

CIÊNCIA DA COMPUTAÇÃO

ISABELLY KEISSA DE C CAETANO _____RA: N472FJ7

MARIA MARIANA CAGNONI FERREIRA __RA: N478GF0

SANDY SAYURI KATSUMATA ISHII _____RA: D89GJC4

TIAGO HENRIQUE SILVA _____RA: D984586

VICTOR CAPELARI _____RA: D8874J9

RELATÓRIO DE APS

TEMA DO PROJETO: “ESPECIFICAÇÃO DE REQUISITOS E MODELAGEM DE UM PRODUTO SOFTWARE”

CAMPINAS

Campus Swift

2022

ISABELLY KEISSA DE C CAETANO

MARIA MARIANA CAGNONI FERREIRA

SANDY SAYURI KATSUMATA ISHII

TIAGO HENRIQUE SILVA

VICTOR CAPELARI

RELATÓRIO DE APS

TEMA DO PROJETO: “ESPECIFICAÇÃO DE REQUISITOS E MODELAGEM DE UM PRODUTO SOFTWARE”

Relatório de projeto apresentado ao curso de Ciência da Computação da Universidade Paulista do campus Swift de Campinas a ser utilizado como diretriz para manufatura do trabalho do sétimo semestre do curso.

CAMPINAS

Campus Swift

2022

SUMÁRIO

•	OBJETIVOS DO TRABALHO	4
•	CONCEITOS GERAIS	6
•	DOCUMENTO DE REQUISITOS	11
•	MODELAGEM	14

4.1 Diagramas de Casos de Uso _____ **14**

4.2 Descrição dos Casos de Uso _____ **15**

4.3 Diagrama de Classes _____

• **CONCLUSÃO** _____

• **REFERÊNCIAS BIBLIOGRÁFICAS** _____

OBJETIVOS DO TRABALHO

Jogos de computador baseados em esporte atualmente são bem comuns hoje em dia e possuem um grande espaço no mercado, possuindo várias didáticas e estilos de esportes, podendo se variar entre simuladores ou a mescla entre ficção e realidade ficção.

Um dos jogos mais famosos é o futebol, e com ele surgiram diversos jogos, entre eles temos o Cartola FC, onde se é escalado um time de com os jogadores reais de diversos times de futebol brasileiro da serie A, e são disputadas partidas fictícias com outros times criados por outros jogadores.

Fonte:

<https://www.techtudo.com.br/noticias/2022/04/cartola-fc-2022-veja-mudancas-no-game-modo-express-e-dicas-para-jogar.ghml>

Outro esporte muito famoso entre os brasileiros é o vôlei, uma competição com times de 6 jogadores onde o objetivo é utilizar as mãos para pontuar fazendo a bola encontra no chão do time adversário.

Apesar de ser um estilo de jogo menos conhecido que o futebol, ainda é um jogo bem popular, porem apesar da fama que o estilo de esporte possui, atualmente não temos um jogo baseado em partidas de vôlei.

Nosso objetivo é a construção de um jogo inspirado no Cartola FC, no qual o objetivo é escalar seu próprio time de vôlei com diversas jogadoras de vôlei inscritas oficialmente na (nome do campeonato de vôlei perguntar para a mari). No qual após ter o seu time fechado será colocado em um campeonato possuindo um posição em um ranking onde dependendo do desempenho das jogadoras escaladas em seu time ira subir o descer na classificação, onde no final do campeonato dependendo de sua aposição no ranking recebera uma recompensa no qual será possível a compra de novas jogadores para serem escaladas em um novo time para um próximo campeonato.

Fonte: Autoral

CONCEITOS GERAIS

Requisitos de Software

No passado, era dito que requisitos são sinônimos de funcionalidade, ou seja, tudo o que o software deve fazer funcionalmente. No entanto, os requisitos de software são agora considerados mais do que simples funções. Além da funcionalidade, objetivos, propriedades, limitações, requisitos são os requisitos do contrato, padrão ou especificação do usuário que o sistema deve atender. De uma forma geral, os requisitos são muito importantes para atingir os objetivos.

Os requisitos tem objetivos como determinar e manter um acordo com o cliente e outros envolvidos sobre o que o sistema tem que fazer, tem que dar aos desenvolvedores, projetistas e testadores do sistema uma melhor compreensão dos requisitos. Definir parâmetros do sistema que definem o que deve ser instalado e o que não deve fazer parte do sistema, fornecer uma base para estimar o tempo e os custos de desenvolvimento do sistema e, finalmente, definir a interação do usuário com o sistema.

Existe dois tipos de requisitos: Requisitos Funcionais (RF) e Requisitos Não-Funcionais (RNF).

Requisitos Funcionais: representa o comportamento do sistema, suas exigências de desempenho para cada item, ou seja, os requisitos que definem a função do sistema. Eles são especialistas em projetos porque definem as tarefas que o sistema deve ter.

Requisitos Não-Funcionais: são condições relativos ao uso do aplicativo em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologia envolvida. E são divididos em três etapas: requisitos do produto final, requisitos organizacionais e requisitos externos.

Engenharia de Requisitos

Engenharia de Requisitos, também conhecido como RE, refere-se ao processo de definir, escrever e manter os requisitos para um projeto de engenharia específico dentro do ambiente de TI. É aqui que a oportunidade de encontrar a maneira certa de entender a solicitação de um cliente é útil.

É responsável por assegurar toda a informação disponível sobre o projeto, o que permite uma gestão eficiente das necessidades, tendo em conta a sua transformação dentro do plano de trabalho. Assim, todos os processos funcionam de forma rápida e realista.

A Engenharia de Requisitos contém etapas específicas que possibilitam o uso do software. Além disso, sua documentação requer algum tipo de cuidado ao longo do tempo, então a etapa do processo é estabelecida para melhorar a qualidade da integração.

Essas etapas do processo estão diretamente ligadas a:

Concepção: é uma etapa na qual são definidos os stakeholders, bem como suas diferentes perspectivas sobre os problemas e impactos gerados no sistema.

Elicitação: nesta etapa, as necessidades do usuário do programa devem ser levantadas nas ideias da categoria requerida, e, no contexto da necessidade.

Elaboração: na elaboração, é necessário que cada necessidade seja claramente definida e interpretada na linguagem natural, levando em consideração os modelos de ideias.

Negociação: nesta fase, é necessário identificar todos os conflitos entre necessidades para possibilitar a negociação de soluções específicas.

Especificação: a especificação trabalha para melhorar os requisitos do sistema que precisam atender às expectativas do cliente e do usuário

Validação: nesta etapa, é realizada uma confirmação da instalação do sistema.

Gerenciamento: interage com todo o ciclo de vida do produto.

UML

UML (Integrated Modeling Language) que significa Linguagem Unificada de Modelagem é um modelo padrão orientado a objetos. Ele combina três modos principais, BOOCH, OMT (Rumbaugh) e OOSE (Jacobson). Essa linguagem não nativa de terceira geração não é o caminho a seguir. Eles ajudam a visualizar o design e a interação entre os objetos. Ele permite que os desenvolvedores visualizem seus produtos de trabalho em gráficos de alta qualidade e é frequentemente usado para modelar programas de software.

O UML é uma linguagem de modelagem padrão para sistemas de modelagem semelhantes aos distribuídos. Ele usa um conjunto de técnicas de comentários gráficos para criar modelos de software visuais para programas em profundidade, combinando os melhores dados, negócios, técnicas de objetos e componente de modelagem. É uma linguagem de modelagem única, comum e amplamente utilizada.

Embora com UML seja possível representar software com modelos focados em objetos, não indica que tipo de trabalho deve ser feito, ou seja, não tem um processo que descreva como o trabalho deve ser desenvolvido. O objetivo é definir “o que é feito”, “como é feito”, “quando é feito” e “por que deve ser feito”. É necessário completar um dicionário de dados para descrever todas as organizações envolvidas, definindo assim os requisitos operacionais do software.

A UML tem diagramas (diagramas de modelos de componentes do sistema) que são utilizados em combinação, para capturar todas as ideias e recursos do sistema.

Os diagramas UML são estruturados por estrutura e comportamento.

Os diagramas estruturais são: o de classe, de objeto, de componentes, de implantação, de pacotes e o de estrutura.

Os diagramas comportamentais são: de Caso de Uso (Use Case), de Máquina de Estados, de Atividades e o de Interação que é dividido em Sequência, Geral interação, comunicação e tempo.

Fonte: <https://www.infoescola.com/engenharia-de-software/uml/>

RUP

O Processo Unificado da Rational conhecido como RUP (Rational Unified Process), é um processo de engenharia de software projetado para dar suporte ao desenvolvimento orientado a objetos, fornecendo uma abordagem sistemática para tirar proveito da UML. Desenvolvido pela Rational Software Corporation e adquirido em fevereiro de 2003 pela IBM.

O principal objetivo do RUP é atender às necessidades dos usuários, atestando a produção de software de alta qualidade que atenda ao cronograma e orçamento previsíveis. O RUP demonstra como o sistema irá estabelecendo, produzindo um modelo de projeto e, opcionalmente, um modelo de análise utilizado para garantir a durabilidade. Ele declara claramente quem é responsável por isso, como deve ser feito e quando deve ser feito, define todos os objetivos de desenvolvimento especificamente para alcançá-los. O RUP organiza o desenvolvimento de software em quatro fases, nas quais são abordadas questões sobre planejamento, coleta de demanda, análise, implementação, teste e implantação de software. Cada categoria desempenha um alcance da meta, que é distribuída entre alguns poucos profissionais, como Analista de sistema, Projetista, Projetista de testes, entre outros.

Fases do RUP

Fase de Concepção / Iniciação: Esta seção do RUP abrange as atividades de comunicação e planejamento do cliente.

Fase de Elaboração: Inclui um modelo de processo padrão.

Fase de Construção: Atualiza ou recebe componentes de software.

Fase de Transição: Inclui entrega de software ao usuário e fase de teste.

Fonte: <https://www.infoescola.com/engenharia-de-software/rup/>

DOCUMENTO DE REQUISITOS

Requisitos Específicos:

- Requisitos Funcionais:

Requisito Funcional 1: Deverá permitir ao usuário a possibilidade de escalar seu time de vôlei com suas jogadoras compradas pelo dinheiro gerado no jogo.

Requisito Funcional 2: Devera permitir ao usuário a possibilidade de compra de jogadoras de vôlei e a possibilidade de escala-las e gerar um time completo.

Requisito Funcional 3: Deverá ter a possibilidade de que caso o usuário não goste do time escalado, tenha a possibilidade de exclusão total do time para criação de um novo.

Requisito Funcionais 4: Deverá ter a possibilidade de que caso o Usuário queira ou precise fazer uma atualização em seu time seja possível fazer a alteração do mesmo.

Requisito Funcionais 5: Quando um usuário novo entrar no website, para poder entrar e jogar o jogo, será necessário realizar um cadastro com um e-mail válido, um nome de login e senha.

Requisito Funcionais 6: Quando o Usuário acessar o website para poder entrar e poder jogar, será necessário digitar seu e-mail e senha para poder entrar em sua conta cadastrada.

Requisito Funcionais 7: Quando o time escalado pelo usuário esta formado e é colocado em um campeonato onde é colocado em um ranking com vários times onde os mesmos terão uma pontuação gerada por um cálculo de cada jogadora escalada terá sua quantidade de pontos específica dependendo de suas habilidades descritas dentro do jogo.

Requisito Funcionais 8: Cada jogadora dentro do jogo terá habilidades específicas onde dependendo das qualidades delas ser realizado um cálculo de pontos onde no total terá um valor total da jogadora.

Requisito Funcional 9: Quando o time for escalado e colocado no ranking do campeonato, haverá um cálculo de posição do time dentro da tabela, o valor do time equivale ao total de pontos gerados pelas jogadoras de escaladas no time.

Requisito Funcional 10: Para atualizações do jogo caso seja necessária a inclusão de uma ou mais jogadora novas no banco de dados.

Requisito Funcional 11: Para atualizações do jogo caso seja necessário a exclusão de uma ou mais jogadoras no banco de dados.

Requisito Funcional 12: Caso o usuário não queria mais jogar o jogo e deseje excluir sua conta, terá a possibilidade de deletar completamente sua conta, removendo-o completamente do banco de dados.

Requisito Funcional 13: Caso o usuário queira ou precise realizar a alteração em sua conta, deverá ser possível a alteração do e-mail da conta para a realização do login.

Requisito Funcional 14: Caso o usuário queira ou precise realizar a alteração em sua conta, deverá ser possível a alteração de senha da conta para realizar o login.

Requisito Funcional 15: Após a finalização do campeonato, dependendo de sua posição no ranking haverá recompensa dentro do jogo, com esse valor do prêmio será possível a compra de novas jogadoras para ser escalada em seu próximo time.

Requisito Funcional 16: Deverá ser possível a inserção ou mudanças das regras dentro do jogo dependendo das mudanças e alterações dos regulamentos da FIVB.

- Requisitos não Funcionais:

Requisito não funcional 1: O website deverá realizar uma identificação do sistema e configurações do dispositivo que foi acessado e realizar um calculo para ver se o dispositivo atende os requisitos mínimos para poder acessar e jogar o jogo.

Requisito não funcional 2: O website deverá ter a possibilidade de funcionar e rodar em qualquer navegador atual e atualizado.

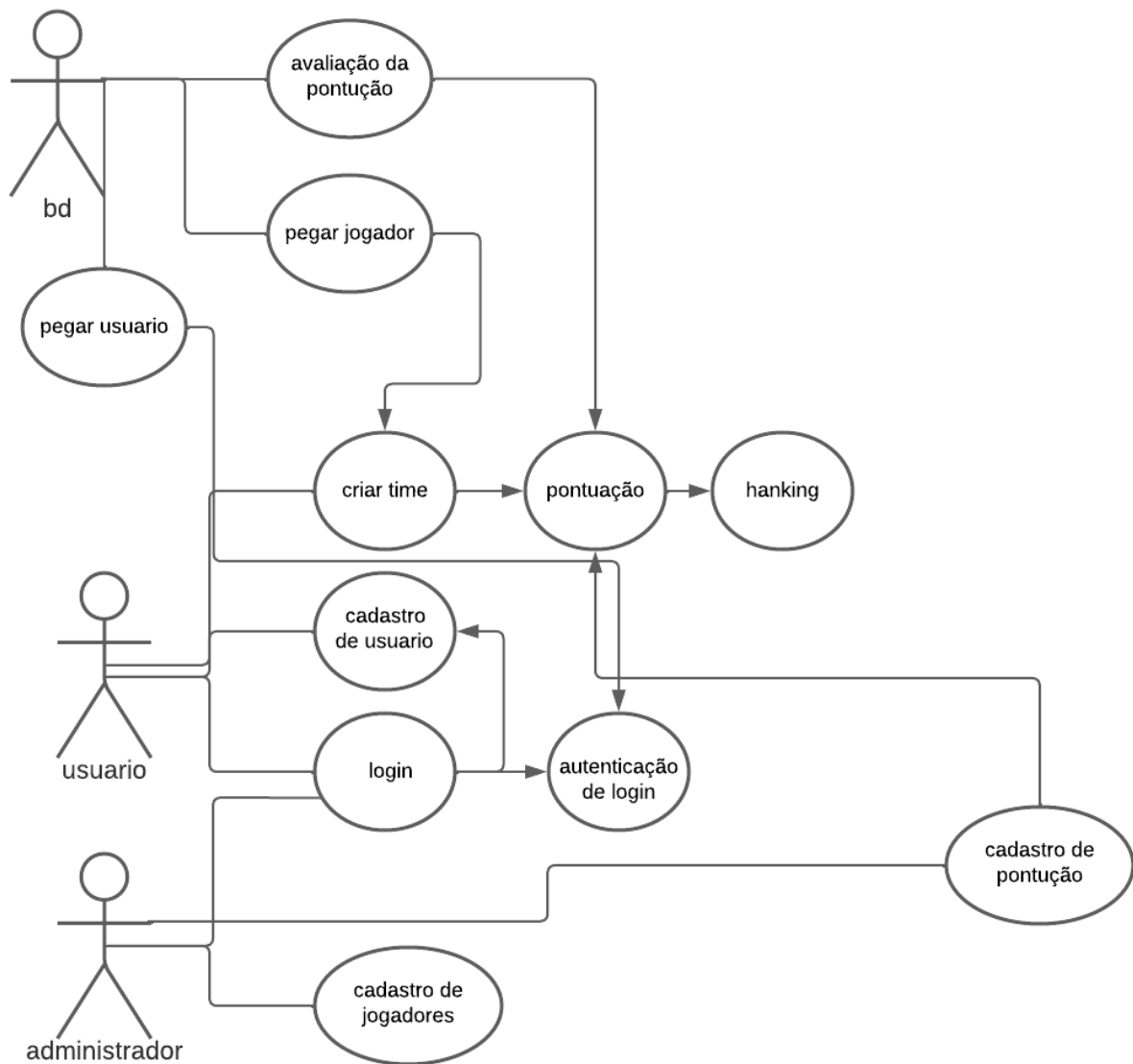
Requisito não funcional 3: O website deverá ter um acesso e alteração rápido ao banco de dados, assim tendo uma dinâmica rápida para entreter o jogado.

Requisito não funcional 4: O website devera respeitar o direito de imagem das jogadoras anexadas ao jogo.

Requisito não funcional 5: O site devera ter uma segurança alta, dando confiabilidade aos usuários, necessário utilizar uma criptografia dentro de seu bando e de sua programação, assim evitando o vazamento de dados dos jogadores.

MODELAGEM

Diagramas de Casos de Uso



Descrição dos Casos de Uso

Nome do caso de uso:	Login
Ator principal:	usuário
Ator Secundário:	administrador ,bd
Descrição:	quando o usuário for entrar na plataforma
Pré-Condições:	NA
Pós-Condições:	ele terá acesso a pontuação dos demais usuários e também vai poder escalar o time
Fluxo Principal	

Ações do Ator: 1.Colocar usuario e senha	Ações do Sistema: 2. validar para ver se o usuário tem.
	3. O sistema executa o sub-fluxo correspondente ao tipo de resposta dos sistemas (se não tiver ou se tiver)
Sub-Fluxo se tiver	
Ações do Ator: 5. ele pode ir onde quiser no jogo	Ações do sistema: 4. libera a entrada o usuário e abre a home do jogo
Sub-Fluxo se não tiver	
Ações do Ator: 5. o usuário terá que colocar nome,email, criar uma senha e colocar a data de nascimento . 7. o usuário entra no jogo	Ações do sistema: 4. irá abrir uma tela de cadastro 6. se estiver tudo certo ele os dados são salvos no bd

	realização da operação.
Fluxos de Exceção:	E1. O sistema não consegue encontrar os dados dos usuários. OSD exibir mensagem de erro. E2. O sistema reconhece os dados como (email, senha , data de nascimento)de um usuário. OSD exibir mensagem de erro. E3. O sistema reconhece que o usuário tem menos de 18 anos. OSD exibir mensagem de erro.
Fluxos alternativos:	A1. OSD permite ao usuário cancelar se ele não clicar no botão me cadastrar .
Regras de negócios:	RN1: Os clientes não podem apresentar o mesmo email. RN2: Não pode excluir um usuário que já tenha uma pontuação relacionada.
Validações:	V1: O campo nome pode ter no máximo 75 caracteres, não podendo ser vazio. V2: O campo email pode ter no máximo 150 caracteres, não podendo ser vazio. V3: a senha tem que ter pelo menos 8 caracteres onde 1 tem que ser caracter especial , 1 letra maiúscula e pelo menos 1 número V4 data de nascimento tem que estar em formato dd/mm/aa

Nome do caso de uso:	Criar time
Ator principal:	Usuário
Ator Secundário:	admin e bd

Descrição:	escalação do time para começar o jogo
Pré-Condições:	tem que ter feito o login
Pós-Condições:	Dados das pontuações tem que mantidos no sistema, permitindo associação com outros jogadores para fazer o ranking
Fluxo Principal	
Ações do Ator: 1. OSD permitir usuário clicar em uma posição 3. OSD permite ao usuário escolher a jogadora até que todos os campos sejam preenchidos. 4.O usuário clica no botão de salvar	Ações do Sistema: 2. OSD apresentar ao usuário as jogadoras que jogam nessa posição com(os seus nomes , tamanhos e preço de contratação) . 5. OSD o sistema soma todos os preços das jogadoras e subtrai do quanto de ” dinheiro” o usuário tem . 6. O SD soma a pontuação e salva ela no bd depois ordena ela e coloca no ranking

Fluxos de Exceção:	E1.O usuário aperta no botão de salvar sem ter 1 jogadora para cada uma das posições. O SD vai exibir mensagem de erro. E2.Se quando o sistema subtrair não der menos que o "dinheiro ” do usuário OSD exibir mensagem de erro.
Fluxos alternativos:	A1.Se o usuário não apertar no botão de salvar todos os danos vão ser perdidos.

Nome do caso de uso:	Cadastro de jogadores
Ator principal:	administrador
Ator Secundário:	bd
Descrição:	Cadastrar um jogador
Pré-Condições:	login
Pós-Condições:	ele terá poderá incluir , excluir e editar um jogador
Fluxo Principal	
Ações do Ator: 1.escolher uma das opções	Ações do Sistema:
	2. ir para a página escolhida
Sub-Fluxo incluir	

Ações do Ator: 3. clicar no botão de adicionar jogadora 4.clica nos campos correspondentes às informações(nome, altura, falhas em campo e acertos) colocar as informações e clica em salvar	Ações do sistema: 5. OSD verifica se todos os campos estão preenchidos 6. OSD verifica se todos dos dados então no formato desejado depois manda para o bd
Sub-Fluxo excluir	
Ações do Ator: 3. ele procura a jogadora 5. clica na jogadora 7. clica em excluir 9. aperta “sim”	Ações do sistema: 4. O SD procura todos com o nome parecido com aquele digitado 6. O SD abre o perfil da jogadora 8. O SD mostra a mensagem se deseja mesmo excluir 10.O SD exclui o jogador
Sub-Fluxo editar	
Ações do Ator: 3.procurar jogadora 5. clica na jogadora 7. clica em editar 9.editar e clica no botão de salvar	Ações do sistema: 4. O SD procura todos com o nome parecido com aquele digitado 6. O SD abre o perfil da jogadora 8. O SD deixa todos os campos virarem campos editáveis 10. O SD verifica se os campos estão nos formatos certos 11. O SD salva o bd

	realização da operação.
Fluxos de Exceção:	E1. O sistema não consegue encontrar os dados dos jogadores no bd. exibir mensagem de erro. E2. O sistema tem que dar erro de os formatos dos dados estiverem errados. OSD exibir mensagem de erro. E3.Não pode ter duplicidade de jogadores se não o sistema tem que exibir mensagem de erro.
Fluxos alternativos:	A1. O Sistema permite ao administrador cancelar se ele não clicar no botão de salvar .
Regras de negócios:	RN1: Os jogadores não podem apresentar o mesmo id . RN2: Não pode atribuir um jogador excluído um usuário RN3:A exclusão tem que ser lógica não física
Validações:	V1: O campo nome pode ter no máximo 75 caracteres, não podendo ser vazio. V2: O campo id pode ter no máximo 20 caracteres, não podendo ser vazio. V3: a tamanho tem que ser float V4 : erros e acertos tem que ser int

Nome do caso de uso:	Pontuação do jogador
Ator principal:	admin
Ator Secundário:	bd
Descrição:	pontuação para o jogador

Pré-Condições:	tem que ter feito o login
Pós-Condições:	Dados das pontuações tem que mantidos no sistema, permitindo associação com outros jogadores para fazer o ranking
Fluxo Principal	
Ações do Ator: 1.escreva o nome da jogadora 3. clica na jogadora 5.clica no botão adicionar pontuação 7.é colocados os acertos e erros 8.clicar em salvar	Ações do Sistema: 2. OSD procura a jogadora 4. abre o perfil da jogadora. 6. abre um pop up com um caixa de texto escrito acertos e erros 9. analisa se tudo oq está escrito está na formatação correta 10. salva no banco de dados 11.tira uma média baseado no historicos de acertos e erros do jogador fazendo a pontuação 12. atualiza a pontuação do jogador e o preço dele também

Fluxos de Exceção:	E1.Se o adim procurar alguma jogadora que não está na base de dados . O SD vai exibir mensagem de erro.
Fluxos alternativos:	A1.Se o usuário não apertar no botão de salvar todos os danos vão ser perdidos.
Validações:	acertos tem que ser int erros tem que ser int

CONCLUSÃO

A ideia do trabalho é criar um jogo com interface gráfica para as jogadoras de vôlei feminino uma vez que quando a ideia foi concebida não existia um jogo desse ,por ter um público muito específico e não ser tão famoso quanto as outras modalidades. Mas o público de vôlei feminino é muito fiel e tão apaixonados quanto outros times de outras modalidades .

REFERÊNCIAS BIBLIOGRÁFICAS

Higor. Introdução a Requisitos de Software: Veja neste artigo o que são requisitos de software, quais são as suas classificações e como os requisitos estão relacionados com o ciclo de vida de um projeto de desenvolvimento de software.. Devmedia, 2013. Disponível em: <https://www.devmedia.com.br/introducao-a-requisitos-de-software/29580>. Acesso em: 25 abr. 2022.

MARTINEZ, Marina. RUP. Infoescola. Disponível em: <https://www.infoescola.com/engenharia-de-software/rup/>. Acesso em: 26 abr. 2022.

MARTINEZ, Marina. UML. Infoescola, 202. Disponível em: <https://www.infoescola.com/engenharia-de-software/uml/>. Acesso em: 26 abr. 2022.

Rodrigo. Introdução à Engenharia de Requisitos: Neste artigo, faremos uma introdução à Engenharia de Requisitos, atividade base para as demais tarefas associadas ao desenvolvimento de software.. Devmedia, 2008. Disponível em: <https://www.devmedia.com.br/introducao-a-engenharia-de-requisitos/8034>. Acesso em: 28 abr. 2022.

CONCEITO: Requisitos Não-Funcionais. [S. l.]. Disponível em: https://www.trt9.jus.br/pds/pdstrt9/guidances/concepts/supporting_requirements_B2C4D610.html#:~:text=Requisitos%20n%C3%A3o%2Dfuncionais%20s%C3%A3o%20os,disponibilidade%2C%20manutenibilidade%20e%20tecnologias%20envolvidas. Acesso em: 28 abr. 2022.

NOLETO, Cairo. Engenharia de requisitos: quais as etapas e como funciona?. Blog.betrybe, 29 nov. 2021. Disponível em: <https://blog.betrybe.com/tecnologia/engenharia-de-requisitos-tudo-sobre/>. Acesso em: 28 abr. 2022.