejável. A disposição das tarefas a serem desenvolvidas em cinco seções bem definidas garante que seja seguida a estrutura lógica da ação racional ([BAX, 2017](#)).

Ainda assim, essas cinco atividades podem ser alocadas em dois ciclos. O primeiro, chamado de Ciclo de Design, engloba as fases de investigação do problema, design do tratamento e validação do tratamento. Nesse primeiro momento a metodologia é aplicada sobre um modelo do contexto real ([WIERINGA, 2014](#)). Em cada uma das fases é possível utilizar métodos de pesquisa distintos (revisão bibliográfica, análises, aplicação de questionários, entrevistas, etc), dependendo dos objetivos de cada ciclo.

Figura 2.1 – Ciclo de Design



Fonte: Adaptado de ([WIERINGA, 2014](#))

Sendo a investigação do problema o ponto de partida, ela procura identificar, definir e averiguar o problema como um todo, desde seus interessados, até suas causas e influências, se mantendo no âmbito do conhecimento teórico sobre o tema. Dessa forma, as próximas etapas possuem uma visão clara de quais serão seus pontos focais de ação. Essa parte da metodologia foi desenvolvida no início do projeto através de pesquisas em sites, artigos acadêmicos, documentários, livros e da experiência pessoal do autor e exposta neste documento nos capítulos da introdução e da problemática.

Ao passar para a etapa de design do tratamento, a formulação da proposta que será utilizada para tentar tornar a realidade dos interessados mais próxima do desejado, é modelada. As teorias e suspeitas levantadas com base no que foi exposto na fase anterior, definem o plano de ação a ser considerado para a sequência do ciclo. O plano definido neste trabalho para prosseguir na metodologia consistiu no desenvolvimento da ferramenta proposta, cujos detalhes são explicados no capítulo 5.

Feito isso, a validação do tratamento se encarrega de questionar sobre os efeitos que a proposta tem sobre um modelo do contexto real. Desde sua eficácia, comparação com outros artefatos sobre o mesmo contexto, aplicabilidade em um ambiente contextual diferente do modelado, até a sua fidelidade com a intenção inicial do desenvolvimento. Esta etapa foi a última a ser englobada neste trabalho, considerando que esses questionamentos são contemplados conforme se avalia o sucesso ou não deste trabalho naquilo que ele se propõe.

Em seguida, o desenvolvimento do trabalho pode passar para o segundo ciclo, esse chamado de Ciclo de Engenharia. Nele estão incluídas as atividades de implementação do tratamento e avaliação do tratamento, pois nessa etapa é preciso transpor a proposta e validá-la dentro do contexto no mundo real ([WIERINGA, 2014](#)).

A implementação do tratamento corresponde, tão somente, à inclusão e garantia de que o tratamento em questão será executado dentro do contexto no mundo real. A partir disso entra-se na fase de avaliação do tratamento. Nessa etapa, será analisado e avaliado como as condições levantadas em nível de conhecimento na etapa de investigação do problema, foram influenciadas ou modificadas por essa implantação.

Dessa forma o ciclo é completado, podendo seguir daí para a criação de novas iterações do mesmo ou para abordagens diferentes usando a mesma lógica cíclica definida. Neste trabalho em questão, foram alocadas as etapas do Ciclo de Design. O Ciclo de Engenharia não pôde ser aplicado devido a questões de tempo hábil.

3 REFERENCIAL TEÓRICO

Este capítulo apresenta o referencial teórico necessário para entender o contexto em que esse trabalho se apresenta. Na seção 3.1 é apresentada uma narrativa contendo as fases que dizem respeito ao cenário do desenvolvimento de jogos. Em seguida na seção 3.2, falaremos sobre o conceito específico de design de jogos. Na sequência a seção 3.3 vai tratar de explicar como se apresenta o mercado no contexto de jogos digitais como produto. Na seção 3.4 iremos abordar o cenário de desenvolvimento de Desenvolvimento de jogos indie. Na seção 3.5, onde serão abordados os conceitos sobre Sistemas Colaborativos, que possui relação direta com a proposta desse trabalho. Na sequência, na seção 3.6, falaremos sobre aspectos de Gestão de equipes criativas. Segue-se então para a seção 3.7 onde será apresentado o conceito de websocket, tendo em vista que essa alternativa foi utilizada no desenvolvimento do trabalho. Por fim na seção 3.8 é apresentado o conceito de sistemas de recomendação.

3.1 HISTÓRICO DO DESENVOLVIMENTO DE JOGOS

O alinhamento de esforços com um objetivo comum sempre se mostrou necessário para minimizar muitos problemas nessa busca. o processo de planejamento, execução e concretização de uma ideia ou tarefa. Seja pela divisão de custos, responsabilidades e problemas, ou buscando apoio criativo, emocional e técnico.

O mundo sempre esteve repleto de boas ideias, planos infalíveis, conceitos e inúmeras tentativas de criar algo realmente interessante, funcional e de qualidade. Porém quando decidimos realmente sair do mundo das ideias e transformar o que era teoria em prática nem tudo sai como deveria.

Analisando por algum tempo o mundo comum como ele é e como ele chegou até esse ponto, nós conseguimos perceber algumas claras nuances, principalmente no comportamento dos seres em geral. A maioria possui uma predisposição a se agrupar em comunidades, grupos e até mesmo sociedades complexas. Esse comportamento está diretamente ligado à intenção de compartilhar e dividir, desde responsabilidades, problemas, recompensas, custos, conhecimentos, qualidades, etc. Quase sempre com um objetivo comum a todos os indivíduos participantes.

Quando colocamos o ser humano na discussão, nós conseguimos perceber que devido ao nosso desenvolvimento e complexidade, incorporamos esse conceito também no processo de desenvolvimento de projetos no geral. O intuito de se agrupar, nesse caso em

times ou equipes, se manteve o mesmo, no entanto novos objetivos começaram a ganhar destaque. O principal deles é dar vazão a nossas ideias e torná-las realidade.

Com isso em mente, diversos projetos durante a história contaram com equipes que os desenvolviam. Algumas grandes, como a que levou o homem à lua, outras pequenas como a responsável por desenvolver um recente fenômeno na indústria dos jogos eletrônicos, Cuphead. Apesar da diferença notável de importância, investimento e grandiosidade, ambas têm algo em comum: **A presença do conceito de programação de computadores como um dos pilares para o funcionamento de ambas as ideias.**

Como fruto de qualquer inovação, o ser humano expande as suas possibilidades de pensamento, resolução de problemas e de ideias. Quando Alan Turing, em 1937, teorizou sobre a máquina de Turing ([BARKER-PLUMMER, 2004](#)), ele provavelmente imaginava que a existência dela traria para o ser humano uma gama enorme de novos caminhos. Porém, talvez não conseguisse supor o tamanho dessa influência, de forma que sua pesquisa se expandisse ao ponto de ser parte intrínseca da vida de quase qualquer ser humano hoje.

Embora talvez isso pareça um pouco exagerado, é apenas uma constatação da realidade. Vários âmbitos essenciais da nossa sociedade hoje fazem uso da programação, mesmo que para a ponta consumidora do processo isso às vezes seja esquecido. Bancos, aparelhos domésticos, alimentação, transporte, gerência, saúde, meio ambiente, comunicação, entretenimento, todas essas esferas da sociedade possuem, em algum nível, influência da programação. Direta ou indiretamente.

O celular e o computador, que hoje são itens presentes na vida da maior parte da população mundial, são exemplos claros de projetos que estão diretamente ligados à programação. Os aplicativos, programas e jogos digitais que consumimos nesses dispositivos nasceram graças a essa inovação.

No entanto, apesar da programação trazer consigo inúmeras novas possibilidades, ela não conseguiu evitar de ter problemas semelhantes aos que projetos de outras áreas enfrentam. O gerenciamento de projeto de software e a gestão da equipe sempre deram muito trabalho a todos os envolvidos, exigindo a criação e implementação de meios para minimizar os problemas.

Quando o conceito de Engenharia de Software é criado e inserido nesse meio em 1968 na conferência da OTAN ([WIRTH, 2008](#)), diversas vertentes e definições surgem buscando esquematizar a melhor forma de planejar e desenvolver software. Cada uma defende suas verdades, conceitos e métodos, mantendo-se igual apenas o objetivo final.

Porém a realidade é um pouco mais complexa. É natural pensar que após o surgimento do conceito de Engenharia de Software, os problemas de qualidade, eficiência e funcionalidade do produto final se tornaram cada vez menores, até o ponto de desaparecerem. No entanto, seja por negligência, falta de conhecimento, prazos apertados ou pressa, muitos projetos de software resultam em produtos ruins.

Comprovar isso chega a ser simples. Basta questionar alguém que já teve a oportunidade de trabalhar, gerenciar ou apenas acompanhar de perto a execução de um projeto dessa natureza. Certamente a pessoa lhe contará diversos casos onde os fatores acima citados aparecem, às vezes para atrapalhar o andamento do projeto ou até mesmo para comprometer totalmente a sequência dele.

3.2 DESIGN DE JOGOS

Embora seja considerado muitas vezes uma forma de atrasar o desenvolvimento e engessar o processo, o design de jogos ocupa papel de organização das ideias e na modelagem dos aspectos de um jogo. Isso permite que parâmetros sejam estabelecidos e auxiliem na tomada de decisões durante o processo.

O conceito, às vezes ignorado, é imprescindível quando o objetivo é ter algo concreto e coerente nas mãos. Ele se refere basicamente ao ato de planejar, desde conceitos, intenções, premissas, tema, mecânica e tudo mais que possa compor um bom jogo. Para tal, é preciso que exista uma visão clara do que se pretende alcançar ao fim do desenvolvimento como um todo ([ENGSTRÖM, 2020](#)).

Com o tempo esse pilar foi sendo cada vez menos considerado. À medida que o deslumbramento com novas tecnologias e possibilidades surgia, o destaque dado e o tempo dedicado a produzir e explorar novas ideias foi se perdendo e o foco foi mudando até assumir um caráter muito mais comercial e de reaproveitamento. Tendo uma “fórmula pronta”, o design se tornou mais uma "pedra no sapato", aos olhos dos desenvolvedores, do que um caminho produtivo ([BURGUN, 2019](#)).

3.3 MERCADO DE JOGOS DIGITAIS

O mercado de jogos digitais começou discretamente a partir do momento que se percebeu o potencial comercial e apelativo que esse entretenimento possuía. No entanto, nos primeiros anos do século era possível perceber um emergente crescimento da presença de consoles nos lares das pessoas. A partir desse ponto, empresas de jogos começaram a investir suas receitas dedicando-se, principalmente, na pesquisa e inovação focada nessa área ([CRANDALL; SIDAK, 2006](#)).

Com a evolução da nossa sociedade e das tecnologias, o mercado dessa área só aumentou de tamanho e se diversificou. O poder de compra cresceu e o acesso a esse tipo de produto se tornou fácil. Sejam jogos indie ou mainstream, é facilmente percebido que é parte cada vez mais presente da nossa realidade. Eles movimentam uma renda enorme e produzem um ambiente que hoje é mais lucrativo que indústrias de entretenimento tradicional como a de filmes e música ([WEPC, 2020](#)).

Devido ao crescimento do interesse e da compra de jogos em suas diversas formas e plataformas, se desenvolveu todo um ecossistema focado no tema. Hoje em dia podemos encontrá-lo em forma de transmissão ao vivo ou por demanda e criação de conteúdo, competições profissionais, como plano de carreira, na resolução de problemas, sendo campo de estudo ou disponíveis na palma da mão ou ao alcance de um clique.

Ao se considerar isso, é muito difícil perceber um desaquecimento desse ramo de negócios no futuro, visto que a projeção é de que nos próximos essa indústria aumente de valor e arrecade cada vez mais, assim como tem feito desde o início do século ([WEPC, 2020](#)).

3.4 DESENVOLVIMENTO DE JOGOS INDIE

O desenvolvimento de jogos indie surgiu como forma de ir em direção contrária a racionalização e o teor profissional da grande indústria de jogos digitais, denominada “*mainstream*”. Esse universo mais independente, como o próprio nome sugere, oferece uma fuga dessa realidade mais formal e uma aproximação, para os desenvolvedores, do sentimento lúdico que os amantes de jogos encontram ao jogar ([FREEMAN; MCNEESE, 2019](#)).

Nesse tipo de desenvolvimento o número de pessoas envolvidas é geralmente pequeno e conforme o processo avança, é comum o acúmulo de funções que normalmente seriam executadas por times ou pessoas específicas na grande indústria de jogos. No entanto para a maioria que faz parte dessa comunidade, isso é fundamental pois dá controle criativo total aos idealizadores da ideia, e nesse caso boas ideias são mais importantes que recursos ([ZAMBON; CHAGAS,](#)).

Esse crescimento e consolidação se deve principalmente devido a plataformas como Steam, Playstation Network (PSN), etc. Elas permitem aos desenvolvedores fazer com que seus jogos sejam vistos e adquiridos pelas pessoas que se interessarem. Além disso, o aumento da popularidade e facilidade de compras e transações monetárias online facilitou para que o movimento indie pudesse ser impulsionado, seja com projetos financiados pela comunidade (*crowdfunding*), ou apenas pelo aumento de pessoas que podiam facilmente

comprar os jogos.

3.5 SISTEMAS COLABORATIVOS

Sistemas colaborativos surgiram e se popularizaram devido a uma demanda criada pelo rápido avanço da nossa sociedade e as peculiaridades percebidas no ambiente de trabalho moderno, circundado por uma realidade muito mais tecnológica e heterogênea (ALMEIDA; BARANAUSKAS, 2008).

Sua principal função é oferecer um ambiente de relação de colaboração que engloba elementos de comunicação, manutenção dos objetos colaborativos, consciência e percepção dos membros do ambiente para com os demais participantes e uma certa maleabilidade quanto a realidade experimentada pelos utilizadores.

Esse tipo de sistema considera fortemente as necessidades existentes em cada contexto de trabalho, levando em consideração que atualmente é muito mais difícil se adaptar às diferentes circunstâncias que possam se apresentar dentro de uma mesma equipe. Desde espaços geográficos diferentes, até meios de acesso e fluxo de trabalho completamente distintos.

Para desenvolver um sistema colaborativo hoje em dia é preciso estar ciente de que se está lidando com um ambiente vivo que pode possuir um apanhado muito mutável de material intelectual, coordenativo e de ações. O sistema deve se desprender de somente considerar as questões estruturais e de controle do processo de colaboração e entender as nuances que cercam o contexto específico ao qual se dedica. Ele deve atentar para a natureza mais pessoal, interativa e menos rígida e dinâmica em comparação com o que já se pensou ser a forma mais eficiente e correta de colaborar em um ambiente de trabalho e projeto (COSTA; PIMENTEL, 2011).

3.6 GESTÃO DE EQUIPES CRIATIVAS

Gerenciar equipes em projetos que envolvem atividades criativas é um tema complexo e uma preocupação que faz parte da realidade desses projetos. Fazer isso em contextos mais mecânicos já é complicado, quando colocamos o fator “criatividade” na equação, a complexidade é multiplicada. É preciso lidar com divergência de visão sobre o que está sendo criado, liderança, maturidade, experiência, sobrecarga de ideias, paciência e vontade de cada membro. Qualquer desarmonia pode gerar uma ruptura (FREEMAN; MCNEESE, 2019).

Atualmente, além da possibilidade de existirem equipes com esse perfil em uma mesma sala física ou prédio, também é preciso lidar com a realidade online em que

muitos de nós vivemos. Estranhos ou conhecidos podem estar presentes nos dois casos, dependendo da perspectiva em questão, tornando todo cuidado e atenção pouco. Ainda mais se engajamento e interesse forem as únicas coisas que mantêm a equipe firme, ou seja, na falta de consequências mais formais ou financeiras ao redor do grupo.

Pode acontecer de a sintonia entre os membros ser quase instantânea e às vezes chegar a gerar amizades duradouras, porém a possibilidade de que o oposto ocorra sempre existe e deve ser considerada e manuseada com resiliência. Ainda não há um consenso sobre a dose de profissionalismo e de intimidade correta nesses casos, mas sabe-se que comunicação e ciência do seu papel na equipe faz toda diferença ([ČURČIĆ, 2018](#)).

3.7 WEBSOCKET

As aplicações para web estão cada vez mais se mostrando dependentes de alternativas que proporcionem uma comunicação em tempo real e bidirecional para fazer troca de mensagens e dados da maneira mais eficiente possível. E mesmo que o modelo cliente servidor tenha sido muito explorado para prover tal funcionalidade, a maioria das técnicas utilizadas, como o HTTP Polling, se mostraram ineficientes e com problemas ([MURLEY et al., 2021](#)).

Dessa forma surgiu uma alternativa mais atraente a estes métodos já existentes, a API de WebSockets. Se aproveitando ao máximo do protocolo HTTP e da camada de segurança usada nos navegadores web, o TCP, a técnica de WebSocket oferece tudo que as aplicações em tempo real precisam, e soluciona os problemas antes encontrados. Isso acontece devido a forma como o seu protocolo foi estabelecido.

O protocolo é constituído por duas partes. A primeira consiste de um *handshake* enviado pelo cliente e um enviado pelo servidor. Logo que essa etapa seja concluída com sucesso, a conexão está firmada e a troca de mensagens pode acontecer tanto de um lado quanto do outro, desde que se respeitem as regras de segurança e as credenciais estabelecidas ([FETTE; MELNIKOV, 2011](#)).

3.8 SISTEMAS DE RECOMENDAÇÃO

Sistemas de recomendação estão fortemente presentes na vida do indivíduo moderno que se comunica de uma forma ou de outra com o mundo da internet. E mesmo que isso não seja algo tão recente, sua presença se acentuou à medida que consumimos produtos e serviços online e cedemos informações sobre as nossas preferências e gostos aos sites e aplicativos que utilizamos.

A premissa básica de um sistema de recomendação é a de que um recurso que

interesse a um indivíduo em uma determinada circunstância, provavelmente também interessa a outro indivíduo em circunstâncias parecidas ou com interesses similares. Portanto, a maioria das estratégias de recomendação utilizadas nos sistemas deste tipo vão partir desse princípio ([MOTTA et al., 2012](#)).

Neste trabalho em questão, a estratégia de recomendação utilizada foi a de filtragem colaborativa. Nessa abordagem os recursos recomendados são aqueles que foram considerados bons ou válidos por indivíduos com os quais o usuário em questão possui um perfil de interesse parecido ([COLMENERO-FERREIRA; OLIVEIRA, 2012](#)).

No entanto, como o foco principal não é aprofundar-se no tema, o sistema de recomendação criado e incluído na plataforma tem como função apenas demonstrar a usabilidade deste tipo de serviço em aplicações como a desenvolvida no presente trabalho.

4 FERRAMENTAS SIMILARES

Neste capítulo estão apresentadas ferramentas que possuem características e funcionalidades semelhantes as que estão presentes na solução proposta por esse trabalho, tendo em vista algumas delas serviram de inspiração ou apoio para o desenvolvimento da pesquisa, pela proximidade e familiaridade de uso por desenvolvedores de jogos indie, os usuários alvos da plataforma.

Através das pesquisas realizadas sobre o contexto de desenvolvimento de jogos, as equipes criativas relacionadas a este tipo de projeto e durante o processo de formular uma proposta de solução, não foram encontradas ferramentas que se encaixassem no contexto específico de desenvolvimento de jogos e que desenvolvessem um trabalho igual ou parecido suficiente com a proposta desenvolvida nesse trabalho. Devido a isso as ferramentas que são apresentadas a seguir não são estritamente relacionadas.

4.1 TRELLO

O Trello¹ é uma plataforma online de gerenciamento de projetos no geral. Ele se baseia na metodologia Kanban para organizar a disposição do projeto para os usuários envolvidos. Seja na versão grátis ou paga (*premium*), é oferecida uma gama interessante de ferramentas e possibilidades, seja na versão grátis ou paga. Com ele é possível gerenciar prazos, estabelecer métricas de produtividade, acompanhar histórico das atividades, entre outras atividades. Também é permitida a inclusão de extensões como forma de automatizar certas etapas da gerência.

Certamente, o Trello, cumpre sua função de gerenciamento de equipes de todos os tipos de projeto de forma satisfatória. Com uma interface fácil de ser aprendida e utilizada, uma certa robustez na hora de fazer a manutenção do fluxo de trabalho, ele cumpre um papel interessante no meio de gerência de projetos. No entanto, às vezes peca por excesso de elementos nos quadros e não apresenta funcionalidades específicas para auxiliar equipes focadas no desenvolvimento de projetos com foco em criatividade. Nem tão pouco aborda análise dos projetos alocados em sua plataforma buscando métricas que possam ser úteis para novos utilizadores.

¹ **Trello**. Disponível em: <<https://trello.com/pt-BR>>

4.2 KANBANFLOW

O KanbanFlow² é uma plataforma online, assim como o Trello, também se baseia, como o próprio nome sugere, na metodologia Kanban para oferecer uma alternativa mais simples ao gerenciamento, nesse caso não somente de projetos, mas também de tarefas no geral. Uma das intenções principais dessa ferramenta é aumentar a produtividade dos envolvidos nos projetos e equipes trazendo uma interface simples e objetiva.

Possui duas versões disponíveis de assinatura, grátis e paga (*premium*). Na versão sem custos no entanto disponibiliza poucas possibilidades, deixando para a versão paga funcionalidades como inclusão de arquivos nas tarefas, relações entre elas, pesquisas, sistema de calendário, entre outras. Existe ainda a possibilidade dividir o projeto em times e criar tarefas recorrentes, mas com poucas extensões para aumentar a abrangência e aumentar um pouco mais sua especificidade.

4.3 NOTION

O Notion³ é um aplicativo para Android e IOS que permite a pessoas individualmente ou a times, listar tarefas para serem executadas. É permitido aos usuários organizar essas tarefas em grupos e criar subtarefas, além de acompanhar o status da tarefa. No caso de times é possível criar documentos colaborativos, delegar a membros e até comentar sobre a execução da mesma.

O aplicativo é grátis apenas para uso individual. Caso se tenha interesse em organizar as tarefas dentro de um time ou equipe é preciso pagar a assinatura. Apesar de parecer eficaz no que se propõe, é de longe o mais simples de todos.

4.4 SLACK

O Slack⁴ é uma plataforma online de colaboração de equipes dentro de uma empresa. Seu foco principal é a melhoria da comunicação e do fluxo de trabalho entre os colaboradores e não exatamente a gestão do projeto em si. Para isso ele inclui um sistema de troca de mensagens, chamadas de vídeo e áudio, a possibilidade de compartilhar arquivos, entre outras, tanto individualmente como em forma de canais específicos para cada setor ou intenção.

Com ele também é possível incluir desde API's disponíveis na própria plataforma, até fazer conexão com diversos aplicativos externos parceiros. A intenção é centralizar o

² **KanbanFlow**. Disponível em: <<https://kanbanflow.com/>>

³ **Notion**. Disponível em: <<https://www.notion.so/>>

⁴ **Slack**. Disponível em: <<https://slack.com/intl/pt-br/>>

fluxo de trabalho e a comunicação em um só lugar.

Na versão grátis, o Slack limita suas possibilidades como o número de pessoas em chamadas e a quantidade de aplicativos externos disponíveis para serem adicionados. Porém ele disponibiliza a possibilidade de acesso a mais recursos de acordo com o perfil da empresa que o utiliza através de planos.

5 IMPLEMENTAÇÃO

Este capítulo se propõe a apresentar de maneira clara os detalhes de implementação referentes ao sistema desenvolvido, desde seus requisitos até a arquitetura utilizada.

5.1 APRESENTAÇÃO DO SISTEMA WINDIE

O software **Windie** é um sistema colaborativo e interativo focado em gestão de equipes criativas engajadas em design e desenvolvimento de jogos, principalmente aqueles que residem no contexto independente de produção.

Para tal finalidade o sistema conta com funcionalidades para gerenciar desde os próprios projetos a serem desenvolvidos, até os usuários envolvidos, suas avaliações, tarefas, trocas de mensagens e agrupamento em times.

Isso tudo pode ser acessado de qualquer dispositivo que possua conexão com internet e navegador, visto que o sistema foi desenvolvido para ser uma aplicação web.

5.2 REQUISITOS

Tendo por referência a análise feita do contexto em questão, suas peculiaridades e a abertura para que possam ser implementadas ideias de solução através do sistema a ser desenvolvido, a proposta deste trabalho pauta sua construção nos requisitos que estão dispostos e detalhados nas seções que se seguem. A Figura 5.1 traz uma visão geral dos requisitos do sistema.



Fonte: Elaborado pelo autor (2021)

RF₁ Gerenciamento de atividades e ciclos. O gerenciamento de atividades e ciclos tem como foco disponibilizar ferramentas e métodos que viabilizem e facilitem o ciclo de vida bem como a manutenção das tarefas e etapas do projeto gerenciado.

RF₂ Gerenciamento de equipes criativas. O gerenciamento de equipes criativas busca oferecer ferramentas e funcionalidades dedicadas para realizar a manutenção dos colaboradores dos projetos de maneira adaptada a equipes no contexto abordado, juntamente com meios de garantir o melhor desempenho de suas funções e acompanhar o fluxo de trabalho entre os membros.

RF₃ Sistema de recomendação de colaboradores por função. O sistema de recomendação de colaboradores organizado por função busca certificar que os projetos desenvolvidos possam dispor de membros dedicados para funções importantes do desenvolvimento e com uma avaliação adequada para tentar garantir uma maior robustez na equipe. Os detalhes de como o algoritmo de recomendação será construído ainda estão em aberto.

RF₄ Ranking de projetos / colaboradores. O ranking de projetos e colaboradores visa criar confiança dentro da comunidade ao evidenciar projetos que estão apresentando um bom andamento durante o seu ciclo de vida, além de explicitar o nível de confiabilidade demonstrado por cada colaborador individualmente.

RF₅ Ferramentas de “descoberta” de projetos / colaboradores. A descoberta de projetos e colaboradores tem como objetivo permitir que a comunidade envolvida acompanhe o desenvolvimento individual dos colaboradores presentes na plataforma, bem como também o andamento e informações sobre os projetos que lhe despertam interesse.

RF₆ Integração com ferramentas existentes (Github/outras ferramentas). A intenção é criar o sistema com uma arquitetura flexível o suficiente para permitir integração com vários serviços, à medida que as necessidades individuais do usuário e do projeto foram sendo identificadas.

5.3 ARQUITETURA

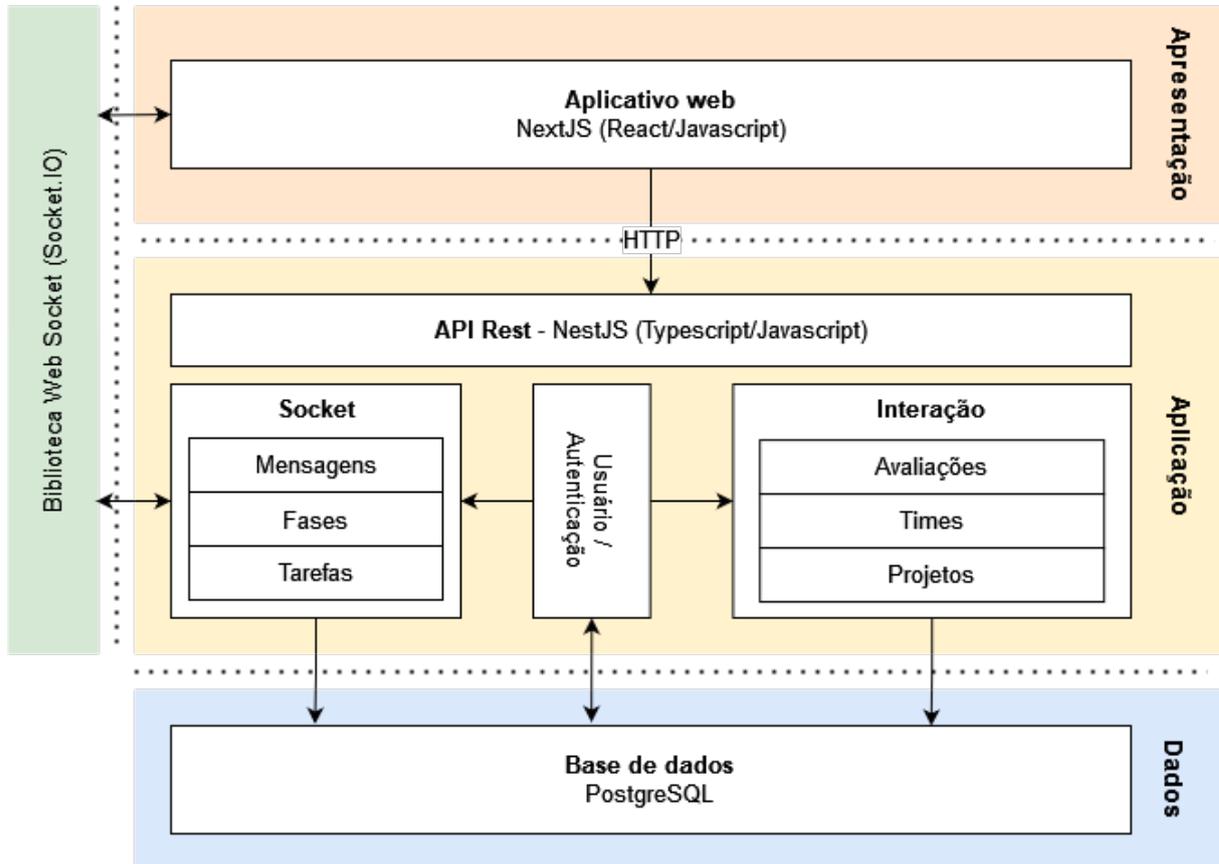
A arquitetura utilizada foi a de cliente servidor, ou seja, o sistema possui uma camada de apresentação disponibilizada para os clientes conectados através de uma aplicação web para que esses possam acessar as funcionalidades oferecidas por um servidor NodeJS. Esse por sua vez se conecta com um SGBD para gerenciar as informações importantes para os usuários do sistema e para o seu funcionamento geral.

A intenção foi construir um sistema com usabilidade focada em desenvolvedores de jogos, buscando inspiração em ferramentas já conhecidas e utilizadas para facilitar a adaptação.

A arquitetura montada para esse sistema consiste de 3 camadas principais e uma

seção paralela com a qual elas se comunicam para prover alguns serviços. A imagem que representa a arquitetura do sistema é mostrada na Figura 5.2.

Figura 5.2 – Arquitetura do sistema Windie



Fonte: Elaborado pelo autor (2023)

A camada de apresentação consiste de um aplicativo web desenvolvido em NextJS, um framework javascript baseado no react. O Aplicativo web foi feito de modo que seja acessado pelo cliente através do seu navegador, possuindo um mínimo de responsividade para não prejudicar a experiência independente do dispositivo que esteja sendo utilizado. Considerando que equipes criativas muitas vezes se encontram em locais separados.

A camada de aplicação é composta por um servidor utilizando o framework NestJS baseado em NodeJS, juntamente com Typescript para maior robustez da aplicação. O protocolo utilizado para a comunicação entre essas duas camadas é o HTTP, já que o servidor nesse caso nada mais é que uma API Rest que dispõe das funcionalidades que satisfaçam as necessidades dos clientes.

Esta camada por sua vez divide-se em 3 componentes distintos. Um componente focado nas funcionalidades e validações inerentes aos usuários dos sistema, outro para as interações básicas com os módulos gerais como projetos, avaliações e times. E por

último um componentes denominado de Socket, pois engloba os módulos que fazem uso da biblioteca de Web Socket para prover dados em tempo real para os usuários. Os módulos envolvidos são de mensagens, fases e tarefas.

A biblioteca Socket.IO é utilizada tanto pela camada de aplicação quanto pela camada de apresentação através do protocolo HTTP, pois é preciso que a comunicação seja aberta dos dois lados para garantir a troca de dados em tempo real.

A camada de dados conta apenas com uma base criada no SGBD PostgreSQL para gerenciamento dos dados da aplicação. Essa conexão é feita através de um pacote instalado no servidor NodeJS que fica responsável por fazer essa integração e acessar a base através do TypeORM, uma biblioteca usada para realizar o mapeamento das entidades.

5.4 REQUISITOS IMPLEMENTADOS

5.4.1 RF1 Gerenciamento de fases e tarefas

O requisito do gerenciamento de fases e tarefas busca prover um quadro *kanban* que pode ser preenchido com as fases do desenvolvimento do projeto e em cada fase a possibilidade de ter tarefas atreladas a ela.

Sempre que um novo projeto é criado, automaticamente são criadas quatro fases por padrão pelo sistema. Essas fases podem ser mantidas pela usuário, como também podem ser excluídas ou modificadas de acordo com a necessidade do projeto.

As propriedades das tarefas e das fases, já incluindo a ordem de cada uma, a localização delas no quadro podem ser alteradas conforme os usuários preferirem e essa alteração será refletida em tempo real para os outros envolvidos no projeto. Para que isso fosse possível, se utilizou mais uma vez da biblioteca Socket.IO.

A tela que representa o requisito em questão, pode ser visualizada na Figura 5.3.

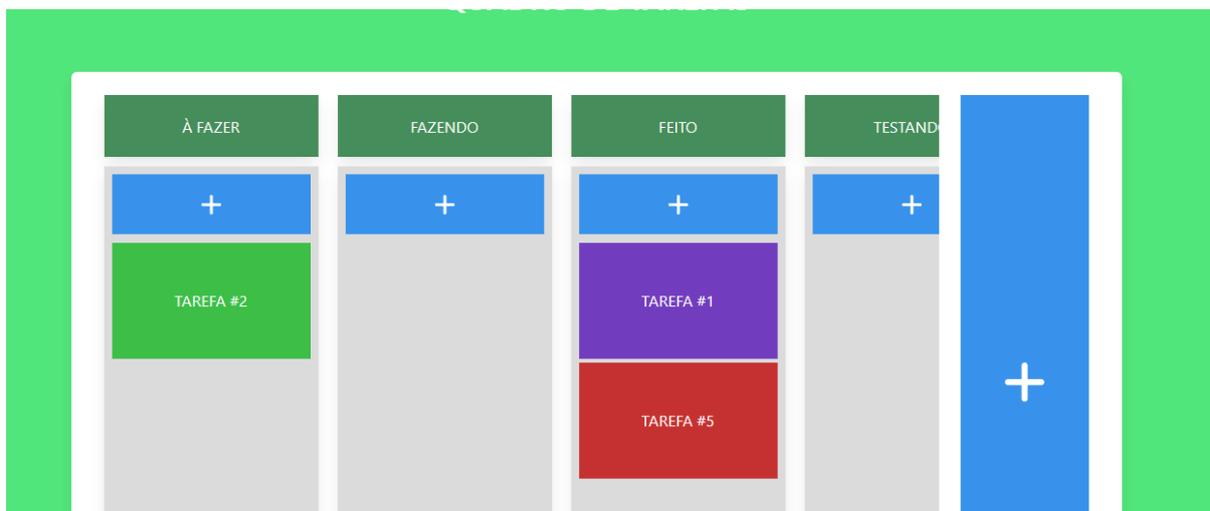
5.4.2 RF2 Gerenciamento de equipes criativas

O requisito de gerenciamento de equipes criativas busca prover aos usuários do sistema a possibilidade de gerenciar os times dentro dos seus projetos.

O gerente do projeto pode pesquisar e convidar usuários para colaborar com o projeto ou excluir alguns que não sejam mais colaboradores do mesmo.

O usuário convidado recebe um email com a mensagem solicitando sua colaboração. Aceitando ele passa a ter permissão para acessar as informações privadas do projeto do

Figura 5.3 – Tela do quadro de tarefas



Fonte: Elaborado pelo autor (2022)

qual agora ele é colaborador e também a possibilidade de interagir com a sala de reuniões, o quadro de tarefas e afins.

O gerente pode então selecionar a partir dos colaboradores do projeto, aqueles que devem integrar cada time presente no mesmo. Isso garantirá uma melhor distribuição e organização das tarefas em grupos específicos para cada uma delas.

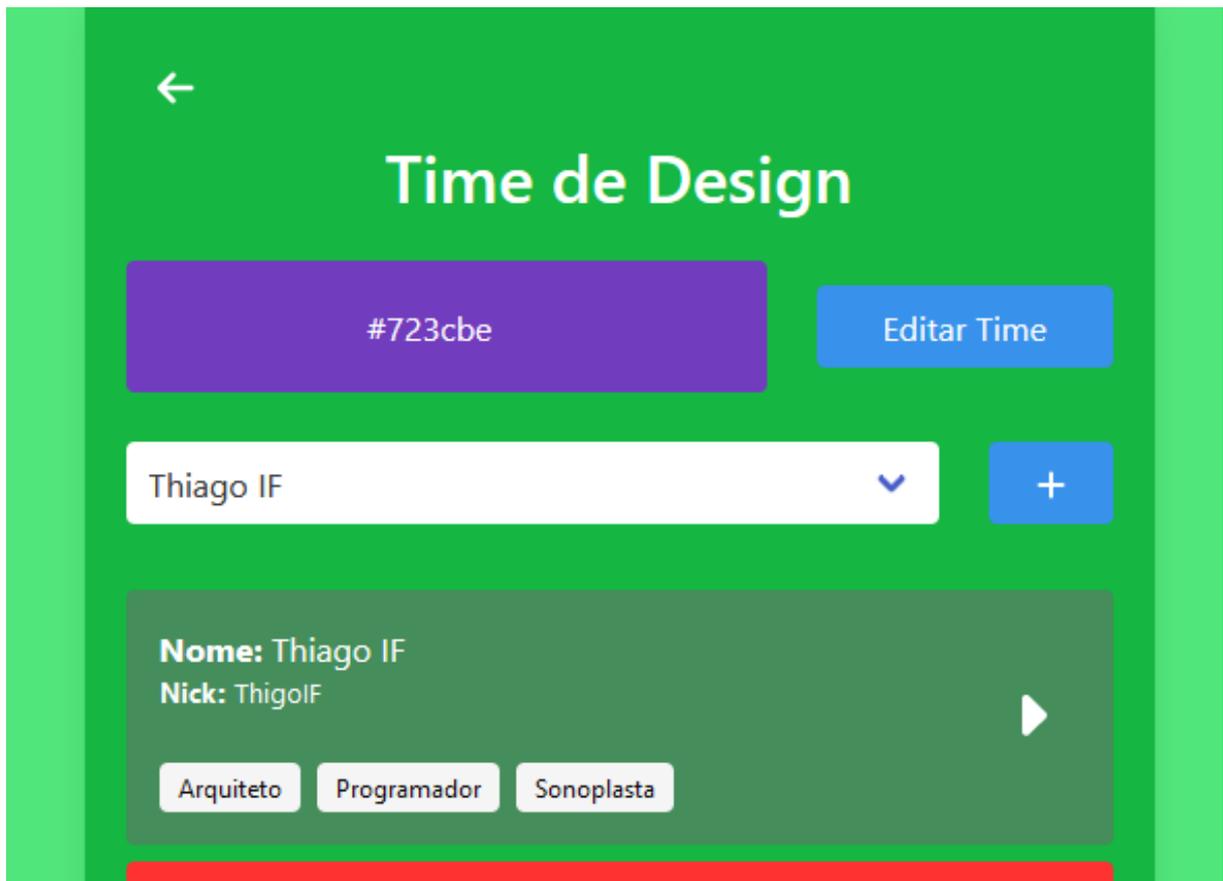
A figura 5.4 mostra a tela de perfil de perfil e gerenciamento de times no sistema. Através dela novos integrantes podem ser adicionados ao time dentre os colaboradores atuais do projeto. O apêndice B contém a figura da tela de pesquisa e convite de colaboradores.

5.4.3 RF3, RF4 e RF5 Avaliação e recomendação de usuários e projetos

O requisito de avaliação e recomendação de usuários e projetos busca trazer a possibilidade dos usuários poderem avaliar, dar notas, aos demais usuários e projetos presentes no sistema. Usando essas avaliações, o sistema provê recomendações de perfis de usuários e projetos tanto a usuários logados quanto a usuários anônimos que apenas estão procurando explorar novos perfis. A Figura 5.5 mostra tanto a tela de perfil de usuário com a opção de avaliar e sua nota atual, quanto a tela de descoberta de usuários e projetos já com os usuários recomendados pelo sistema baseado no algoritmo utilizado.

O algoritmo utilizado para realizar a recomendação dos perfis foi o algoritmo de recomendação baseado em filtragem colaborativa. Esse algoritmo consiste em analisar padrões de comportamento parecidos com os do usuário alvo da recomendação e supor os itens que possam lhe interessar baseado nisso.

Figura 5.4 – Tela de perfil do time



Fonte: Elaborado pelo autor (2022)

Para recomendar usuários, o algoritmo leva em consideração usuários que fizeram avaliações parecidas com o usuário alvo e que não tenham tido contato com o ele ainda.

Para recomendar projetos, o algoritmo considera projetos que o usuário alvo ainda não teve contato e que foram avaliados bem por usuários que já trabalharam com ele.

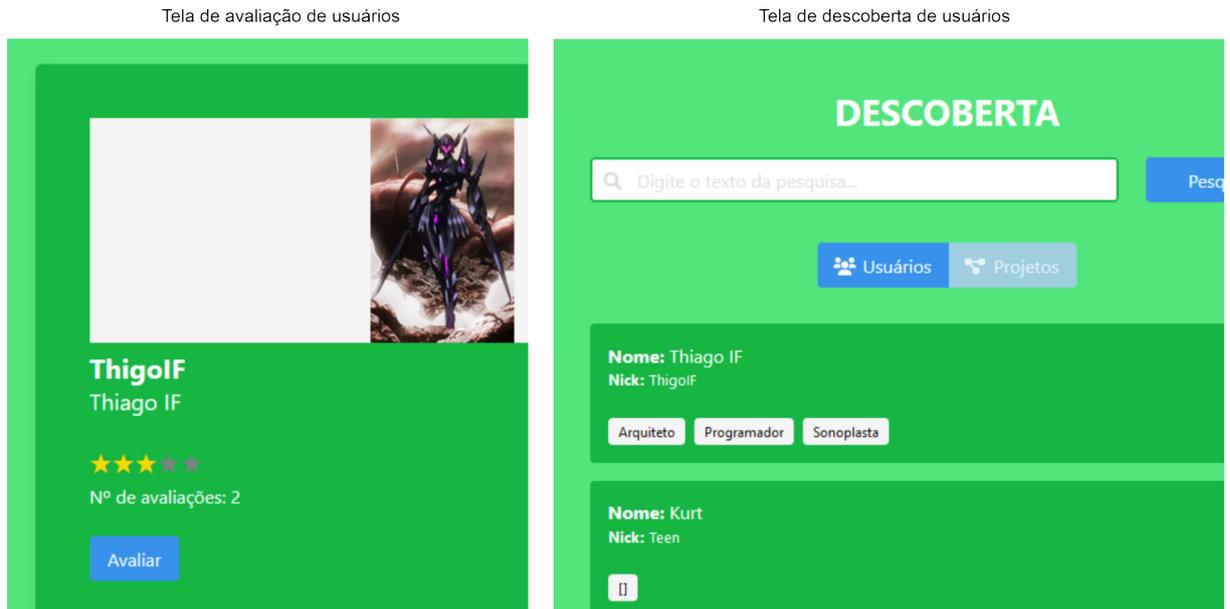
O algoritmo utilizado é demonstrado na Figura 5.6.

5.4.4 RF6. Integração com ferramentas existentes

O requisito em questão consiste em construir o sistema com uma arquitetura flexível que permita a integração com ferramentas já existentes no mercado e que permitam tal interação.

Para a implementação correta desse requisito, o sistema foi construído em cima de uma arquitetura cliente servidor, tendo este sido desenvolvido totalmente sob o conceito de modularização. Dessa forma a integração total ou parcial pode ser feita posteriormente sem que atrapalhe a execução e experiência com o sistema ou impeça o funcionamento

Figura 5.5 – Avaliação e recomendação de usuários



Fonte: Elaborado pelo autor (2022)

geral do mesmo.

Os detalhes sobre a arquitetura utilizada estão descritos na seção 6.3 deste documento.

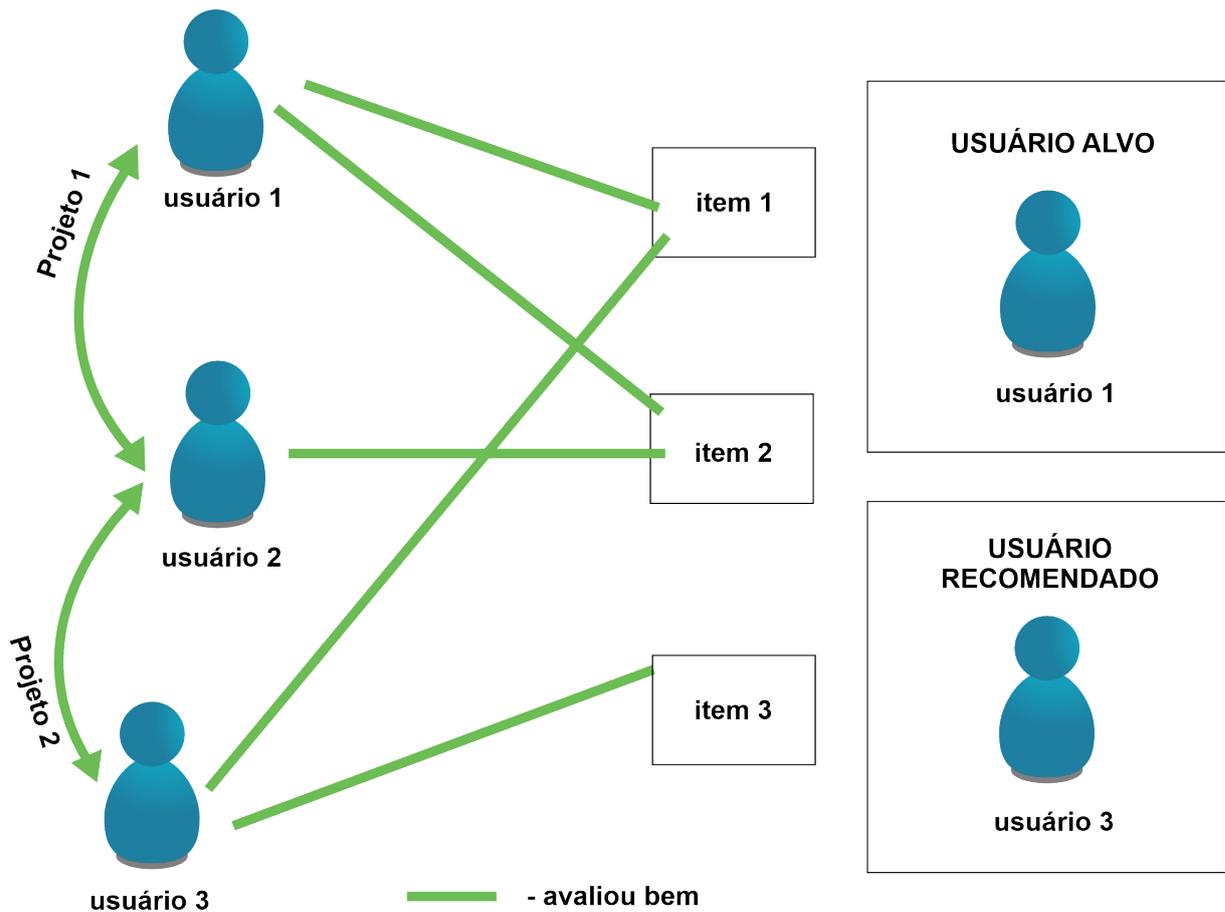
5.4.5 RF7. Gerenciamento de imagem do perfil de usuários e projetos

O requisito de gerenciamento de imagem de perfil de usuários e projetos busca prover a capacidade de encaminhar um arquivo de imagem, de sua escolha, através do formulário de cadastro para que ela seja salva pelo sistema e possa ser visualizada posteriormente. O apêndice A mostra a tela de formulário para envio da imagem de perfil do usuário.

Para a implementação desse requisito, foi utilizado o serviço de armazenamento provido pela plataforma Firebase. A imagem é recebida pelo servidor da aplicação, processada e enviada para o serviço do Firebase utilizando a biblioteca para javascript `firebase-admin`.

Depois que o arquivo é salvo pelo Firebase, o serviço retorna um link que é persistido em uma base de dados relacional na tabela referente a entidade. Sempre que o usuário solicitar essa informação através do front-end da aplicação, o link para acesso é recuperado da base de dados e a imagem é exibida para o usuário. A Figura 5.7 representa o fluxo de processamento do arquivo dentro do sistema.

Figura 5.6 – Tela de demonstração do algoritmo de recomendação



Fonte: Elaborado pelo autor (2022)

5.4.6 RF8. Sala de reuniões

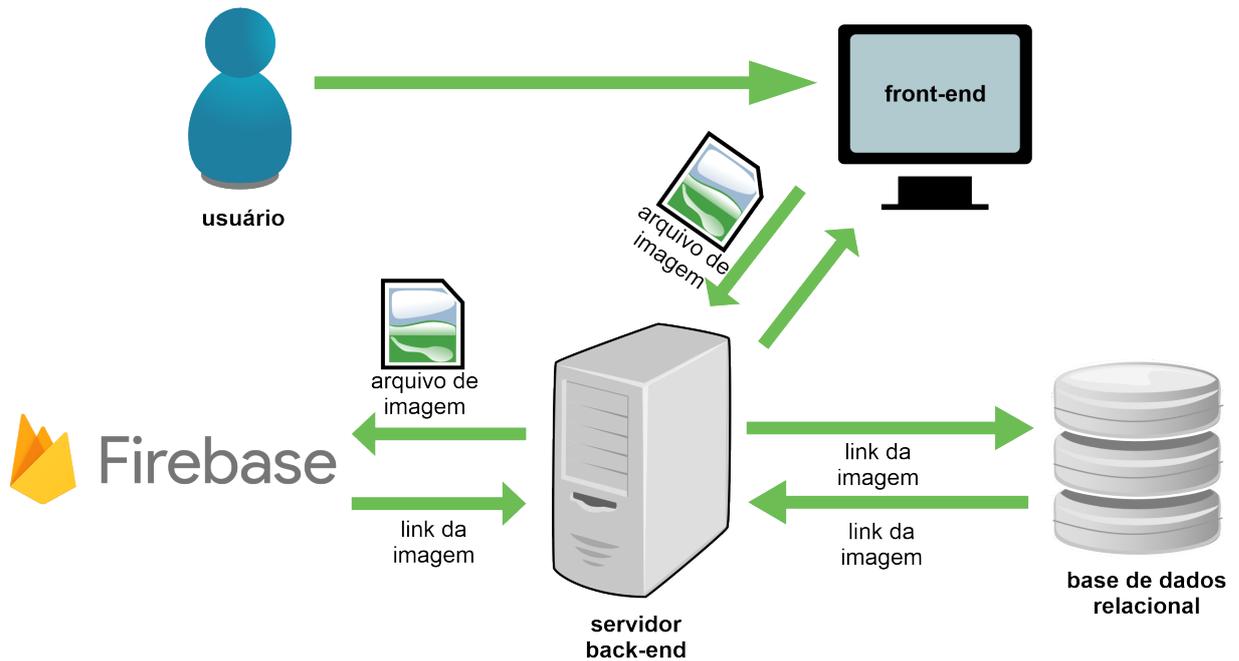
O requisito sala de reuniões busca prover aos usuários participantes de um projeto no sistema, a capacidade de poder se comunicar através de mensagens de texto. O apêndice C mostra a tela de sala de reuniões contida no sistema.

Para a implementação desse requisito de forma correta, foi utilizada uma biblioteca orientada a eventos para aplicações em tempo real, chamada Socket.IO¹. Através do uso dessa biblioteca, foi possível permitir que um usuário envie uma mensagem para a sala de reuniões e um outro usuário logado no sistema e dentro da mesma tela visualize essa mensagem em tempo real. O fluxo de funcionamento da biblioteca está descrito na Figura 5.8.

A mensagem enviada fica persistida em uma base relacional criptografada e atrelada ao projeto em questão. Dessa forma apenas membros do deste conseguem acessar o seu conteúdo e enviar mensagens para seus companheiros de trabalho.

¹ **Socket.IO**. Disponível em: <<https://socket.io/pt-br/>>

Figura 5.7 – Fluxo de processamento da imagem no sistema



Fonte: Elaborado pelo autor (2022)

5.4.7 Outros requisitos

Os demais requisitos, por não terem detalhes de implementação que tragam nenhuma necessidade de explicação detalhada não serão descritos.

5.5 COMPARAÇÃO COM OUTRAS FERRAMENTAS

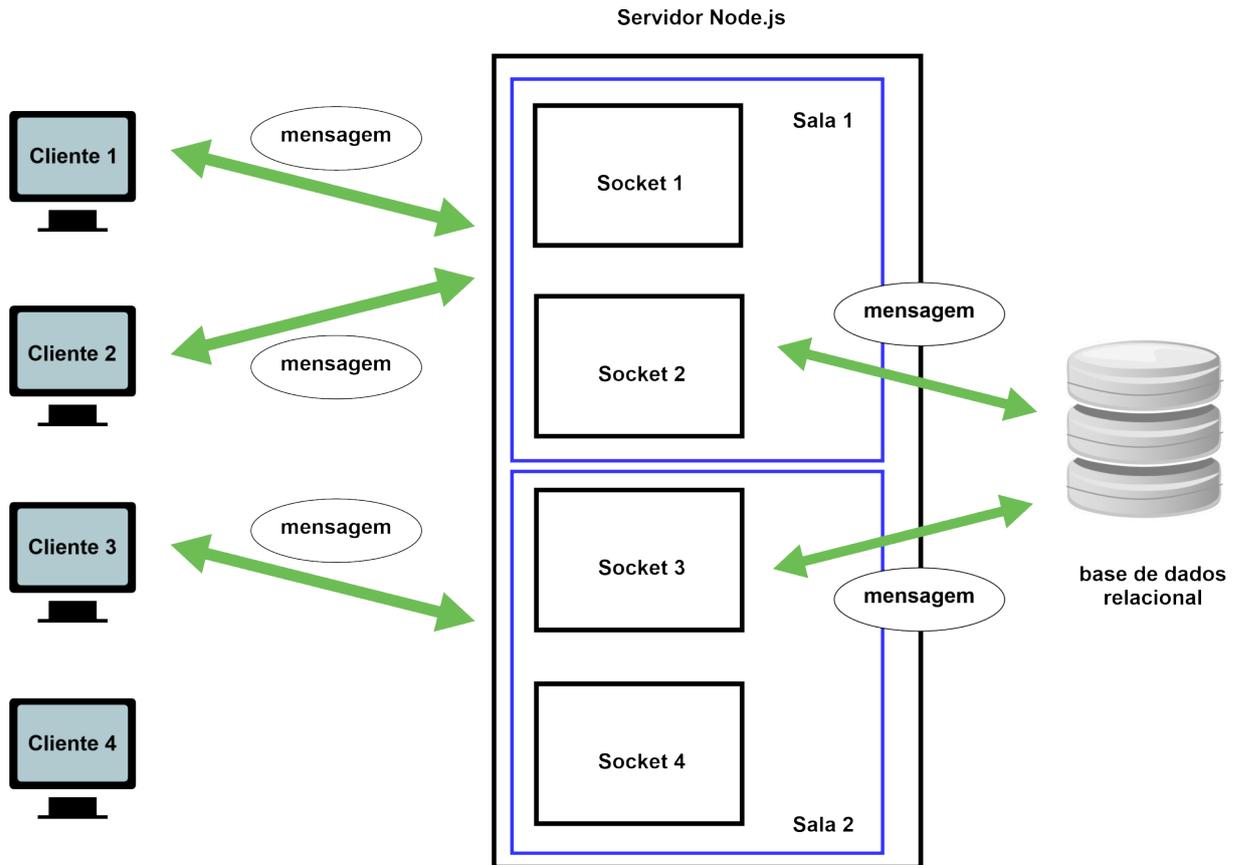
A comparação da proposta do sistema em relação a outras ferramentas ou abordagens já utilizadas na indústria será feito a partir de uma pesquisa qualitativa, ainda com técnicas a serem definidas. Essa atividade será desenvolvida em um trabalho futuro.

5.6 DIFICULDADES ENCONTRADAS

Durante a implementação do trabalho, alguns obstáculos foram encontrados e exigiram uma atenção maior para serem superados e dar prosseguimento com a execução. O primeiro deles partiu definição de uma arquitetura mais robusta e que se adequasse aos conhecimentos dos pesquisadores e a solução proposta.

O segundo obstáculo consistiu em organizar os dados armazenados no banco de forma que fizessem sentido e permitissem o desenvolvimento das funcionalidades como formuladas. Isso aconteceu porque o entendimento sobre a necessidade de algumas tabelas e como as informações estariam dispostas nelas, foram se alterando conforme a definição da plataforma ficava mais clara e as necessidades mais bem postas.

Figura 5.8 – Fluxo de funcionamento da biblioteca Socket.IO



Fonte: Elaborado pelo autor (2022)

No entanto, a maior dificuldade encontrada foi relacionada a necessidade de implementar duas funcionalidades com respostas quase em tempo real para o cliente. Tanto a sala de reuniões como o quadro de tarefas precisavam refletir as mudanças feitas por outros usuários envolvidos no projeto em tempo real para que pudesse ser mantido o conceito de sistema colaborativo. Isso exigiu uma pesquisa sobre as melhores formas de fazer isso conceitualmente e como implementar essa solução na prática.

6 CONSIDERAÇÕES FINAIS

Este trabalho teve como meta geral desenvolver uma plataforma de apoio ao desenvolvimento de jogos indie de forma que oferecesse ao mercado um sistema colaborativo mais adequado ao contexto abordado. Para tal, considerando-se as características inerentes e o caráter científico em prol de desenvolver uma solução prática para um problema real, o presente trabalho se pautou na metodologia Design Science para atingir tal objetivo, produzindo os artefatos esperados. Como forma de modularizar o objetivo geral, foram definidos quatro objetivos específicos.

Os dois primeiros estavam diretamente ligados, já que o primeiro estabelecia um levantamento das ferramentas utilizadas pelos desenvolvedores de jogos indie e para o segundo ficou pretendida uma comparação entre elas para definir pontos fracos e fortes. O levantamento foi concluído com sucesso e está disposto no capítulo sobre ferramentas semelhantes neste trabalho, servindo de base e ponto de partida para as demais intenções.

O terceiro por sua vez, definia propor uma ferramenta baseada nos requisitos e pontos de melhorias oriundos do levantamento e comparativo feito anteriormente, dando uma opção diferente e completa ao contexto apresentado. Este objetivo foi cumprido com sucesso e todas as etapas e detalhes de implementação, dificuldades e estrutura foram descritos e esclarecidos neste trabalho.

Por último ficou estabelecida a realização de uma avaliação da ferramenta através do desenvolvimento de um jogo exemplo, no entanto esse objetivo não pode ser concluído devido ao tempo hábil e a falta de indivíduos para participar da experimentação proposta, portanto não se pôde aplicar o que determina o segundo ciclo da metodologia estabelecida.

Algumas dificuldades foram encontradas durante a execução da pesquisa. Entre elas podem ser citadas a rasa abordagem e ou a limitada quantidade de pesquisas científicas dentro da temática de jogos e a baixa disseminação de conhecimentos sobre a área. Somando-se isso com o tempo limitado para o desenvolvimento do trabalho, também não foi possível comparar a plataforma desenvolvida com outras ferramentas semelhantes, mas fica como sugestão para pesquisas futuras, assim como a construção da avaliação da ferramenta em um exemplo prático do mundo real.

É possível concluir que a proposta estabelecida no presente trabalho foi bem sucedida ao desenvolver e oferecer ao contexto de desenvolvimento de jogos indie uma plataforma alternativa que pode melhorar a organização, interação e descoberta de colaboradores para tentar amenizar ou sanar as dificuldades encontradas no processo.

REFERÊNCIAS

- ALMEIDA, L. D. A.; BARANAUSKAS, M. C. C. Um prospecto de sistemas colaborativos: modelos e frameworks. In: **Proceedings of the VIII Brazilian Symposium on Human Factors in Computing Systems**. [S.l.: s.n.], 2008. p. 204–213.
- BARKER-PLUMMER, D. Turing machines. 2004.
- BAX, M. P. Design science: filosofia da pesquisa em ciência da informação e tecnologia. 2017.
- BURGUN, K. **Game design theory: A new philosophy for understanding games**. [S.l.]: AK Peters/CRC Press, 2019.
- COLMENERO-FERREIRA, F.; OLIVEIRA, A. A. d. Os sistemas de recomendação na web como determinantes prescritivos na tomada de decisão. **JISTEM - Journal of Information Systems and Technology Management**, TECSI Laboratório de Tecnologia e Sistemas de Informação - FEA/USP, v. 9, n. JISTEM J.Inf.Syst. Technol. Manag., 2012 9(2), May 2012. ISSN 1807-1775. Disponível em: <<https://doi.org/10.4301/S1807-17752012000200008>>.
- COSTA, A. M. Nicolaci-da; PIMENTEL, M. Sistemas colaborativos para uma nova sociedade e um novo ser humano. **Sistemas colaborativos**. PIMENTEL, M.; FUKS, H.(Orgs.). Rio de Janeiro: Elsevier, 2011.
- CRANDALL, R.; SIDAK, J. G. Video games: Serious business for america's economy. **Entertainment Software Association Report**, SSRN, 2006.
- ENGSTRÖM, H. **Game Development Research**. [S.l.]: University of Skövde, 2020.
- FETTE, I.; MELNIKOV, A. **The websocket protocol**. [S.l.], 2011.
- FREEMAN, G.; MCNEESE, N. J. Exploring indie game development: Team practices and social experiences in a creativity-centric technology community. **Computer Supported Cooperative Work (CSCW)**, Springer, v. 28, n. 3, p. 723–748, 2019.
- LOGUIDICE, B.; BARTON, M. **Vintage Game Consoles: an inside look at Apple, Atari, Commodore, Nintendo, and the greatest gaming platforms of all time**. [S.l.]: CRC Press, 2014.
- MOTTA, C.; GARCIA, A. C.; VIVACQUA, A.; SANTORO, F.; OLIVEIRA, J. Sistemas de recomendação. In: _____. [S.l.: s.n.], 2012. p. 230–244. ISBN 9788535250862.
- MURLEY, P.; MA, Z.; MASON, J.; BAILEY, M.; KHARRAZ, A. Websocket adoption and the landscape of the real-time web. In: **Proceedings of the Web Conference 2021**. [S.l.: s.n.], 2021. p. 1192–1203.
- VGInsights. **Infographic: Indie game revenues on Steam**. 2020. Disponível em: <<https://vginsights.com/insights/article/infographic-indie-game-revenues-on-steam>>.
- WEPC. **Video Game Industry Statistics, Trends and Data In 2021**. 2020. Disponível em: <<https://www.wepc.com/news/video-game-statistics/>>.

WIERINGA, R. J. **Design science methodology for information systems and software engineering**. [S.l.]: Springer, 2014.

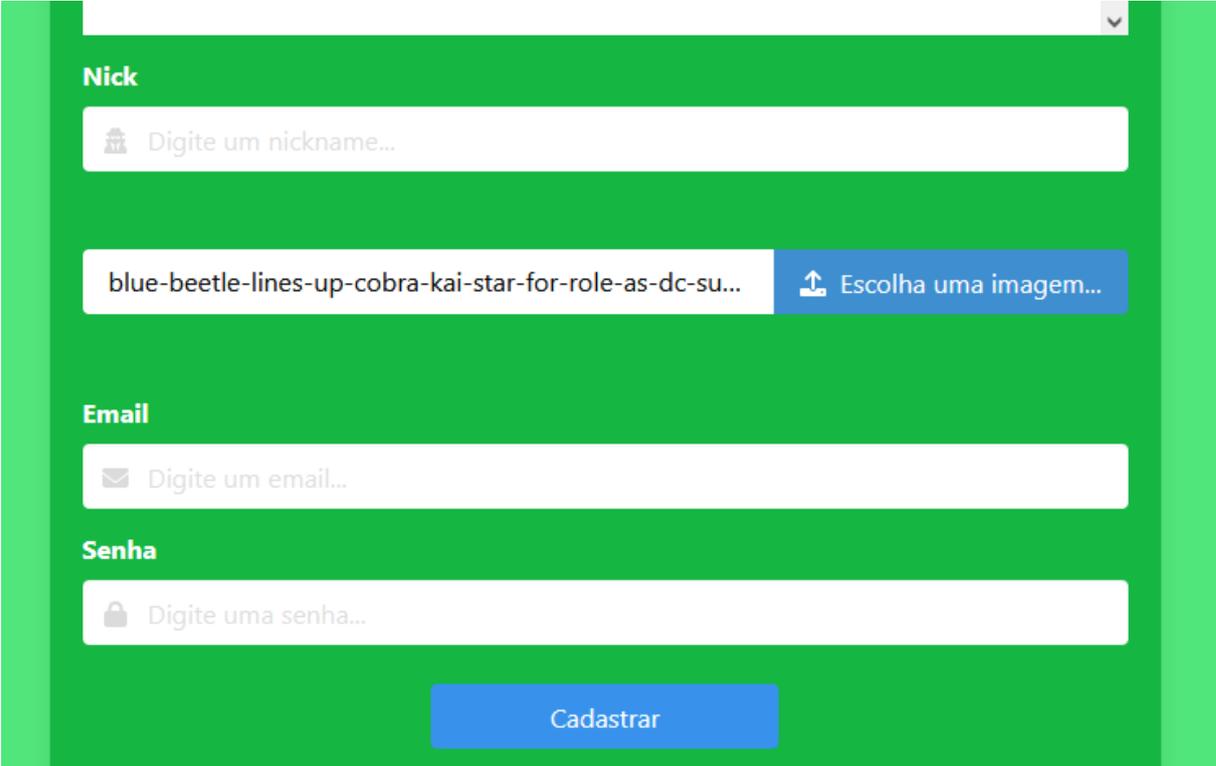
WIRTH, N. A brief history of software engineering. **IEEE Annals of the History of Computing**, IEEE, v. 30, n. 3, p. 32–39, 2008.

ZAMBON, P. S.; CHAGAS, C. J. R. Produção independente de jogos digitais: o desenvolvedor "lone wolf".

ĆURČIĆ, L. **Challenges of indie game making**. 2018. Disponível em: <<https://marbleit.rs/blog/challenges-of-indie-game-making/>>.

APÊNDICE A – IMAGEM DO FORMULÁRIO DE ENVIO DE IMAGEM NO CADASTRO DE USUÁRIO

Figura A.1 – Formulário de envio de imagem no cadastro de usuário



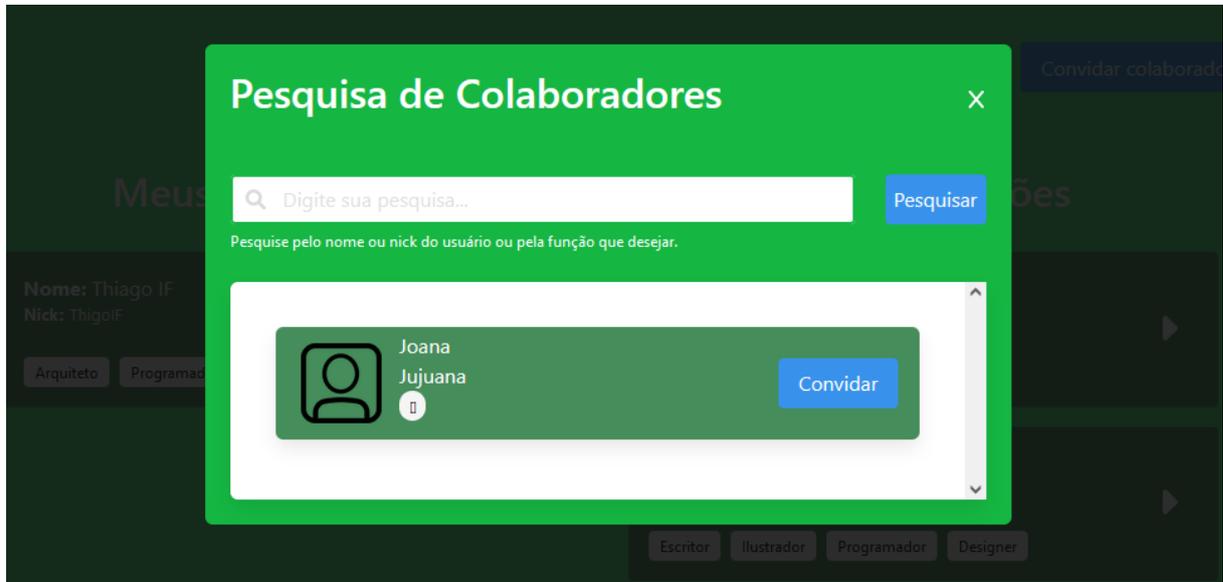
The image shows a registration form with a green background. It contains the following elements:

- Nick:** A text input field with a placeholder "Digite um nickname...". Below it is a preview of a long alphanumeric string: "blue-beetle-lines-up-cobra-kai-star-for-role-as-dc-su...". To the right of the preview is a blue button with an upload icon and the text "Escolha uma imagem...".
- Email:** A text input field with a placeholder "Digite um email..." and an envelope icon on the left.
- Senha:** A text input field with a placeholder "Digite uma senha..." and a lock icon on the left.
- Cadastrar:** A blue button at the bottom center with the text "Cadastrar".

Fonte: Elaborado pelo autor (2022)

APÊNDICE B – IMAGEM DA TELA DE PESQUISA E CONVITE DE COLABORADORES

Figura B.1 – Tela de pesquisa e convite de colaboradores



Fonte: Elaborado pelo autor (2022)

APÊNDICE C – IMAGEM DA TELA DA SALA DE REUNIÕES

Figura C.1 – Tela da sala de reuniões



Fonte: Elaborado pelo autor (2022)



Documento Digitalizado Restrito

Trabalho de conclusão de curso

Assunto: Trabalho de conclusão de curso
Assinado por: Thiago Almeida
Tipo do Documento: Projeto
Situação: Finalizado
Nível de Acesso: Restrito
Hipótese Legal: Informação Pessoal (Art. 31 da Lei no 12.527/2011)
Tipo do Conferência: Cópia Simples

Documento assinado eletronicamente por:

- **Thiago Yure Vieira de Almeida, ALUNO (202022010007) DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - CAJAZEIRAS**, em 25/09/2023 16:37:34.

Este documento foi armazenado no SUAP em 25/09/2023. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 953205
Código de Autenticação: 1455cc7001

