

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DE SÃO PAULO**

Campus São João da Boa Vista

Trabalho Final de Curso

4º ano – Curso Técnico em Informática

Prof. Breno Lisi Romano

**Especificação dos Casos de Uso do Módulo de Treinos do Projeto Mais
Saúde São João**

Aluno: Lucas Rubini Silva

Prontuário: 1520091

São João da Boa Vista – SP

2018

Resumo

Este documento tem o papel de introduzir e descrever ao leitor as etapas de análise referentes à elaboração e documentação dos casos de uso do Módulo de Treinos do projeto Mais Saúde São João, através da explanação de processos e métodos referentes ao levantamento de requisitos, à concepção, diagramação e prototipagem dos casos de uso, e às dinâmicas ambientais realizadas pelos envolvidos na análise.

Sumário

1	Introdução	7
1.1	Objetivo Geral	11
1.2	Objetivos Específicos	11
2	Desenvolvimento	12
2.1	Levantamento Bibliográfico	12
2.1.1	Ciclo de vida do desenvolvimento de software	12
2.1.2	Análise de Software (Papel do Analista)	14
2.1.3	Requisitos	14
2.1.4	Linguagem de Modelagem Unificada (UML)	16
2.1.5	Diagrama de casos de uso	17
2.1.6	Documentação de Casos de Uso	24
2.2	Etapas para o Desenvolvimento da Pesquisa	27
2.2.1	Entendimento dos métodos iniciais de análise para o posterior levantamento de requisitos e viabilidades do Módulo de Treinos	27
2.2.2	Concepção do processo de conversão dos requisitos do sistema nas funcionalidades detalhadas que constituem os Casos de Uso	31
2.2.3	Explicação da prototipagem prática de um sistema e seus impactos no ambiente de trabalho cotidiano, explicitando a relação dinâmica entre analista e desenvolvedor	46
3	Conclusões e Recomendações	49
4	Referências Bibliográficas	51

Lista de Figuras

Figura 1. Esquema representando os Subsistemas do Projeto Mais Saúde São João [4].....	8
Figura 2. Representação da divisão dos módulos do Projeto Mais Saúde São João e seus respectivos subsistemas [5].....	9
Figura 3. Representação visual do produto final de funcionalidades do Módulo 04: Treinos do projeto Mais Saúde São João [7].....	10
Figura 4. Representação da estrutura de etapas e processos do Modelo em Cascata utilizado em ciclos de vida do desenvolvimento de software [11]	13
Figura 5. Esquema representando os graves problemas causados por uma especificação falha de requisitos [14].....	15
Figura 6. Exemplos de requisitos funcionais e requisitos não funcionais de um sistema hipotético [14].....	16
Figura 7. Representação de um simples cenário de um sistema e seus eventos [16].....	17
Figura 8. Representação visual do ator dos casos de uso segundo a UML [16]	18
Figura 9. Representação visual do caso de uso segundo a UML [16]	19
Figura 10. Exemplo de relacionamento de associação entre ator e caso de uso [18].....	20
Figura 11. Exemplo de relacionamento de inclusão entre dois casos de uso [18]	21
Figura 12. Exemplo do relacionamento de extensão entre dois casos de uso [18] [19].....	22
Figura 13. Exemplo de relacionamento de generalização entre dois casos de uso específicos e um caso de uso geral [18] [19]	23
Figura 14. Exemplo de relacionamento de generalização entre um ator ascendente (pai) e um ator descendente (filho) [20]	24
Figura 15. Exemplo de protótipo de interface de um sistema de gestão [23]	26
Figura 16. Descrição da concepção inicial das funcionalidades do Módulo de Treinos [5]..	28
Figura 17. Esquema de representação do grupo de funcionalidades do módulo de Treinos de acordo com a interação entre usuário comum e educador físico [5].....	29
Figura 18. Principais aspectos do processo de identificar e caracterizar os atores dos casos de uso [25]	34
Figura 19. Esquema indicando a utilização do relacionamento de associação dentro da dinâmica de agendamento e gerenciamento de consultas [25]	36
Figura 20. Esquema indicando a introdução do relacionamento de extensão dentro da dinâmica de agendamento e gerenciamento de consultas [25]	37

Figura 21. Esquema indicando a introdução do relacionamento de inclusão dentro da dinâmica de agendamento e gerenciamento de consultas [25]	38
Figura 22. Representação do diagrama de casos de uso referentes ao agendamento e gerenciamento de consultas, segundo os padrões da UML [25]	39
Figura 23. Produto final do diagrama de casos de uso [25]	40

Lista de Tabelas

Tabela 1. Cabeçalho de contextualização do caso de uso “Visualizar consultas do usuário” [25]	41
Tabela 2. Documentação do fluxo principal do caso de uso “Visualizar consultas do usuário” [25]	42
Tabela 3. Documentação dos fluxos alternativos do caso de uso “Visualizar consultas do usuário” [25]	42
Tabela 4. Estrutura de documentação do caso de uso “Marcar consulta” [25]	44
Tabela 5. Protótipo de interface homem-máquina da documentação do caso de uso “Visualizar consultas do usuário” [25]	46
Tabela 6. Protótipo de interface homem-máquina da documentação do caso de uso “Marcar consulta” [25]	48

1 Introdução

Fundado no ano de 1909 com o nome de Escola de Aprendizes Artífices, o Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – IFSP – é uma autarquia federal de ensino, que oferece cursos com abrangência em diversos campos, como licenciatura, formação inicial e continuada, tecnologias, engenharias e pós-graduação. O IFSP é organizado em diversos câmpus e possui mais de 40 mil alunos matriculados nas 36 unidades distribuídas pelo estado de São Paulo [1].

Uma destas unidades institucionais encontra-se na cidade de São João da Boa Vista, município localizado no interior do estado de São Paulo e cuja população é estimada em 90.089 habitantes, segundo o IBGE [2]. O câmpus de São João da Boa Vista deu início às suas atividades no mês de janeiro de 2007, e no ano de 2018 já oferece cursos técnicos integrados ao ensino médio ou concomitantes e cursos de graduação, tecnologia, licenciatura, baixarelado e pós-graduação [1].

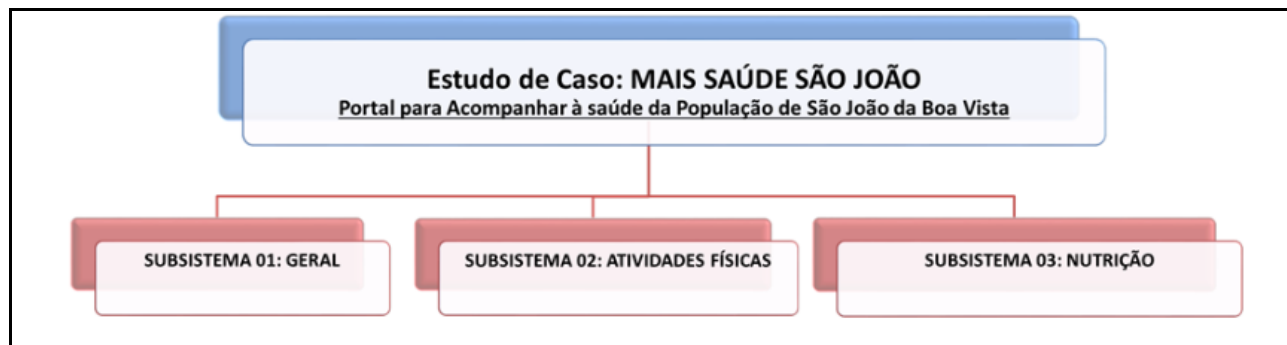
Um dos cursos mais procurados, o Curso Técnico Integrado em Informática capacita o aluno do ensino médio nas matérias curriculares tradicionais e, simultaneamente, lhe provê o conhecimento das modalidades técnicas da área de Informática. No quarto e último ano do curso, os alunos contêm em sua grade curricular a disciplina de Prática e Desenvolvimento de Sistemas, onde são introduzidos à vivência de um cotidiano baseado no trabalho coordenado em grupo. A disciplina tem como principal objetivo capacitar aos alunos os conceitos de projetos de software, de modo que os alunos aprendam a desenvolver softwares profissionalmente, de acordo com as necessidades atuais de mercado [3].

No ano de 2018, os alunos do 4º ano do Integrado em Informática tiveram a ideia de desenvolver um sistema ao qual atribuíram o nome de Mais Saúde São João, que tem como objetivo atender à carência da população de São João da Boa Vista por um serviço de saúde e nutrição de qualidade que seja eficiente, gratuito, e que aproxime os habitantes do município dos profissionais capacitados da área [4].

Tendo em vista o assunto do projeto e as diversas funcionalidades que possivelmente seriam implantadas, o planejamento da disciplina de Prática e Desenvolvimento de Sistemas para 2018 visava, pela primeira vez na história do câmpus, integrar as duas turmas do último ano do curso, 4º A e B, para atender à demanda de um projeto de grande porte e que apresentaria a necessidade de um grupo de quase 50 alunos trabalhando de maneira conjunta e organizada, com planejamento e prazos de entrega de atividades definidos, para que o resultado final se mostrasse satisfatório, atendendo não só aos objetivos da disciplina, mas também conferindo aos alunos experiências significativas para se inserir no mercado de trabalho [4].

Com a grande escala do projeto em mente, o seu processo de desenvolvimento em todas as fases, desde as análises e documentações iniciais até a própria programação, foi dividido em três subsistemas, o Subsistema 01 Geral, o Subsistema 02, de Atividades Físicas, e o Subsistema 03, de Nutrição, como mostra a Figura 1 [4].

Figura 1. Esquema representando os Subsistemas do Projeto Mais Saúde São João [4]



Cada um dos subsistemas corresponde a um conjunto de funcionalidades específicas a um determinado assunto, como, por exemplo, o Subsistema 02 engloba um grupo de ferramentas relacionadas a gestão de atividades físicas, interações entre educador físico e usuário, entre outras [4].

Além da divisão entre subsistemas, foi realizada outra hierarquia de divisões, com o intuito de focalizar os alunos em diversos grupos. Esta hierarquia foi realizada dividindo as múltiplas funcionalidades dos subsistemas em pequenos grupos, estes que por sua vez seriam distribuídos em módulos. Ao fim do processo, a distribuição de funcionalidades e requisitos, que está representada na Figura 2 abaixo, ficou caracterizada em uma divisão entre 9 módulos, sendo os módulos 01 e 02 pertencentes ao Subsistema 01 Geral, os módulos 03, 04, 05 e 06 pertencentes ao Subsistema 02, e os módulos 07, 08 e 09 caracterizando o Subsistema 03 [4].

Figura 2. Representação da divisão dos módulos do Projeto Mais Saúde São João e seus respectivos subsistemas [5]

SUBSISTEMAS	MÓD. 1 - USUÁRIOS	MÓD. 2 - REDE SOCIAL	MÓD. 3 - CHECKUPS	MÓD. 4 - TREINOS
SUBSISTEMA 1 USUÁRIOS	<ul style="list-style-type: none"> ESPECIFICAÇÃO DE CADASTRO PESSOAL LOGIN E SAÍDA DA REDE "SOBRE" DO PROJETO E DA INSTITUIÇÃO IFSP GESTÃO DE ADMINISTRADORES GESTÃO DE PATOLOGIAS E MEDICAÇÕES VISUALIZAÇÃO DE PATOLOGIAS E MEDICAÇÕES EDIÇÃO DE DADOS PESSOAIS DOS 04 PERFS GERENCIAMENTO DO CONTROLE DE ACESSO RECUPERAÇÃO DE SENHA PÁGINAS DE RECEPÇÃO FALE CONOSCO COM DÚVIDAS E FEEDBACK 	<ul style="list-style-type: none"> GESTÃO DE MENSAGENS COM FOTOS OPÇÃO "CURTIR" PARA MENSAGENS PUBLICADAS GESTÃO DE COMENTÁRIOS NAS MENSAGENS PUBLICADAS OPÇÃO "DENUNCIAR" PARA MENSAGENS PUBLICADAS PUNIÇÃO A USUÁRIO DENUNCIADO GESTÃO DE ORIENTAÇÕES NUTRICIONAIS E ATIVIDADES FÍSICAS GESTÃO DE DICAS NUTRICIONAIS E ATIVIDADES FÍSICAS NOTIFICAÇÕES DE NOVAS POSTAGENS 	<ul style="list-style-type: none"> GESTÃO DE TESTES RELACIONADOS A SAÚDE DO USUÁRIO: <ul style="list-style-type: none"> ÍNDICE DE MASSA CORPÓREA; TAXA DE GORDURA; BATIMENTOS CARDÍACOS; SATURACÃO DE OXIGÊNIO NO SANGUE (SPO2); PRESSÃO SANGÜÍNEA; VISÃO; CAPACIDADE PULMONAR E TAXA RESPIRATÓRIA. CLASSIFICAÇÃO DOS RESULTADOS DOS TESTES DETERMINAÇÃO APROXIMADA DA SAÚDE DO USUÁRIO GESTÃO DE RELATÓRIOS TABULARES E GRÁFICOS DOS RESULTADOS DOS TESTES 	<ul style="list-style-type: none"> GESTÃO DA INTERAÇÃO USUÁRIO - EDUCADOR FÍSICO AGENDAMENTO DE CONSULTAS PRESENCIAIS GESTÃO DE DADOS REFERENTES À CONSULTA GESTÃO DE FICHAS DE TREINAMENTO GESTÃO DE MEDIDAS CORPORAIS E DOBRAS CUTÂNEAS ENVIO DE NOVAS FICHAS DE TREINAMENTOS VIA E-MAIL GESTÃO DE SUPLEMENTOS ALIMENTARES
MÓD. 5 - RESULTADOS DOS TREINAMENTOS	MÓD. 6 - FERRAMENTAS ESPORTIVAS	MÓD. 7 - PLANO ALIMENTAR/ CARDÁPIO	MÓD. 8 - DIÁRIO DE BORDO NUTRICIONAL	MÓD. 9 - FERRAMENTAS NUTRICIONAIS
<ul style="list-style-type: none"> GESTÃO DE DESEMPENHOS DAS FICHAS DE TREINAMENTO GESTÃO DE DESEMPENHOS DE PROGRAMAS DE TREINAMENTO PRONTOS GESTÃO DE HISTÓRICO DE DESEMPENHO DE TREINAMENTO GESTÃO DE GASTO CALÓRICO DIÁRIO RELATÓRIOS TABULARES E GRÁFICOS GASTO CALÓRICO VS CONSUMO CALÓRICO RELATÓRIOS DE FADIGA MUSCULAR POR GRUPO MUSCULAR RELATÓRIOS TABULARES E GRÁFICOS DE DOBRAS CUTÂNEAS 	<ul style="list-style-type: none"> GUIA DE EXERCÍCIOS FÍSICOS PESQUISA E VISUALIZAÇÃO GUIA DE SUPLEMENTOS ALIMENTARES GESTÃO DE UNIDADES DE MEDIDA DE ATIVIDADES FÍSICAS GESTÃO DE TIPOS DE EXERCÍCIOS FÍSICOS GESTÃO DE PROGRAMAS PRONTOS DE TREINAMENTO GESTÃO DE TESTES PSICOLÓGICOS GESTÃO DE LEMBRETES PARA AVISO DE CADASTRO DE RESULTADOS DE TREINAMENTO 	<ul style="list-style-type: none"> GESTÃO DA INTERAÇÃO USUÁRIO - NUTRICIONISTA AGENDAMENTO DE CONSULTAS PRESENCIAIS GESTÃO DE DADOS REFERENTES À CONSULTA GESTÃO DE CALORIAS DIÁRIAS NECESSÁRIAS GESTÃO DE CARDÁPIOS PERIÓDICOS INDIVIDUAIS RELATÓRIOS TABULARES E GRÁFICOS DE CALORIAS DOS CARDÁPIOS ENVIO DE NOVOS CARDÁPIOS AO USUÁRIO VIA E-MAIL 	<ul style="list-style-type: none"> CÁLCULO DO PESO IDEAL DO USUÁRIO GESTÃO DE DIÁRIO DE BORDO NUTRICIONAL INDIVIDUALIZADO GESTÃO DE ALIMENTOS FAVORITOS RELATÓRIOS TABULARES E GRÁFICOS CALORIAS CONSUMIDAS VS CALORIAS RESTANTES (DIÁRIO) RELATÓRIOS TABULARES E GRÁFICOS DE CALORIAS POR REFEIÇÃO E NUTRIENTES (DIÁRIO) HISTÓRICO TABULAR DE NUTRIENTES E CALORIAS ATUALIZAÇÃO DE PESO DO USUÁRIO HISTÓRICO POR GRÁFICO DE PESO 	<ul style="list-style-type: none"> PESQUISA CALÓRICA DE ALIMENTOS GESTÃO DE UNIDADES DE MEDIDA DOS ALIMENTOS GESTÃO DE TIPOS DE ALIMENTOS GESTÃO DE INFORMAÇÃO NUTRICIONAL DOS ALIMENTOS GESTÃO DE "MEUS ALIMENTOS" GESTÃO DE RECEITAS E SEU COMPARTILHAMENTO AUTORIZAÇÃO PARA PUBLICAR RECEITAS GESTÃO DE LEMBRETES PARA AVISO DE CADASTRO DE DIÁRIO DE BORDO

Cada um dos módulos representados acima, de acordo com suas potenciais funcionalidades, define a abordagem de um tema específico do projeto. Para realizar a divisão dos quase 50 alunos entre módulos, a estrutura padronizada de funções dentro do módulo seria composta por um Analista de Sistemas – e testador em um segundo momento – que seria responsável por encabeçar as atividades de documentação, análise e testes de protótipos de desenvolvimento, uma dupla de Analistas de Banco de Dados, responsável pela criação, gerenciamento e integração do banco de dados do módulo e projeto, e três Desenvolvedores de *front-end* e *back-end* – para eventuais excepcionalidades, alguns módulos contam com apenas dois Desenvolvedores – responsáveis pelo desenvolvimento de protótipos, *templates* e funcionalidades, seja seus aspectos visuais ou comportamentais [6].

Algumas das necessidades funcionais do sistema a ser desenvolvido neste projeto estão compreendidas por um grupo de ferramentas e funcionalidades do tema de treinos e atividades físicas, e entre os módulos do projeto Mais Saúde São João, o módulo designado a ser responsável por este conjunto foi o Módulo 04: Treinos. As modalidades a serem desenvolvidas dentro escopo deste módulo estão descritas abaixo na Figura 3 [7].

Figura 3. Representação visual do produto final de funcionalidades do Módulo 04: Treinos do projeto Mais Saúde São João [7]



Acerca do tema deste módulo, suas funcionalidades a serem desenvolvidas estão estritamente ligadas à relação entre o usuário da população e o educador físico, e a mecanismos relacionados à gestão e processos de uma rotina consistente de acompanhamento profissional e exercícios físicos, e para que seja possível o desenvolvimento destas ferramentas, é necessário passar por um processo detalhado e fundamental de análise [7].

Baseando-se em um ambiente e modelo de processo de desenvolvimento de software semelhante àquele utilizado nas empresas, os principais objetivos da análise de sistemas – que

envolve diversos processos e técnicas – são utilizar métodos para a redução de custos e eventuais problemas na prática do desenvolvimento, aumentar a qualidade e confiabilidade do produto final e sua vantagem competitiva, e aumentar a produtividade dos desenvolvedores, que irão concentrar seus esforços em sua principal função de desenvolvimento do sistema, evitando o descumprimento de prazos e orçamentos. [8]

1.1 Objetivo Geral

Entre as atividades e técnicas de análise utilizadas no projeto, surge a necessidade de redigir um processo que permita às partes envolvidas no desenvolvimento do sistema, o mapeamento do escopo do sistema, a representação visual de suas funcionalidades e seu gerenciamento [9]. Apresentada a situação em questão, a especificação dos Casos de Uso tem um papel vital no processo de identificação e modelagem do sistema. Sendo assim, o Módulo 04: Treinos elaborou um documento de casos de uso utilizado para documentar a diagramação dos casos de uso e o detalhamento de suas funcionalidades.

Portanto, o objetivo geral deste trabalho compreende e integra a especificação de todos os processos, etapas e adaptações que englobam a elaboração dos Diagramas de Caso de Uso do Módulo 04: Treinos do Projeto Mais Saúde São João, e sua posterior documentação.

1.2 Objetivos Específicos

Além do objetivo geral citado acima, é necessário evidenciar com minúcias os objetivos específicos deste documento, que correspondem a cada uma das etapas de desenvolvimento dos casos de uso, tais como seus processos de execução. Seus objetivos específicos compreendem:

- o entendimento dos métodos iniciais de análise para o posterior levantamento de requisitos e viabilidades do sistema;
- a concepção do processo de conversão dos requisitos do sistema nas funcionalidades detalhadas que constituem os Casos de Uso, assim como a elaboração da dinâmica de suas coexistências dentro de um modelo visual de sistema representado em diagramas e fluxos de interação com o usuário;
- a explanação da prototipagem prática de um sistema e seus impactos no ambiente de trabalho cotidiano, explicitando a relação dinâmica entre analista e desenvolvedor;

2 Desenvolvimento

2.1 Levantamento Bibliográfico

Anterior à elaboração dos processos descritivos em relação às etapas de desenvolvimento deste trabalho, é necessário a realização de um levantamento bibliográfico que tenha como objetivo introduzir ao leitor os conceitos, princípios e modalidades do conhecimento relevantes ao entendimento do conteúdo específico que será abordado ao longo das etapas para o desenvolvimento da pesquisa.

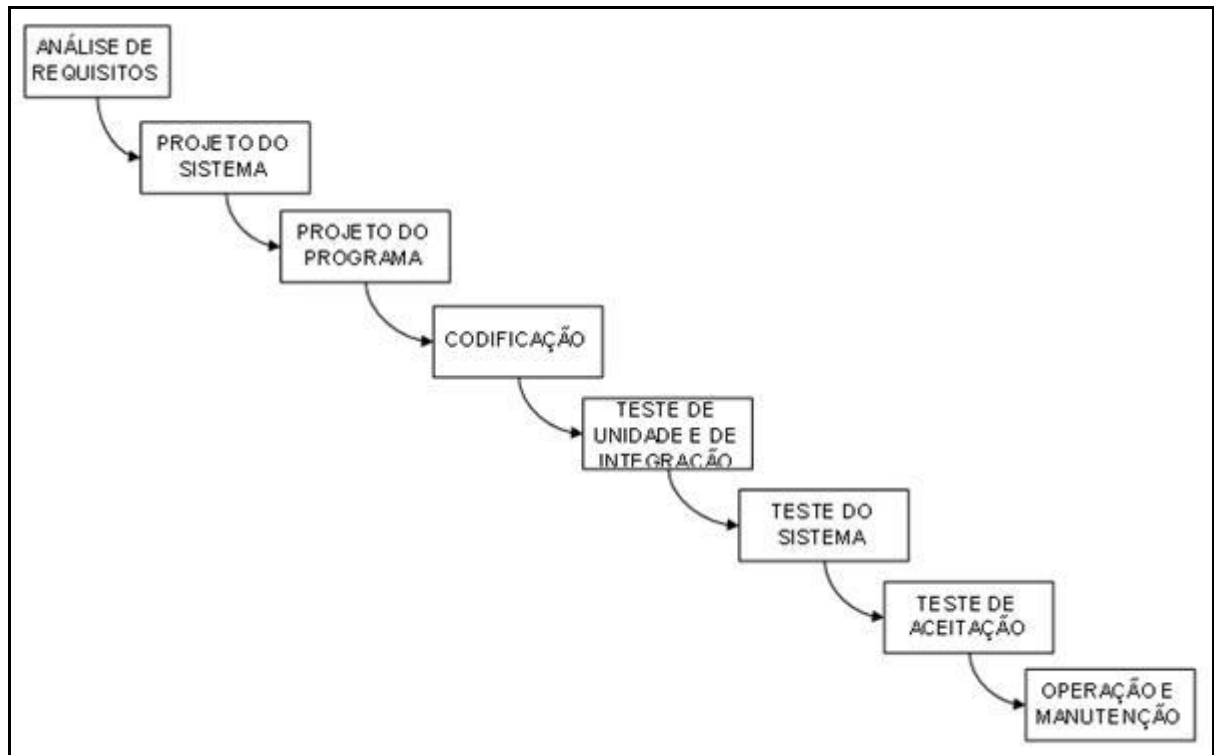
2.1.1 Ciclo de vida do desenvolvimento de software

Descrito na norma NBR ISO/IEC 12207:1998, o ciclo de vida do desenvolvimento de software é uma estrutura de processos, atividades e tarefas que abrangem a vida de um sistema, desde suas etapas iniciais de concepção até sua implantação e manutenção. Sua principal função é ser aplicado como um esquema que indique e descreva de forma coordenada e sustentável as fases, prazos e responsabilidades de cada envolvido no processo de desenvolvimento de software, alinhando o planejamento dos estágios do sistema às necessidades e expectativas do cliente, e assim garantindo a viabilidade da utilização do sistema [10].

O modelo do ciclo de vida de um software varia de acordo com a ordem em que suas fases irão ocorrer e quais fases receberão maior ênfase e compreenderão um maior período de tempo ao longo do processo. A escolha deste modelo é feita de acordo com fatores específicos do projeto, como orçamento, tempo disponível, equipe, ambiente operacional e perfil ou complexidade do negócio, que podem ser administrados e tratados de maneira mais eficiente por um modelo do que por outros, dependendo das circunstâncias em que as fases devem ser aplicadas [11].

Existem inúmeros modelos de ciclo de vida de *software*, alguns mais adequados ao panorama atual do mercado de desenvolvimento de sistemas e outros que já estão presentes há muitas décadas no ramo e ainda assim são utilizados até hoje, como o Modelo em Cascata, formalizado por Royce em 1970, cuja estrutura de processos está descrita na Figura 4 [11].

Figura 4. Representação da estrutura de etapas e processos do Modelo em Cascata utilizado em ciclos de vida do desenvolvimento de software [11]



Porém, para todos estes modelos existem etapas indispensáveis para o êxito do processo de software que, apesar de suas variações, estão fundamentalmente presentes em todos os modelos. O ciclo de vida do software é composto por três etapas essenciais, que podem assumir diferentes nomes ou serem fragmentadas em mais de uma etapa, dependendo do modelo no qual estão inseridas. São elas: definição, desenvolvimento e operação [12].

A fase de definição do software é o conjunto inicial de processos no ciclo de vida do software que tem como objetivo definir – de maneira ampla – os requisitos do software, ou seja, quais são as condições e viabilidades sob as quais aquele software deve ser desenvolvido – ou não – abrangendo desde aspectos econômicos e operacionais até restrições técnicas e uso de tecnologias específicas. Esta fase compreende diversas atividades que procuram estudar e especificar estes aspectos iniciais no processo de desenvolvimento do software, e será abordada de maneira mais ampla no próximo tópico [12].

Já a fase de desenvolvimento compreende diversas atividades, que variam para cada caso e modelo. Entre estas atividades estão englobadas a concepção e especificação visual, funcional e interativa do produto (design), a prototipação do software, a programação – caracterizada pela codificação e solução de problemas sob a forma de um programa – e a avaliação ou verificação, que permitem a correção, validação e usabilidade do software através de testes [12].

Por fim, o processo de operação dedica-se à instalação, utilização e manutenção do software, que compreende atividades como a implantação do software em máquinas, o uso do software por usuários, e o gerenciamento da qualidade do software para satisfazer novos requisitos de usuário, o transporte para novos sistemas operacionais e plataformas e corrigir erros [12].

2.1.2 Análise de Software (Papel do Analista)

A análise de software é uma denominação utilizada para definir etapas correspondentes ao processo de definição do sistema, dentro do ciclo de vida do desenvolvimento de software. A definição dos requisitos do sistema também recebe o nome de análise de requisitos, por conta do caráter de observação e minuciosidade presente na descrição de problemas e na proposta de soluções referentes aos componentes do sistema e suas viabilidades [12]. Tendo em vista a importância do processo de análise do software a ser desenvolvido, o papel do analista de sistemas é fundamental para o gerenciamento adequado do ciclo de vida do desenvolvimento de softwares [13].

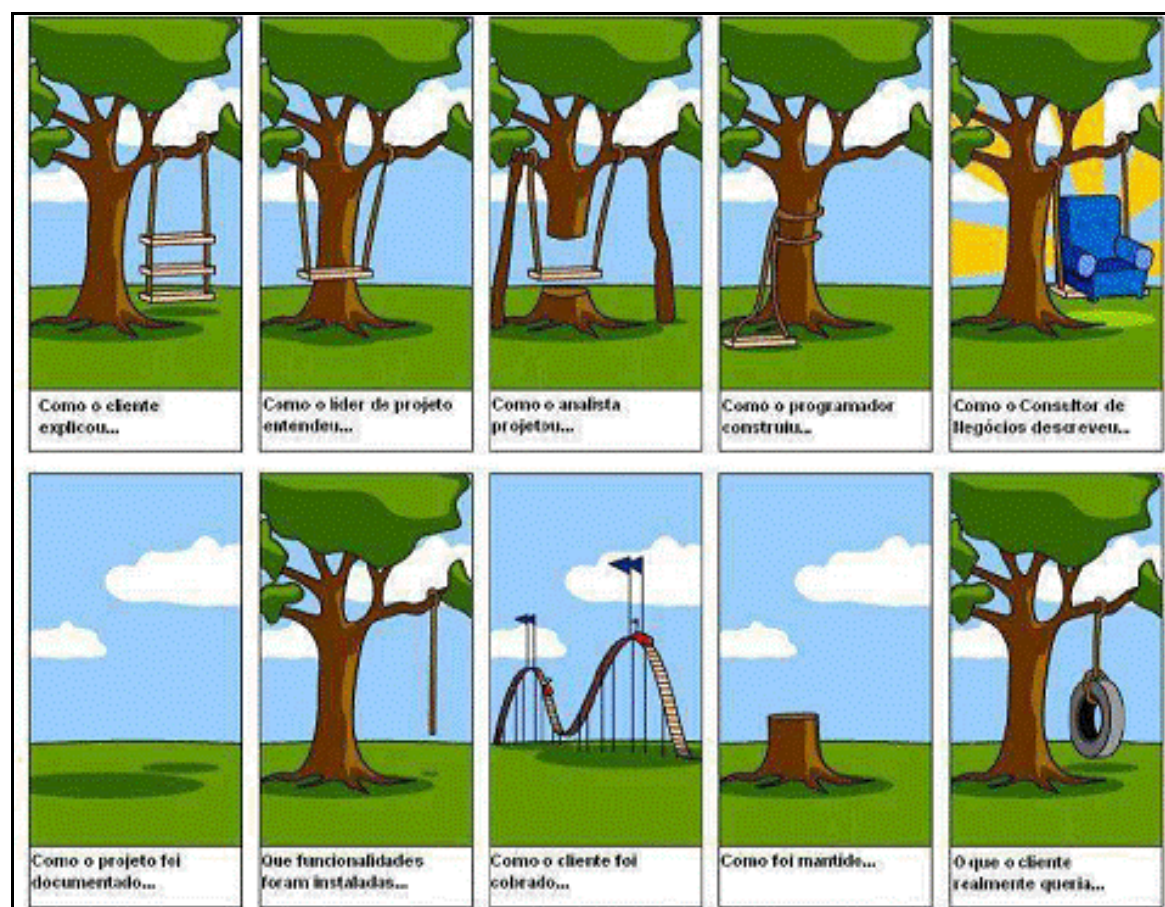
O analista de sistemas é o profissional responsável por atuar com análises e projetos de sistemas, levantamento de requisitos, documentação e mapeamentos de rotinas e processos, entre outros. É caracterizado como o principal encarregado da fase de definição e especificação dos componentes do sistema [13].

2.1.3 Requisitos

Os requisitos de um sistema são interpretados como as condições, restrições e propriedades sob as quais o software deve ser planejado e desenvolvido. Um requisito de um sistema é um aspecto que deve ser atendido pelo software para satisfazer especificações e padrões de acordo com as partes envolvidas no projeto. Portanto, o principal objetivo dos requisitos são definir o escopo e as características do projeto e alinhar as prioridades e a visão de software de todas as partes envolvidas ou interessadas no sistema [14].

Grande parte dos fracassos relacionados ao desenvolvimento de um software são provenientes de problemas causados de forma direta ou indireta através de falhas e erros referentes à especificação e análise de requisitos. Requisitos sem organização e detalhamento ou poucos suscetíveis a mudanças constantes durante o ciclo de vida do software geram problemas graves de coesão entre as partes envolvidas no sistema, contribuindo para um ambiente de grande propensão a transtornos no desenvolvimento do software e alinhamento à visão do cliente, exemplificados na Figura 5 [14].

Figura 5. Esquema representando os graves problemas causados por uma especificação falha de requisitos [14]



2.1.3.1 Requisitos funcionais e não funcionais

Existem duas classificações referentes à especificação dos requisitos e suas características, os requisitos funcionais (RF) e os requisitos não funcionais (RNF), que tem como objetivo proporcionar uma estrutura organizada e extremamente específica, mas que facilite o entendimento e compreensão conjunta de todas as partes envolvidas no sistema [14].

Os requisitos funcionais são identificados como os requisitos do sistema que se referem a tudo o que o sistema deve ser capaz de realizar e a maneira como deve se comportar, caracterizando-se como suas funcionalidades e informações. Enquanto isso, os requisitos não funcionais são utilizados como um mecanismo de suporte e parâmetros aos requisitos funcionais, sendo aplicados como critérios, propriedades e restrições que os qualificam.

Estes critérios podem ser pertinentes à qualidade do software, portanto, especificando ao sistema condições de usabilidade, confiabilidade, performance, entre outros, ou ainda, podem se referir ao processo do ciclo de vida do software em si, conferindo requisitos de entregas, implementação, utilização de tecnologias, entre outros. Na Figura 6 abaixo estão descritos alguns exemplos de requisitos funcionais e não funcionais [14].

Figura 6. Exemplos de requisitos funcionais e requisitos não funcionais de um sistema hipotético [14]

Requisitos Funcionais (RF)	Requisitos Não Funcionais (RNF)
[RF001] O Sistema deve ser capaz de cadastrar três tipos de usuários, sendo eles usuário comum, educador físico e nutricionista.	[RNF001] O Sistema deve disponibilizar opções de gênero no cadastro que atendam às normas do setor XYZ.
[RF002] O Sistema gerar o IMC do usuário automaticamente a partir do peso e altura do usuário.	[RNF002] O Sistema deve ser implementado em HTML, CSS, JavaScript e PHP.
[RF003] O Sistema deve disponibilizar uma função que permita ao usuário alterar sua senha.	[RNF003] O Sistema deve enviar o e-mail de recuperação de senha ao usuário em até 15 segundos.

2.1.4 Linguagem de Modelagem Unificada (UML)

A Linguagem de Modelagem Unificada, ou UML (acrônimo do idioma inglês), é uma linguagem padrão utilizada para especificar, visualizar e documentar o projeto de um sistema ou aplicativo, utilizando-se de recursos que permitam a elaboração de uma espécie de modelo visual que suporte o desenvolvimento do software, semelhante à ideia de realizar o desenho de uma planta para um projeto de arquitetura [15].

O modelo UML de um sistema, através de uma abordagem visual composta de diagramas, símbolos, e outros elementos, garante aos envolvidos no projeto uma visão objetiva e funcional de todos os aspectos operacionais do software, que serão desenvolvidos e levados em consideração durante todo o seu ciclo de vida [15].

A UML utiliza-se de diversos artefatos nesta complexa tarefa de modelar o sistema, e para isso possui nove tipos de diagramas, que são utilizados para representar diferentes enfoques de uma aplicação. Dentre estes nove, o diagrama de casos de uso é um dos principais mecanismos de especificação da UML, e será abordado com maior profundidade no próximo tópico e ao longo deste documento [16].

2.1.5 Diagrama de casos de uso

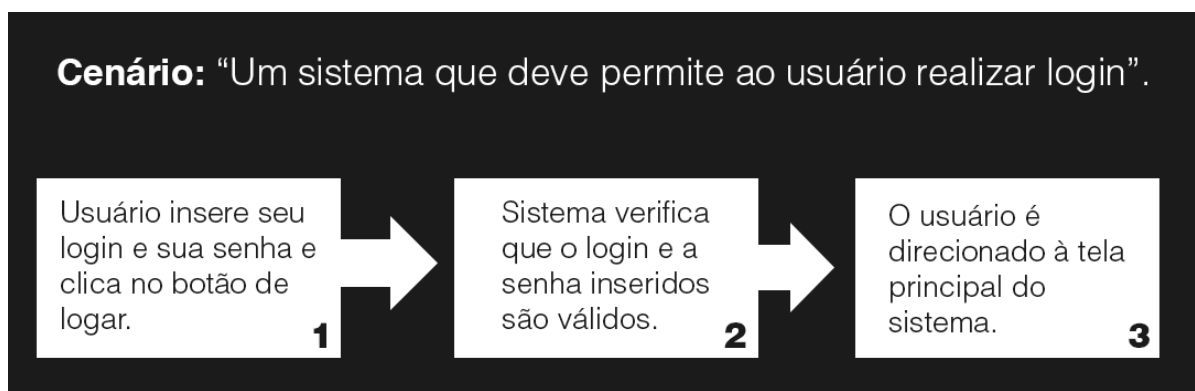
O diagrama de casos de uso tem como foco principal descrever o sistema e o uso de suas principais funcionalidades, relacionando-as com a interação do usuário. Em outras palavras, a função deste diagrama é simplesmente representar tudo aquilo que o sistema irá realizar de acordo com o ponto de vista do usuário da aplicação. O diagrama de casos de uso é um mecanismo que, no processo mais comum de desenvolvimento de software, deriva da especificação e documentação de requisitos, apresentando-se como um mecanismo visual que enfatiza o entendimento do sistema [16].

O diagrama de casos de uso é determinado através de quatro principais componentes. São eles: o cenário, o ator, o caso de uso, e os relacionamentos [16].

2.1.5.1 Cenário

O cenário do diagrama de casos de uso é caracterizado como uma sequência de eventos e ações que ocorrem continuamente, através da interação do usuário com o sistema. Qualquer desencadeamento consecutivo de acontecimentos no sistema causados através da utilização da aplicação representam um cenário específico, como no exemplo da Figura 7 abaixo [16].

Figura 7. Representação de um simples cenário de um sistema e seus eventos [16]



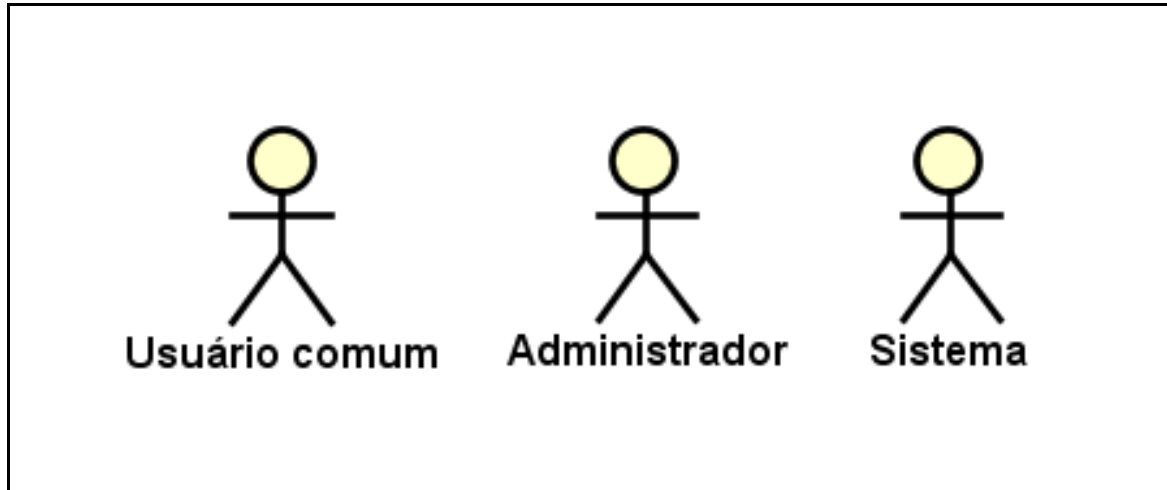
Dentro do cenário de casos de uso, estão contidos os outros três componentes fundamentais do diagrama. São eles: o ator, o caso de uso e o relacionamento [16]. Como o diagrama de casos de uso representa uma visão externa do sistema, tudo aquilo que constitui este diagrama está dentro de uma região que demarca seus limites, ao qual se atribui o nome de fronteira do sistema.

2.1.5.2 Ator

Dentro do diagrama de casos de uso, o ator é a representação de um tipo de usuário específico que atua de alguma forma nas funcionalidades da aplicação. Esse ator pode ser o

próprio usuário comum do sistema, um usuário com permissões mais avançadas, como um administrador, ou até mesmo o próprio sistema, que também atua ativamente nos cenários dos casos de uso e suas interações [16]. Na Figura 8 abaixo, é possível notar como o ator é representado visualmente no diagrama de casos de uso segundo o padrão da UML [16].

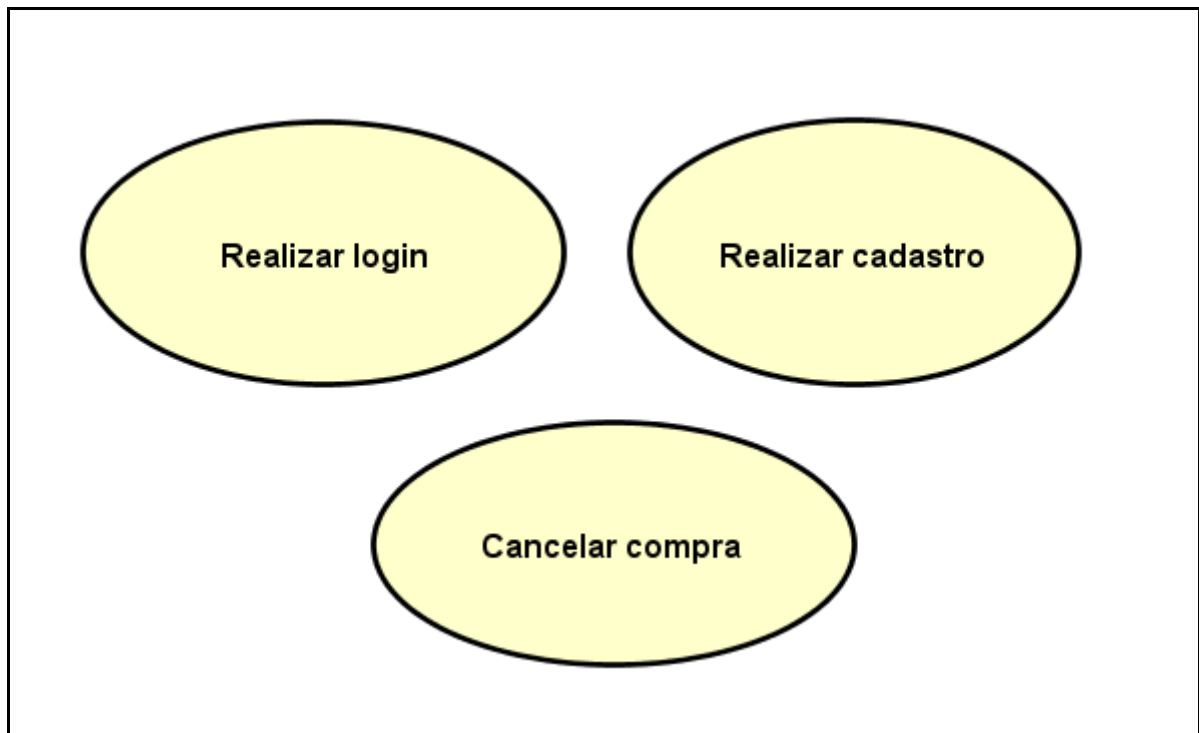
Figura 8. Representação visual do ator dos casos de uso segundo a UML [16]



2.1.5.3 Caso de uso

O caso de uso é o componente que indica uma funcionalidade ou tarefa que é realizada por um ator, dentro de um cenário do sistema. Portanto, o nome dado ao diagrama de casos de uso faz jus à função vital deste mecanismo para descrever o que será realizado, associar a quem irá realizar e quais eventos serão desencadeados dentro do sistema. Nos padrões da UML, o caso de uso é representado visualmente como na Figura 9 abaixo [16].

Figura 9. Representação visual do caso de uso segundo a UML [16]



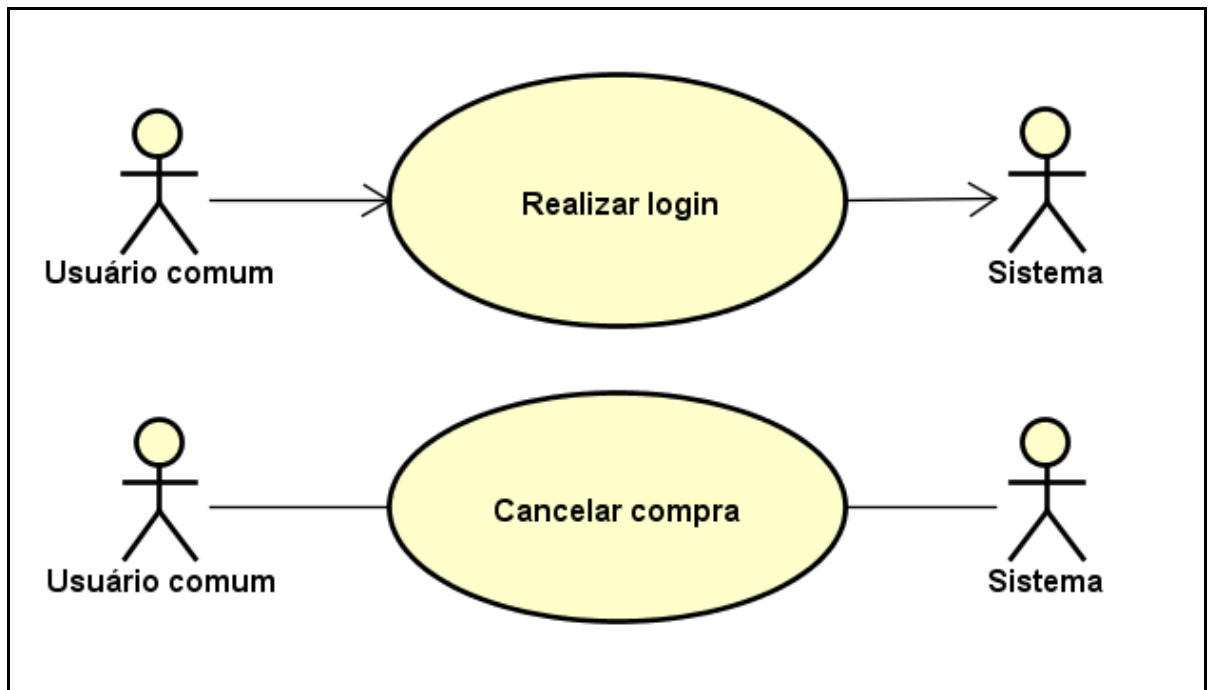
2.1.5.4 Relacionamentos

O relacionamento de um diagrama de casos de uso é um instrumento de conexão entre os demais componentes do diagrama, podendo caracterizar a interligação entre uma funcionalidade e um ator, entre funcionalidades, ou entre atores. Utilizando-se de uma analogia, os relacionamentos do diagrama são uma espécie de cola que irá grudar e unir os elementos do sistema e indicar que tipo de comportamento estes elementos irão adotar nas relações entre si [17].

Para isso, existem diversos tipos de relacionamentos na UML, que se distinguem justamente nas características de conexão que são atribuídas aos elementos interligados por estes relacionamentos [17]. Para os propósitos deste trabalho, iremos abordar quatro tipos de relacionamentos: associação, inclusão, extensão e generalização [18].

O relacionamento de associação é o mais simples e comumente utilizado no diagrama de casos de uso, e tem a função de indicar uma conexão entre um ator e um caso de uso, demonstrando que o ator em questão utiliza a funcionalidade descrita no caso de uso, seja através de sua execução ou do recebimento do seu resultado. A associação é representada por uma linha reta, podendo ser direcionada ou não – a linha não direcionada é mais comum – como indigita o exemplo da Figura 10 [18].

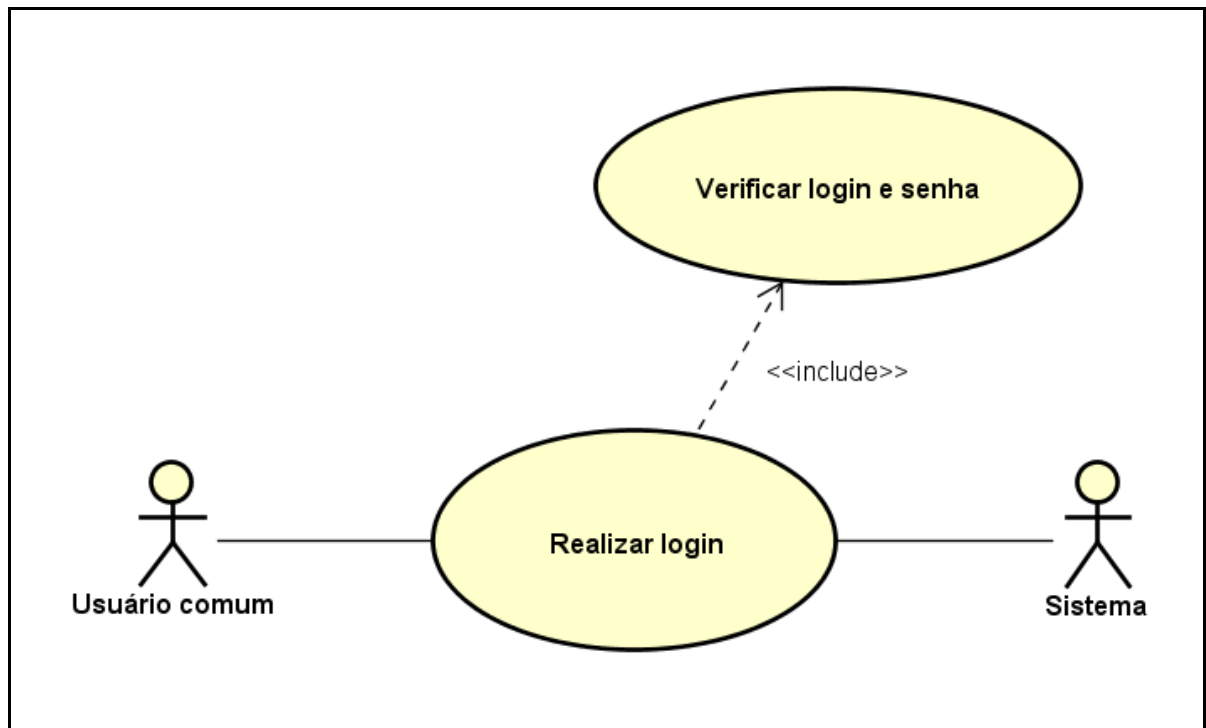
Figura 10. Exemplo de relacionamento de associação entre ator e caso de uso [18]



O relacionamento de inclusão (*include*) é utilizado para interligar dois casos de uso, de maneira a indicar que o segundo caso de uso está incluso dentro do primeiro, indicando obrigatoriedade. Portanto, a execução do primeiro caso de uso caracteriza a execução mandatória do segundo caso de uso. O relacionamento de inclusão é representado por uma seta tracejada que aponta na direção do caso de uso que está sendo incluído à sequência de eventos, e que carrega consigo o termo “<<include>>”.

Na Figura 11 abaixo, o *include* está indicando que, para que o caso de uso “Realizar login” seja executado, obrigatoriamente o caso de uso “Verificar login e senha” também deve ser executado [18].

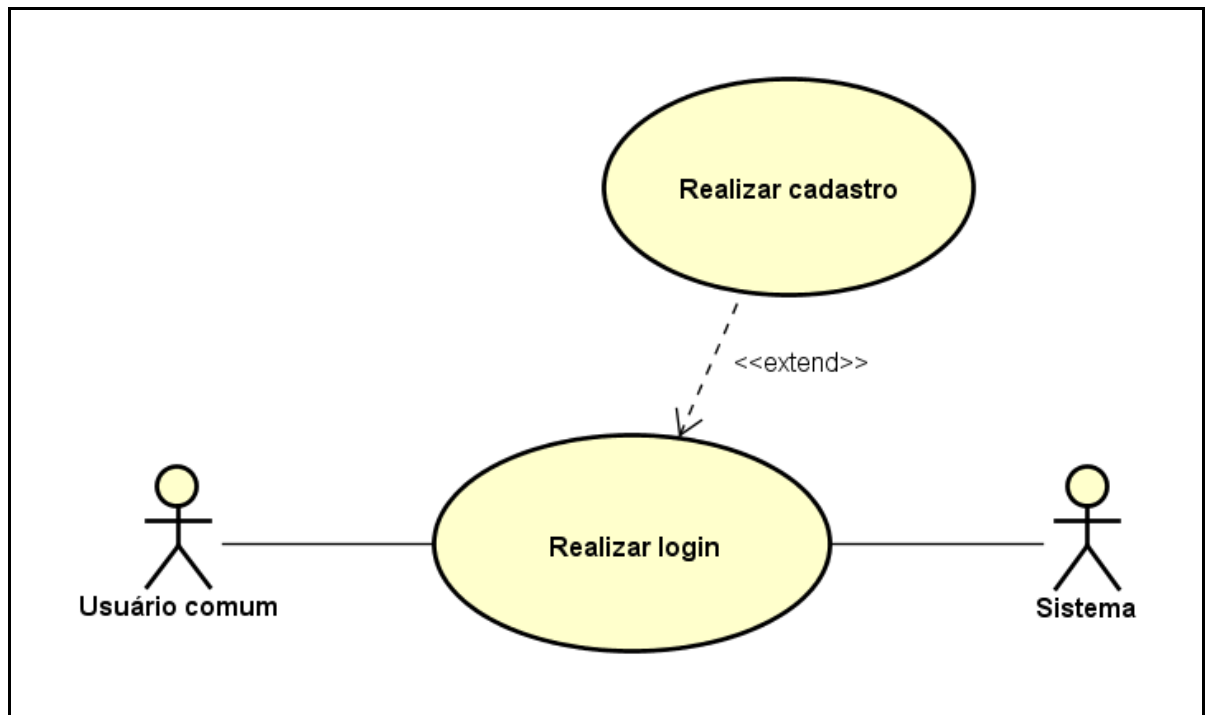
Figura 11. Exemplo de relacionamento de inclusão entre dois casos de uso [18]



O relacionamento de extensão (*extend*) é uma conexão que caracteriza um caso de uso como uma funcionalidade opcional de outro caso de uso, ou seja, uma extensão que não precisa ser executada obrigatoriamente. O *extend* é utilizado, por exemplo, em cenários que somente acontecerão sob condições específicas. Sua representação visual é semelhante à do *include*, uma seta tracejada que parte do caso de uso extensor, apontando na direção do caso de uso estendido [18] [19].

No exemplo da Figura 12 a seguir, a direção da seta do *extend* indica que o caso de uso “Realizar cadastro” é uma funcionalidade opcional do caso de uso “Realizar login”. Portanto, a execução de “Realizar cadastro” não é obrigatória e representa uma condição específica, que neste caso seria o usuário ainda não ter se cadastrado no sistema [18] [19].

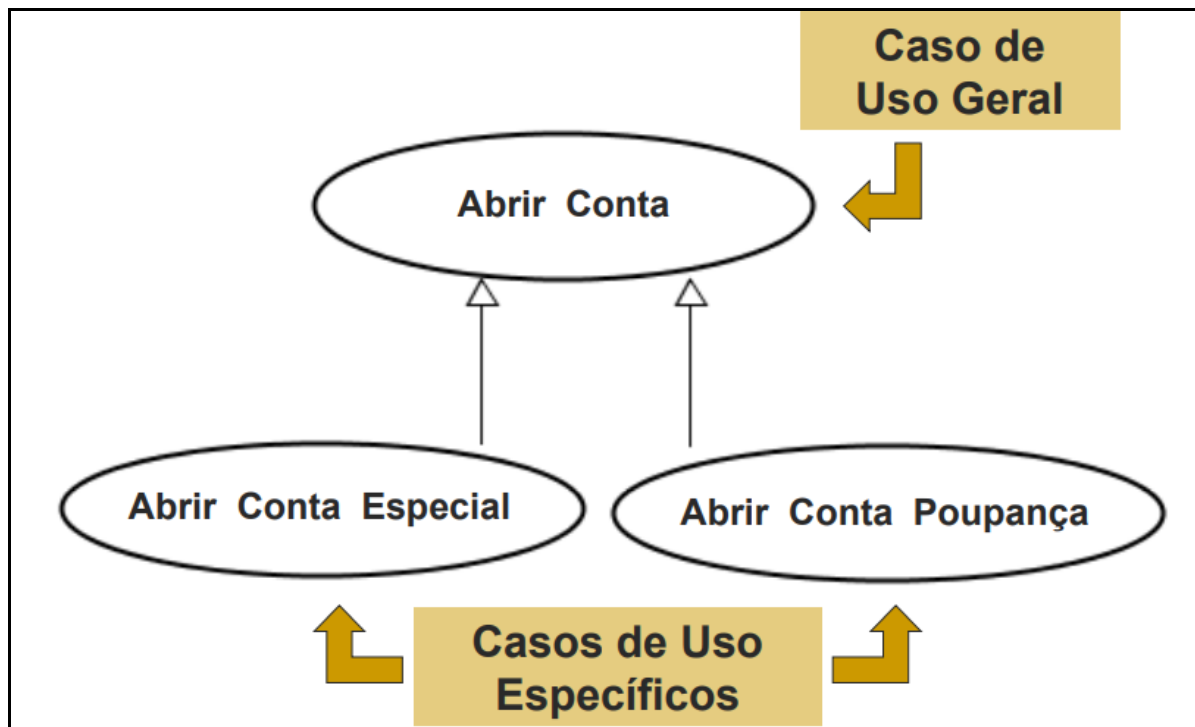
Figura 12. Exemplo do relacionamento de extensão entre dois casos de uso [18] [19]



Por último e não menos importante, o relacionamento de generalização é utilizado para estabelecer uma relação de herança e semelhança entre dois casos de uso ou atores. A generalização (ou especificação) indica a existência de, por exemplo, dois casos de uso com características semelhantes, porém um deles possui um grau de especificidade mais elevado. Para garantir a reutilização e evitar repetições, o caso de uso específico (filho) irá herdar as características e propriedades do caso de uso geral (pai) [18] [19].

O relacionamento de generalização é representado por uma seta mais espessa que aponta na direção do caso de uso geral, o pai. O exemplo da Figura 13 abaixo indica o grau de especificidade que os casos de uso “Abrir Conta Especial” e “Abrir Conta Poupança” contêm, necessitando de um relacionamento de generalização, comum ao caso de uso geral “Abrir Conta”, herdando todas as características que compartilham. Desse modo, os casos de uso específicos se encarregam apenas de simbolizar suas próprias características, que são referentes somente àquele caso de uso em particular [18] [19].

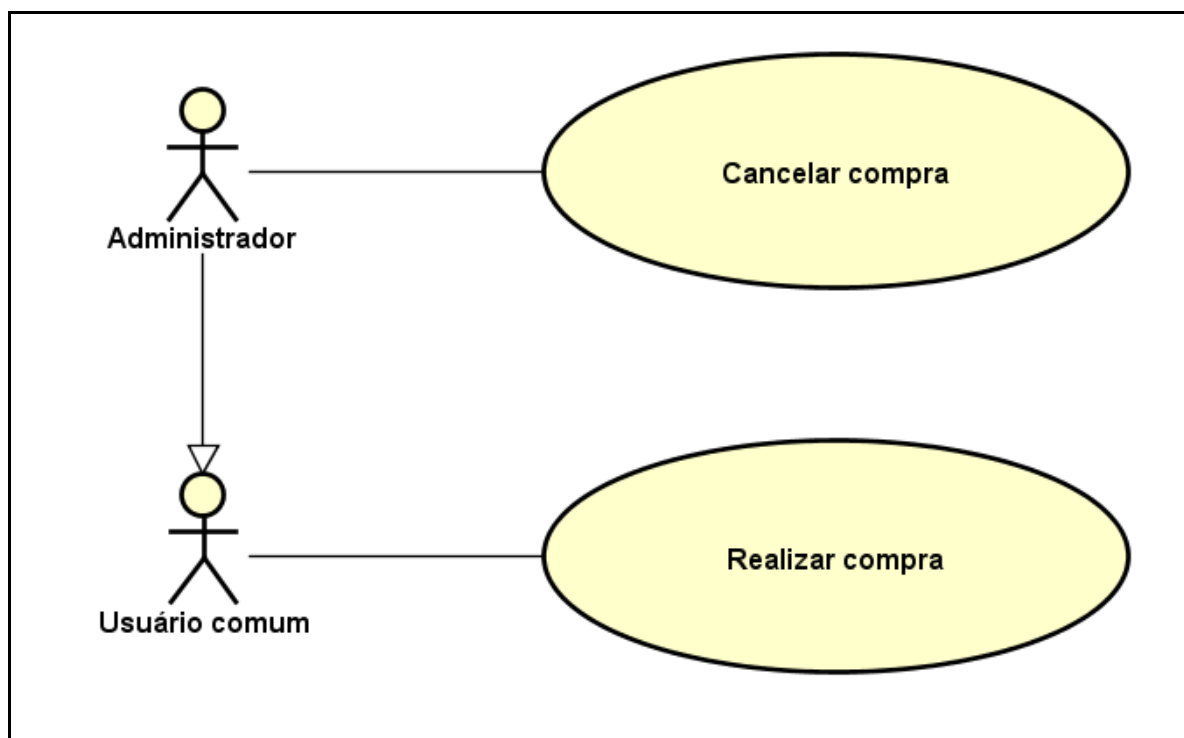
Figura 13. Exemplo de relacionamento de generalização entre dois casos de uso específicos e um caso de uso geral [18] [19]



Esta mesma dinâmica de generalização exemplificada acima vale também para o relacionamento de generalização entre atores. Um relacionamento de generalização entre atores é utilizado para estabelecer uma relação de herança, indicando que um ator descendente (filho) recebe todos os atributos e comportamentos de seu ator ascendente (pai). Desta forma, o ator filho pode atuar como o ator pai e realizar todas as suas funcionalidades, assim representando apenas as funcionalidades específicas que seu ator pode realizar em particular [20].

No exemplo da Figura 14 representada abaixo, podemos notar que o ator filho “Administrador” herda todos os atributos e comportamentos do ator pai “Usuário comum”. Em outras palavras, o ator “Administrador” pode agir como o ator “Usuário comum”, e realizar todas as suas funcionalidades, que neste caso trata-se apenas do caso de uso “Realizar compra”. Além disso, o ator “Administrador” também pode associar-se a outros casos de uso, que indicam funcionalidades específicas que podem ser realizadas somente pelo ator “Administrador” em particular. Neste exemplo, esta funcionalidade é descrita pelo caso de uso “Cancelar compra” [20].

Figura 14. Exemplo de relacionamento de generalização entre um ator ascendente (pai) e um ator descendente (filho) [20]



2.1.6 Documentação de Casos de Uso

Nos tópicos anteriores, foram introduzidos ao leitor todas as questões e componentes relevantes acerca do diagrama de casos de uso. Ao item final do levantamento bibliográfico deste trabalho dedicam-se as modalidades referentes ao método de documentação dos casos de uso de um sistema, de modo a contribuir para a compreensão dos processos a serem discutidos nas etapas para o desenvolvimento da pesquisa, que tem como foco os casos de uso do módulo de treinos do projeto Mais Saúde São João.

A documentação de casos de uso tem como objetivo traduzir a representação visual dos diagramas da UML em uma linguagem escrita simples e de fácil entendimento, de forma a aprofundar o nível de detalhe das funcionalidades do software a ser desenvolvido. Através dessa documentação, é possível estabelecer uma coerência entre as partes envolvidas na análise e especificação do sistema e no seu desenvolvimento [21].

Portanto, é essencial unir todos os componentes de fundamental importância ao desenvolvimento do sistema em um documento, e para isso diversos elementos podem ser documentados, como atores que interagem com os casos de uso, parâmetros que devem ser fornecidos, etapas de funcionalidades a serem executadas, entre outros [21].

Contudo, existem diversas maneiras de se documentar os casos de uso de um sistema, não havendo um formato específico padronizado. A forma como a documentação será conduzida e

estruturada depende de diversos fatores que podem variar de acordo com o projeto que está sendo realizado [21]. Neste trabalho, a documentação de casos de uso será abordada de acordo com o método e estrutura utilizados na especificação de casos de uso de todos os módulos do projeto Mais Saúde São João.

Dentro desta estrutura, estão situados três componentes principais que tem como objetivo descrever e detalhar o comportamento e as etapas de cada caso de uso. São eles: fluxo principal, fluxos alternativos, e protótipo de interface homem-máquina [21].

2.1.6.1 Fluxo principal

O fluxo principal de um caso de uso é também chamado de caminho feliz, pois descreve as etapas que caracterizam um cenário padrão e desejável da funcionalidade. Ou seja, o fluxo principal é o que o usuário tentará fazer primariamente sempre que utilizar a funcionalidade daquele caso de uso. A ideia deste fluxo é indicar a sequência de eventos que deve ser percorrida para que o objetivo primário da funcionalidade seja atingido [22].

É possível exemplificar as características do fluxo principal de uma funcionalidade na vida real. José mora em São João da Boa Vista e deseja chegar o mais rápido possível na cidade vizinha de Águas da Prata. Para isso, José tem três opções de transporte: ir de carro, de bicicleta, ou a pé. Nesta situação, caso José deseje chegar o mais rápido possível à Águas da Prata, sua principal forma de transporte é o carro. Ou seja, o carro é seu fluxo principal [22].

2.1.6.2 Fluxos Alternativos

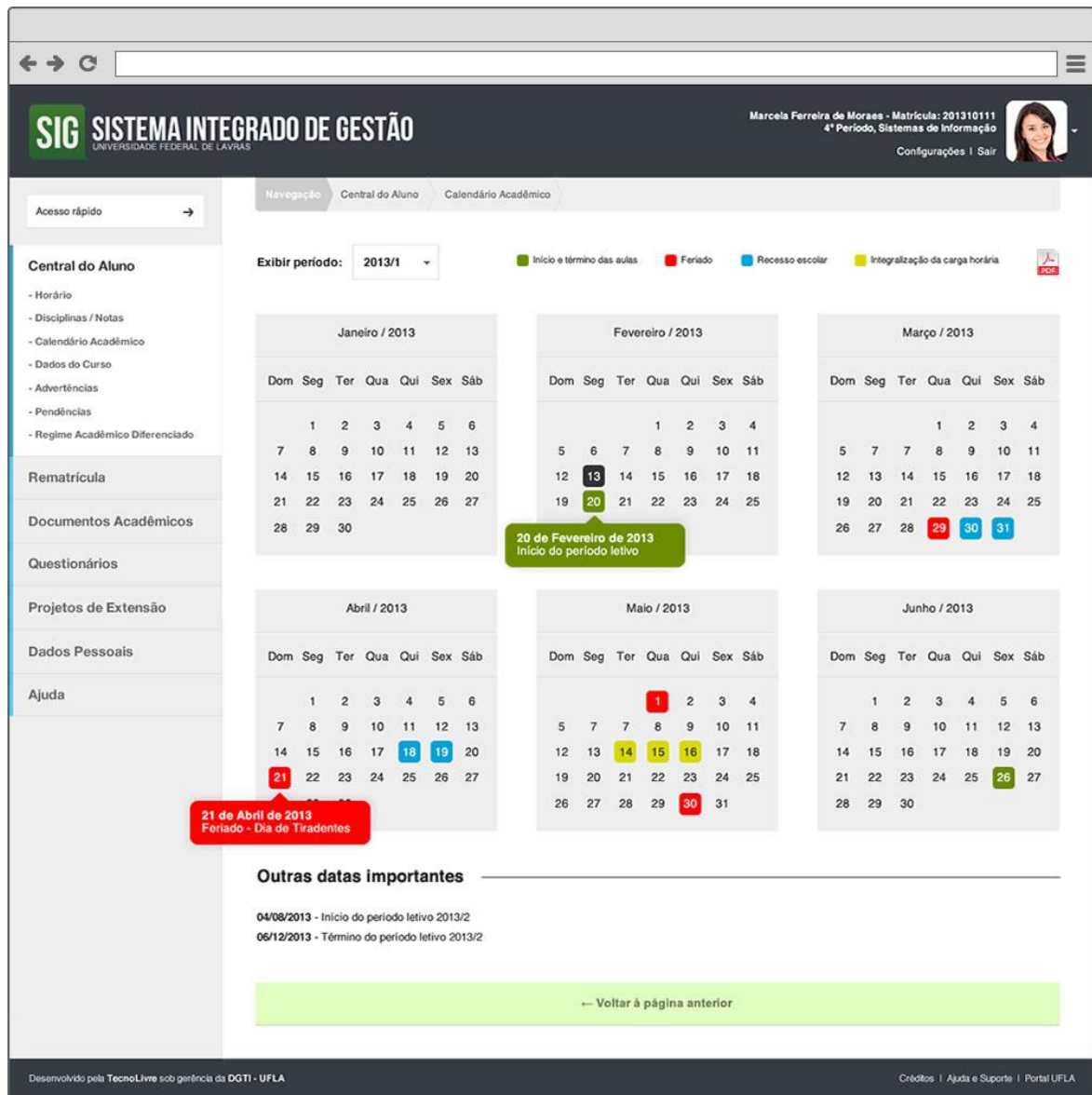
Os fluxos alternativos de um caso de uso também podem ser compreendidos utilizando-se do exemplo anterior. A forma de transporte principal para José chegar a Águas da Prata o mais rápido possível é o carro. Sendo assim, ir de bicicleta ou a pé até Águas da Prata são formas alternativas de transporte que fogem do objetivo primário de José, mas que estão disponíveis caso José não consiga utilizar seu carro. Esses são os fluxos alternativos de um caso de uso. São cenários alternativos ao objetivo principal da funcionalidade do caso de uso, mas que devem existir caso haja algum erro ou exceção durante a execução do fluxo principal daquela funcionalidade [22].

2.1.6.3 Protótipos

Os protótipos de interface de um caso de uso são uma representação visual das funcionalidades descritas nos fluxos principal e alternativos do caso de uso. A utilização de protótipos é um método eficaz de reconhecer e simular de maneira prática e visual a interface e suas tarefas que serão implantadas no desenvolvimento do software [23].

A interface é o principal mecanismo de comunicação entre um usuário e o sistema, e esta caracterização conduz a importância da realização de um protótipo desta interface, estabilizando as etapas dos casos de uso de uma maneira operacional com um caráter mais prático e visual. Na Figura 15 abaixo, está indicado o protótipo de interface de um sistema [23].

Figura 15. Exemplo de protótipo de interface de um sistema de gestão [23]



2.2 Etapas para o Desenvolvimento da Pesquisa

Neste tópico do documento serão descritos e aprofundados os tópicos referentes às etapas do desenvolvimento da pesquisa que contém seu enfoque na especificação dos casos de uso do Módulo de Treinos do projeto Mais Saúde São João.

2.2.1 Entendimento dos métodos iniciais de análise para o posterior levantamento de requisitos e viabilidades do Módulo de Treinos

Nos estágios iniciais de concepção do projeto, o desencadeamento de uma série de processos e ajustes impactaram na divisão e definição das funcionalidades do sistema Mais Saúde São João. Baseando-se nas necessidades da população da cidade de São João da Boa Vista e região, e na proposta do produto como sendo uma ferramenta que possibilitaria aos cidadãos o acesso a uma série de recursos e ferramentas referentes à saúde, exercícios físicos, nutrição e esporte, a divisão entre módulos tinha como objetivo vital demarcar limites e setores de interação distintos dentro do mesmo sistema, assim como contribuir para uma organização dos pontos de acesso e ação dos diversos atores do projeto, como a população comum, educadores físicos, nutricionistas, administradores, mecanismos como Google Maps e Google Charts, entre outros [5].

2.2.1.1 Definição de abordagem da proposta inicial do grupo de funcionalidades do Módulo de Treinos

Acerca da divisão do sistema, o Módulo de Treinos encaixou-se à frente de um conjunto de funcionalidades que pareciam extremamente desconexas dentro do mesmo módulo nos estágios prematuros de análise e concepção. Ao mesmo tempo em que o módulo estaria encarregado das interações entre usuário comum e educador físico, também estaria atrelado a uma série de aspectos do sistema que tratavam de setores totalmente distintos do complexo usuário-educador físico.

As funcionalidades do Módulo de Treinos que se difundiam além desse complexo de interação eram relacionadas à gestão de suplementos alimentares, dados do usuário, consultas e fichas de treinamento, e pareciam abordar uma série de interfaces e focos muito distantes entre si, como exemplificado na Figura 16 abaixo.

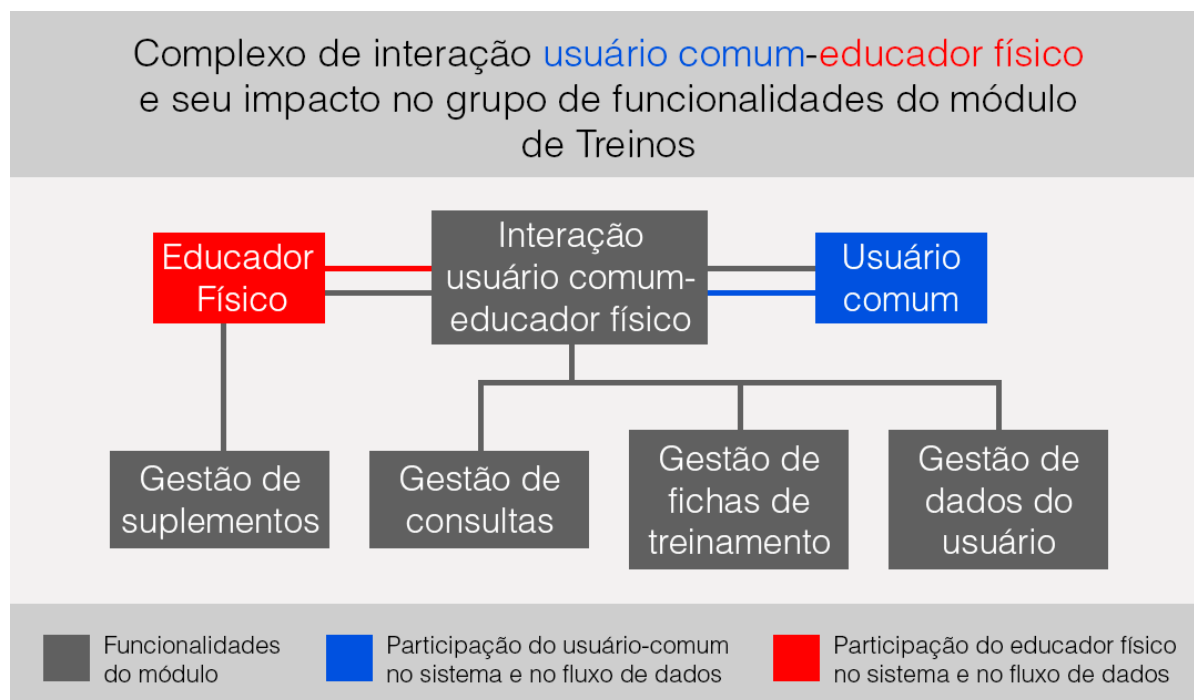
Figura 16. Descrição da concepção inicial das funcionalidades do Módulo de Treinos [5]

- 4. Criar um conjunto de ferramentas de Treinos com as seguintes especificações:**
- *Gestão e restrição da interação entre usuário e educador físico a apenas um educador físico para cada usuário;*
 - *Gestão do agendamento de consultas presenciais, com o educador físico, para o usuário;*
 - *Definição e controle do caráter e contexto da consulta, de acordo com o feedback do usuário e do educador físico;*
 - *Controle das fichas de treinamentos específicos e detalhados destinados ao usuário;*
 - *Apresentação ao usuário sobre Suplementos Alimentares;*
 - *Controle do gerenciamento das medidas corporais e dobras cutâneas do usuário; e*
 - *Execução e gestão do envio por e-mail de novas fichas de treinamento cadastradas ao usuário.*

Entretanto, a aparente desconexão entre as funcionalidades iniciais do módulo foi solucionada através de uma nova abordagem dos envolvidos na análise nas fases iniciais do projeto. A solução foi direcionada às propostas de concepção processual do módulo de Treinos, que passou a ser estruturado e pensado tomando como referência o próprio complexo de interação entre usuário comum e educador físico.

Cada funcionalidade ou grupo de funcionalidades propostas ao módulo tratava de um aspecto distinto, relacionado às necessidades do público alvo, mas todas continham um potencial muito próximo com a população e a figura do educador físico, caracterizando uma oportunidade de abordar todas as funcionalidades do módulo através da centralização na troca constante de dados entre estes dois atores e sua interação ativa com o sistema, como mostra a Figura 17.

Figura 17. Esquema de representação do grupo de funcionalidades do módulo de Treinos de acordo com a interação entre usuário comum e educador físico [5]



Desta maneira, o conjunto de requisitos do módulo se tornaria muito mais integrado e coeso de acordo com as necessidades do público alvo e com as propostas de funcionalidades do usuário comum e do educador físico. Outro benefício seria aumentar a acessibilidade do sistema para todos os tipos de usuários ao simplesmente torná-lo mais fácil de compreender através do destaque de setores distintos, fazendo com que qualquer usuário consiga assimilar e associar de forma prática que as funcionalidades X, Y e Z fazem parte do mesmo grupo e propósito de funções.

2.2.1.2 Elaboração das dinâmicas operacionais das funcionalidades com enfoque na interação e logística do usuário comum e educador físico

Utilizando-se da abordagem do grupo de funcionalidades do módulo em função do complexo entre usuário comum e educador físico, os envolvidos na análise focaram-se em discussões e tentativas constantes de adequar as viabilidades e a dinâmica de operação do sistema – que seriam documentados e detalhados através do levantamento de requisitos – de acordo com as noções de logística e interação já existentes que envolvam o perfil do público alvo, do educador físico, e suas características.

A partir deste divisor de águas no processo de análise, e tendo como influência o contexto de restrições e condições ambientais e técnicas do módulo, começaram a surgir os primeiros levantamentos das funcionalidades do Módulo de Treinos aplicando a abordagem definida e explicada no subtópico anterior. Para caracterizar e aprofundar o nível de detalhe do processo de

análise destes requisitos prematuros, será utilizado um exemplo prático de uma das documentações realizadas pelos envolvidos na análise que discorre sobre a gestão do agendamento de consultas com o educador físico, serviço destinado ao usuário comum do sistema [24].

O sistema deve se dispor de uma interface que compreenda duas vias de dados, a disponibilidade de consultas do educador físico e o agendamento de consultas do usuário comum. Para que haja o agendamento do usuário comum, o educador físico deve gerir os dados de disponibilidade através de uma interface que permita definir o local, as datas e os horários em que o educador físico possa realizar consultas. A possibilidade de customização da disponibilidade atende às necessidades logísticas do educador físico de evitar consultas em momentos em que não esteja disponível [24].

Simultaneamente, o agendamento de consultas do usuário comum é realizado através de uma interface que possibilite marcar, remarcar e cancelar consultas. As datas e horários de até 24 horas após o momento em que a consulta está sendo marcada estarão bloqueados. Desta maneira, as dificuldades logísticas entre usuário comum e educador físico foram adaptadas a uma dinâmica de operação que permita ao usuário comum flexibilizar suas consultas em caso de imprevistos, e forneça ao educador físico um ambiente organizado de atendimento através do recebimento de notificações pontuais e marcadas com antecedência considerável de no mínimo um dia [24].

A seguir encontra-se outro exemplo de documentação, relacionado à própria gestão da de interação entre o usuário comum e o educador físico [24].

O sistema deve prover um mecanismo de controle que restrinja o usuário comum a associar-se a apenas um único educador físico de acordo com sua escolha. Para garantir uma interação eficiente e de acordo com as necessidades do público alvo, o educador físico é tratado pelo sistema como um tipo especial de usuário que contém dados de cunho profissional e pessoal – que sejam relevantes e convenientes com a utilização do sistema. Portanto, o sistema se dispõe de diversas informações que irão auxiliar na procura do educador físico por parte do usuário comum, que será feita através de uma interface semelhante à uma loja online que exiba educadores físicos de acordo com filtros de pesquisa definidos pelo usuário comum, como gênero, faixa etária e idade, e classificações do sistema, como a disponibilidade do educador físico para associar-se a novos usuários, sendo que os educadores físicos indisponíveis nem sequer estão incluídos na interface de pesquisa) [24].

Após a seleção do educador físico de sua preferência, o sistema deve permitir ao usuário comum a visualização completa dos dados do educador físico e a possibilidade de enviar uma solicitação de acompanhamento ao educador físico através de um sistema de notificações, de maneira a simular uma dinâmica ambiental semelhante à de envio de solicitações de amizade em

redes sociais, tornando o processo de solicitação de educador físico uma funcionalidade familiar, acessível e prática à maioria eminente do público alvo do sistema. A partir deste cenário, o educador físico tem o recurso de avaliar se deseja aceitar ou não a solicitação enviada pelo usuário comum [24].

Após uma eventual resposta do educador físico, o usuário recebe uma notificação de retorno atestando se a partir de agora está associado ou não ao educador físico em questão. Caso a resposta seja positiva, o usuário já deve ser habilitado pelo sistema de recursos de interação com o educador físico, como agendamento de consultas, acesso à uma interface de perfil de seu novo associado, e até mesmo a opção de dispensar o educador físico, finalizando a interação direta entre os dois indivíduos e desbloqueando o usuário comum a novas oportunidades de acompanhamento [24].

Novamente por conta de questões logísticas, o sistema deve gerir a interação de maneira que, enquanto o usuário comum estiver já estiver associado a um educador físico, ele poderá apenas pesquisar por outros educadores físicos, mas não poderá enviar solicitação a nenhum deles. Ou seja, o usuário comum pode avaliar e entrar em contato com novas opções caso esteja insatisfeito com seu educador físico, mas deverá se comprometer a trocar de educador físico, evitando ocasiões em que o usuário não compreende a rigidez do sistema com relação à divisão igualitária de serviços à população [24].

Como exemplificado nos parágrafos anteriores, o levantamento inicial de requisitos do Módulo de Treinos foi concebido de acordo com a abordagem definida para a proposta das funcionalidades, que foi centralizada em torno dos atores protagonistas do módulo, o usuário comum e o educador físico. Além disso, os envolvidos na análise focaram-se na necessidade de identificar componentes essenciais para criar interfaces e mecanismos de interação que fossem práticos, acessíveis ao vasto público alvo do projeto, e representassem circunstâncias cotidianas que facilitam o entendimento das funções de cada ferramenta do sistema.

2.2.2 Concepção do processo de conversão dos requisitos do sistema nas funcionalidades detalhadas que constituem os Casos de Uso

Após a fase inicial de levantamento e discussão com relação à análise de funcionalidades, a especificação de requisitos passou a ser documentada e adaptada a um alto nível de detalhamento que possibilitasse uma transformação precisa dos requisitos em casos de uso do sistema, desta maneira facilitando as circunstâncias futuras de documentação.

2.2.2.1 Implementação de alto nível de detalhamento aos requisitos

O nível de detalhamento implementado pelos envolvidos na análise envolveu a concepção das dinâmicas das interfaces que constituem os requisitos definidos anteriormente. Desta maneira, o levantamento de requisitos inicialmente realizado – que continha o objetivo exclusivo de descrever as intenções das funcionalidades e condições logísticas do módulo – foi aprimorado de forma a caracterizar os cenários propostos aos mecanismos de interação e explicar minuciosamente como os objetivos da abordagem de análise em torno do usuário e comum e do educador físico seriam alcançados, através das estruturas visuais que teoricamente funcionariam para cada situação específica dos requisitos.

Um exemplo desta implementação – retirado de um dos produtos finais desta etapa de análise do módulo – é a descrição detalhada de como o agendamento de consultas, que está presente em um dos requisitos iniciais, seria realizado pelo usuário. Nesta descrição, foram especificados quais as etapas exatas do cenário em que o usuário comum marca uma consulta, quais as exceções logísticas deste cenário, através de quais elementos visuais o usuário comum realizaria esta funcionalidade, e de que maneira estes elementos impactariam na estrutura da interface [24].

2.2.2.2 Conversão do levantamento de requisitos com foco na concepção inicial dos casos de uso

A partir desta etapa do processo de análise do Módulo de Treinos, os indivíduos envolvidos direcionaram-se ao método de interpretar e utilizar os requisitos como uma ferramenta para facilitar a especificação dos casos de uso.

Os objetivos a serem cumpridos para que a definição e documentação dos casos de uso fosse realizada com êxito são diversos e dependentes entre si. Estes objetivos englobam:

- a identificação de todos os atores que interagem e executam as funcionalidades propostas nos requisitos, incluindo tecnologias externas ao desenvolvimento;
- a fragmentação de cada requisito especificado em um ou mais casos de uso, de acordo com a necessidade de desagregar cenários e fluxos operacionais para aprimorar a dinâmica e a praticidade do sistema;
- a elaboração das coexistências entre funcionalidades e usuários dentro de um mesmo sistema, resultando na diagramação e interligação de casos de uso e atores através de relacionamentos;
- o posterior enfoque nas microestruturas e processos que compõem os casos de uso, caracterizando a classificação de seus cenários, fluxos principais e alternativos.

A listagem dos objetivos acima indica a ordem na qual os processos de análise referentes a este subtópico serão abordados e procura garantir ao leitor uma melhor compreensão do método de elaboração dos casos de uso utilizado nesta situação específica. Inicialmente, é preciso notar em que pé começou este processo, que se deu através dos denominadores comuns provenientes de todas as interações e fluxos de dados, os atores.

A identificação de atores foi discutida pelos envolvidos na análise como sendo o meio mais simples e prático de iniciar o processo de elaboração dos casos de uso, considerando a facilidade na qual é possível realizar esta identificação. Ao encargo desta etapa inicial foram atribuídas uma série de perguntas que, quando respondidas, auxiliam na tarefa de concluir quais atores estão envolvidos nos casos de uso em questão.

Estas perguntas foram concebidas baseando-se nas questões logísticas e operacionais dos requisitos, e procuravam identificar a presença de qualquer usuário ou tecnologia que indicasse qualquer tipo de participação ativa nas funcionalidades em questão. Tendo isso em mente, o caráter de questionamento era direcionado a:

- quais são os usuários que realizam ou estão envolvidos nos processos das funcionalidades;
- caso sejam utilizadas, quais são as tecnologias ou aplicações, externas ao sistema, que oferecem mecanismos visuais e auxiliares a quaisquer operações;
- caso sejam utilizadas, quais são as ferramentas que garantem suporte ao fluxo e armazenamento de dados entre o sistema e destinos externos.

A partir deste método de identificação, os envolvidos na análise rapidamente concretizaram uma relação de atores com diversas caracterizações. Em seguida, estes atores receberam uma classificação descritiva de sua função dentro do sistema, e de que maneira seu comportamento se procederia em contato com os outros elementos. Os aspectos principais deste processo foram levantados na Figura 18 abaixo.

Figura 18. Principais aspectos do processo de identificar e caracterizar os atores dos casos de uso [25]



Como pode ser observado, a elaboração de níveis de permissões e setores impactados pelos atores já começa a ser elaborado em conjunto com este processo. É importante compreender que o próprio sistema também pode ser considerado um ator de caso de uso, afinal de contas ele é onipresente e responsável pela operação de todas as funcionalidades. Entretanto, o sistema, neste caso, não foi considerado um ator, pois sua atuação e impacto foram interpretados como circunstâncias implícitas que englobam todos os casos de uso, não representando uma necessidade de tornar o sistema um ator.

Tendo os atores já identificados e caracterizados, se evidencia uma próxima etapa na especificação de casos de uso que tem o objetivo de passar a definir que funcionalidades e casos de uso o sistema irá conter, em contato com estes atores, e tomando como base o levantamento de requisitos das fases anteriores de análise.

Para exemplificar o processo de fragmentação dos requisitos, será utilizado como base uma descrição já apresentada anteriormente, relacionada à gestão e agendamento de consultas, evidenciando o complexo de interação entre usuário comum e educador físico. Para recapitular, a proposta presente no levantamento deste requisito implicava na existência de um conjunto de interfaces que possibilitassem ao usuário comum marcar, remarcar e cancelar consultas a partir de parâmetros de disponibilidade definidos pelo educador físico.

De maneira a preservar a logística e acessibilidade do sistema, os envolvidos na análise procuraram desenvolver uma fragmentação que definisse explicitamente as potenciais funcionalidades presentes neste requisito. Assim, o requisito de gestão e agendamento de consultas

foi quebrado em quatro casos de uso, concretizando a existência de duas interfaces distintas para gerir o agendamento de uma consulta, uma única interface destinada ao gerenciamento da disponibilidade da consulta, e um último caso de uso encarregado de representar a funcionalidade envio de notificação do agendamento via e-mail a ambos os atores envolvidos no processo [25].

O principal método para definição dos casos de uso foi identificar na descrição dos requisitos a existência de cenários específicos e que deve ser abordado individualmente de acordo com suas exceções e particularidades. Geralmente, este cenário indica a existência de uma interface própria para conduzir este processo, e, conseqüentemente, um novo caso de uso é criado em função disso. Uma melhor compreensão pode ser obtida através dos quatro casos de uso resultantes do requisito descrito acima.

Um primeiro caso de uso, chamado “Visualizar consultas do usuário”, representa a funcionalidade inicial do usuário comum para verificar se ele já tem uma consulta marcada com o seu educador físico, que é expressada por uma interface que irá representar cenários distintos caso o usuário comum já tenha marcado uma consulta ou não. Dependendo das circunstâncias, o usuário comum terá a opção de marcar uma nova consulta, ou remarca-la caso já tenha a agendado. Por questões práticas, ambas as opções são interpretadas pelo sistema como o agendamento de uma nova consulta [25].

De qualquer maneira, esta opção resulta na necessidade de um segundo caso de uso, denominado “Marcar consulta”, que prevê uma interface para selecionar o dia e o horário da consulta, de acordo com as predefinições de disponibilidade que são gerenciadas pelo educador físico através de uma outra interface, descrita em um terceiro caso de uso “Gerenciar disponibilidade e local da consulta”, que permite definir e editar horários e datas em que está disponível e o local de realização da consulta [25].

Por fim, com a eventual concretização de um cenário bem sucedido de agendamento da consulta, se torna necessário um quarto caso de uso “Enviar notificação de agendamento da consulta” que, como o próprio nome já indica, envia uma notificação de agendado para os e-mails do usuário comum e do educador físico, avisando a data, horário e local da consulta. Este processo é realizado em conjunto com um servidor de e-mail e a aplicação do Google Maps, atores já introduzidos, que possibilitam a utilização de um serviço de correio eletrônico, e a indicação visual do local da consulta através de um mapa [25].

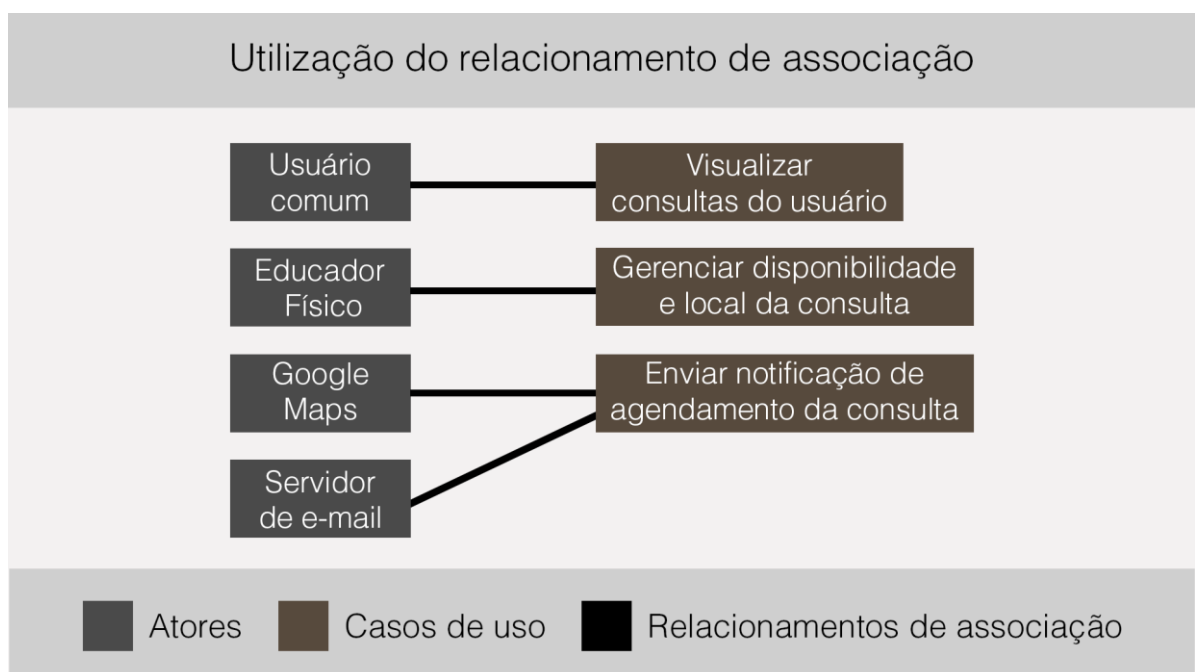
Com a definição de quais atores e casos de uso estarão presentes no sistema, o próximo objetivo torna-se desenvolver uma relação de coexistência entre estes elementos já estabelecidos. Tal relação irá ser concebida e representada através de um diagrama de casos de uso e seus recursos provenientes do padrão da linguagem UML.

Inicialmente, as noções de associação entre atores e casos de uso foram definidas. Prosseguindo com o exemplo anterior referente a consultas, verifica-se a existência de quatro atores – usuário comum, educador físico, servidor de e-mail e Google Maps – e quatro casos de uso – “Visualizar consultas do usuário”, “Marcar consulta”, “Gerenciar disponibilidade e local da consulta” e “Enviar notificação de agendamento da consulta”.

Na posse dessas informações, os envolvidos na análise inicialmente utilizaram o tipo de relacionamento mais simples e explícito para o diagrama de casos de uso, o relacionamento de associação, para indicar quais atores executariam ou teriam participação em cada caso de uso, ou também apontar uma sequência de casos de uso.

Com isso, seguindo a proposta do levantamento de requisitos, o usuário comum foi associado ao caso de uso “Visualizar consultas do usuário”, e o educador físico ao caso de uso “Gerenciar disponibilidade e local da consulta”, pois tratam-se de funcionalidades designadas especificamente à execução destes dois atores. Além disso, os atores “servidor de e-mail” e “Google Maps” foram associados ao caso de uso “Enviar notificação de agendamento da consulta”, com o intuito de indicar que estes mecanismos participam do fluxo de dados e introduzem recursos visuais ao caso de uso em questão. Estas relações estão representadas na Figura 19 abaixo [25].

Figura 19. Esquema indicando a utilização do relacionamento de associação dentro da dinâmica de agendamento e gerenciamento de consultas [25]



É possível notar que na explanação anterior o caso de uso “Marcar consulta” não foi citado, pois ele não estabelece um relacionamento de associação direta com nenhum ator. Sua execução está diretamente atrelada ao caso de uso “Visualizar consultas do usuário”, indicando a existência

de um fluxo alternativo neste cenário. Em outras palavras, o ator usuário comum irá realizar a funcionalidade “Visualizar consultas do usuário”, e a partir disso, poderá opcionalmente executar a funcionalidade “Marcar consulta” [25].

A partir deste momento se torna necessária a utilização de relacionamentos que irão ditar o comportamento entre dois elementos do diagrama de casos de uso, como nesta situação, em que o caso de uso “Marcar consulta” será indicado como uma extensão do caso de uso “Visualizar consultas do usuário” através do *extend*, como mostra a Figura 20 [25].

Figura 20. Esquema indicando a introdução do relacionamento de extensão dentro da dinâmica de agendamento e gerenciamento de consultas [25]

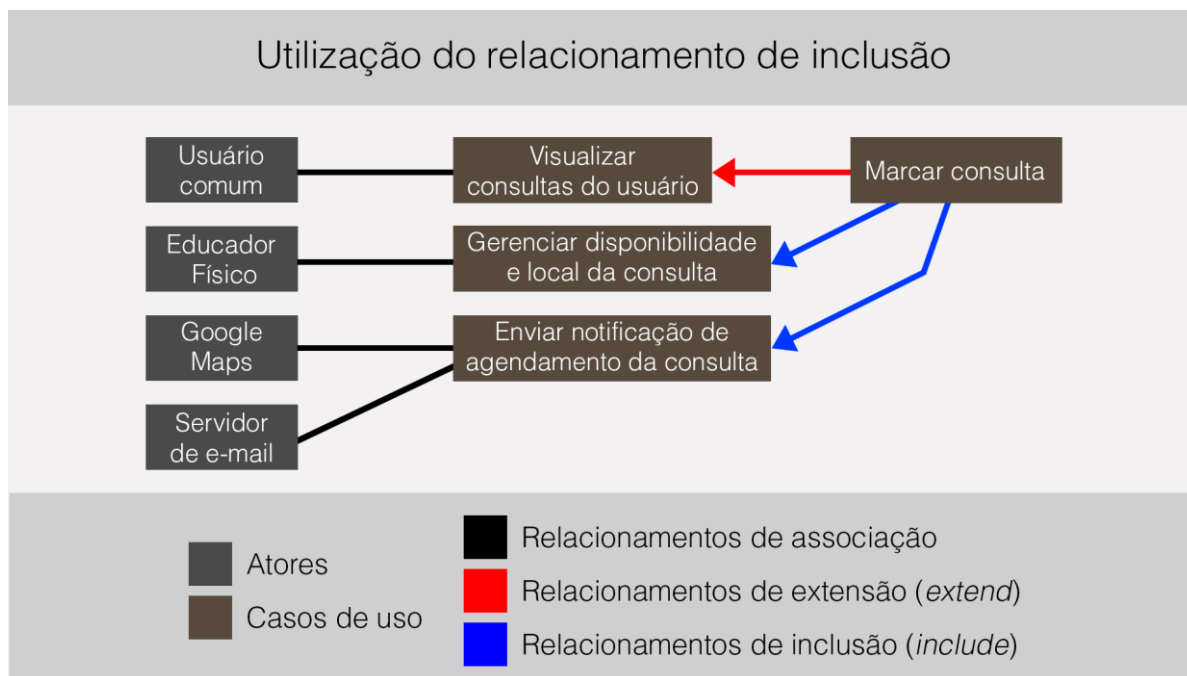


Seguinte à utilização do relacionamento de extensão, ainda é necessário estabelecer relacionamentos que garantam aos casos de uso atuar de maneira conjunta e coesa dentro de um mesmo sistema, de maneira a representar situações em que um caso de uso tenha sua execução atrelada à execução de outro caso de uso.

Dentro desta dinâmica de consultas, existem operações que indicam a obrigatoriedade de outras. Não há como o usuário comum marcar ou remarcar uma consulta sem saber quais horários estarão disponíveis para tal. Portanto, o caso de uso “Marcar consulta” só pode ser executado pelo usuário comum caso o educador físico realize “Gerenciar disponibilidade e local da consulta”. Também é possível notar que após uma consulta ser marcada ou remarcada, é extremamente essencial que uma notificação seja enviada ao usuário comum e ao educador físico. Ou seja, se o caso de uso “Marcar consulta” for realizado, obrigatoriamente o caso de uso “Enviar notificação de agendamento da consulta” também será realizado [25].

Tendo em vista estas relações de obrigatoriedade, utiliza-se uma representação visual através do relacionamento de inclusão entre casos de uso, como descrito na Figura 21 abaixo [25].

Figura 21. Esquema indicando a introdução do relacionamento de inclusão dentro da dinâmica de agendamento e gerenciamento de consultas [25]

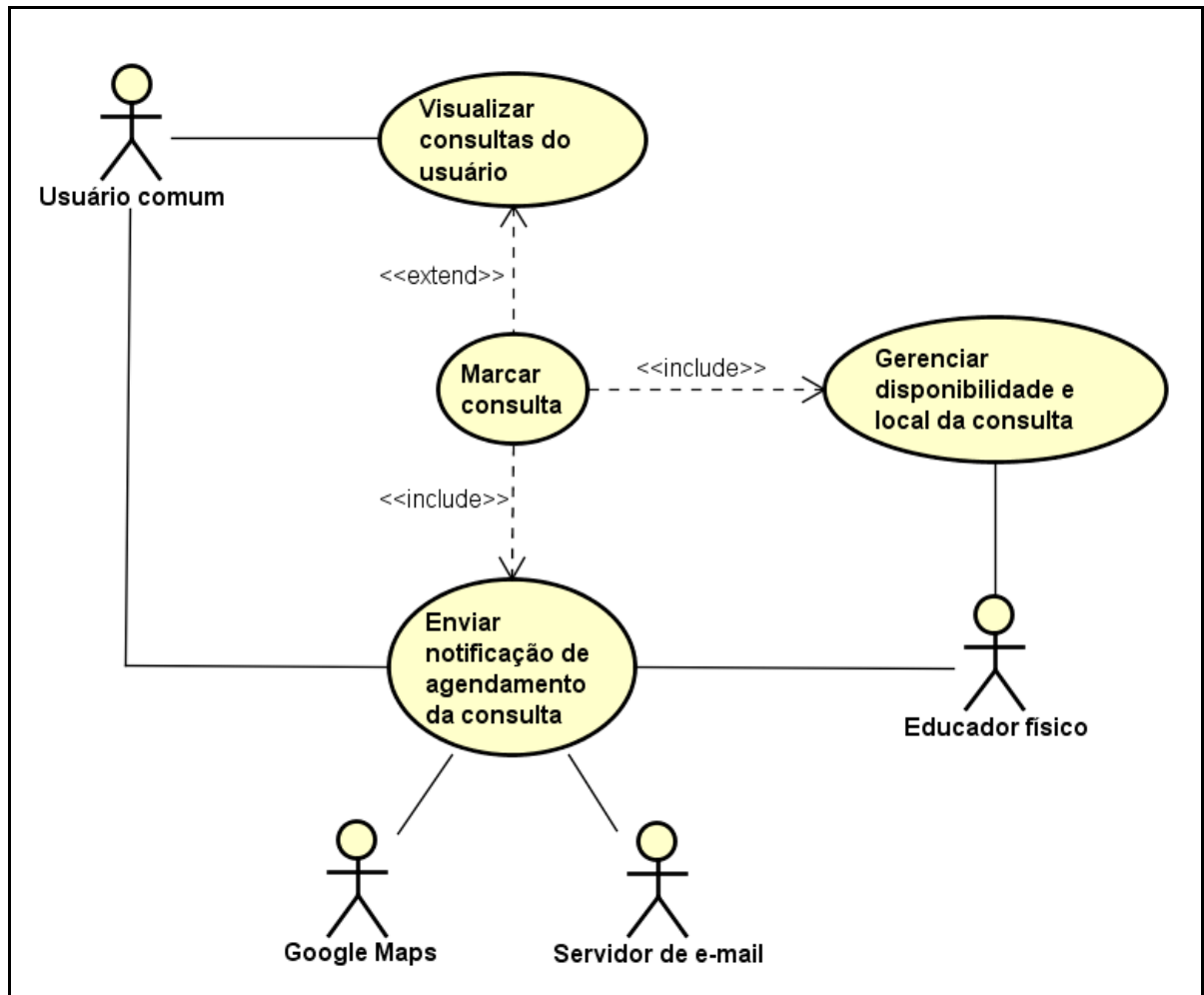


A partir da inserção destes três tipos de relacionamentos foi possível anexar e compreender todos os componentes apresentados em um único sistema, preservando os requisitos e viabilidades propostos e levando em consideração a necessidade de elaborar uma coexistência de processos que fosse prática, acessível e coesa.

Por fim, a diagramação de casos de uso também demonstrou aos envolvidos na análise a versatilidade contida no relacionamento de associação. Além de indicar a execução de funcionalidades por um ator específico, ou a participação de tecnologias em um caso de uso, o relacionamento de associação também indica o ator ao qual está destinado um envio de dados resultante de uma funcionalidade.

Na dinâmica dada, o caso de uso “Enviar notificação de agendamento da consulta” envia dados aos atores usuário comum e educador físico, salientando a necessidade do relacionamento de associação para representar a participação destes atores no processo referido. Com a adição destas noções finais, a diagramação das funcionalidades do requisito referente ao agendamento de consultas foi finalizada, como mostra a Figura 22 abaixo [25].

Figura 22. Representação do diagrama de casos de uso referentes ao agendamento e gerenciamento de consultas, segundo os padrões da UML [25]



Recapitulando os processos apresentados para a concepção do diagrama de casos de uso, pode-se assimilar a representação visual segundo os padrões da UML da figura acima da seguinte maneira: o usuário comum visualiza se tem uma consulta marcada ou não, e após realizar esta função, tem a opção de marcar ou remarcar a consulta, dependendo do cenário em questão. Para facilitar o entendimento, será suposto que o usuário comum irá marcar uma consulta. Só é possível que o usuário comum o faça caso o educador físico gerencie a disponibilidade e o local da consulta.

Caso o usuário comum marque sua consulta com sucesso, é obrigatório que uma notificação de agendamento da consulta seja enviada ao educador físico e ao próprio usuário comum via e-mail. Esse envio é realizado com auxílio de um servidor de e-mail, e integra o Google Maps ao fornecimento de recursos visuais para a gestão de mapas e locais [25].

Este método de realização das coexistências dos casos de uso foi aplicado em todas as instâncias dos requisitos, sendo a dinâmica de agendamento e gerenciamento de consultas apenas uma fração do diagrama de casos de uso final indicado na Figura 23 abaixo [25].

O diagrama de Caso de Uso para o SAGAPS apresenta os seguintes elementos:

- Atores:**
 - Administrador
 - Usuário comum (População)
 - Educador físico
 - Servidor de e-mail
 - Google Maps
- Casos de Uso:**
 - Usuário comum (População):**
 - Visualizar dados do usuário
 - Visualizar consultas do usuário
 - Enviar notificação de agendamento da consulta
 - Visualizar perfil do meu educador físico
 - Visualizar ficha de treinamento
 - Excluir ficha de treinamento
 - Enviar ficha de treinamento por e-mail
 - Educador físico:**
 - Registrar medidas corporais
 - Registrar dobras cutâneas
 - Registrar dados obtidos na consulta
 - Registrar solicitação de acompanhamento de usuário
 - Responder solicitação de acompanhamento de usuário
 - Registrar ficha de treinamento
 - Visualizar ficha de treinamento
 - Excluir ficha de treinamento
 - Enviar ficha de treinamento por e-mail
 - Visualizar suplementos alimentares
 - Adicionar suplemento alimentar
 - Excluir suplemento alimentar
 - Visualizar suplemento alimentar
 - Administrador:**
 - Registrar medidas corporais
 - Registrar dobras cutâneas
 - Registrar dados obtidos na consulta
 - Registrar solicitação de acompanhamento de usuário
 - Responder solicitação de acompanhamento de usuário
 - Registrar ficha de treinamento
 - Visualizar ficha de treinamento
 - Excluir ficha de treinamento
 - Enviar ficha de treinamento por e-mail
 - Visualizar suplementos alimentares
 - Adicionar suplemento alimentar
 - Excluir suplemento alimentar
 - Visualizar suplemento alimentar
 - Servidor de e-mail:**
 - Enviar notificação de agendamento da consulta
 - Google Maps:**
 - Enviar notificação de agendamento da consulta
- Relações e Dependências:**
 - Dependências (dashed arrows):
 - Visualizar dados do usuário para Visualizar consultas do usuário, Enviar notificação de agendamento da consulta, Visualizar perfil do meu educador físico, Visualizar ficha de treinamento, Excluir ficha de treinamento, Enviar ficha de treinamento por e-mail, Visualizar suplementos alimentares, Adicionar suplemento alimentar, Excluir suplemento alimentar, Visualizar suplemento alimentar.
 - Registrar medidas corporais para Registrar dobras cutâneas.
 - Registrar dobras cutâneas para Registrar dados obtidos na consulta.
 - Registrar dados obtidos na consulta para Registrar solicitação de acompanhamento de usuário.
 - Registrar solicitação de acompanhamento de usuário para Responder solicitação de acompanhamento de usuário.
 - Registrar ficha de treinamento para Visualizar ficha de treinamento.
 - Visualizar ficha de treinamento para Excluir ficha de treinamento.
 - Enviar ficha de treinamento por e-mail para Visualizar suplementos alimentares.
 - Visualizar suplementos alimentares para Adicionar suplemento alimentar.
 - Adicionar suplemento alimentar para Excluir suplemento alimentar.
 - Excluir suplemento alimentar para Visualizar suplemento alimentar.
 - Inclusões (dashed arrows with «include»):
 - Enviar notificação de agendamento da consulta para Visualizar consultas do usuário.
 - Enviar notificação de agendamento da consulta para Gerenciar disponibilidade e local da consulta.
 - Gerenciar disponibilidade e local da consulta para Visualizar dados do usuário.
 - Gerenciar disponibilidade e local da consulta para Registrar medidas corporais.
 - Gerenciar disponibilidade e local da consulta para Registrar dobras cutâneas.
 - Gerenciar disponibilidade e local da consulta para Registrar dados obtidos na consulta.
 - Gerenciar disponibilidade e local da consulta para Registrar solicitação de acompanhamento de usuário.
 - Gerenciar disponibilidade e local da consulta para Responder solicitação de acompanhamento de usuário.
 - Gerenciar disponibilidade e local da consulta para Registrar ficha de treinamento.
 - Gerenciar disponibilidade e local da consulta para Visualizar ficha de treinamento.
 - Gerenciar disponibilidade e local da consulta para Excluir ficha de treinamento.
 - Gerenciar disponibilidade e local da consulta para Enviar ficha de treinamento por e-mail.
 - Gerenciar disponibilidade e local da consulta para Visualizar suplementos alimentares.
 - Gerenciar disponibilidade e local da consulta para Adicionar suplemento alimentar.
 - Gerenciar disponibilidade e local da consulta para Excluir suplemento alimentar.
 - Gerenciar disponibilidade e local da consulta para Visualizar suplemento alimentar.
 - Extensões (dashed arrows with «extend»):
 - Visualizar perfil do meu educador físico para Efetuar perfil de treinamento.
 - Visualizar perfil do meu educador físico para Editar perfil de treinamento.
 - Visualizar ficha de treinamento para Adicionar ficha de treinamento.
 - Visualizar ficha de treinamento para Excluir ficha de treinamento.
 - Visualizar ficha de treinamento para Enviar ficha de treinamento por e-mail.
 - Visualizar suplementos alimentares para Adicionar suplemento alimentar.
 - Visualizar suplementos alimentares para Excluir suplemento alimentar.
 - Visualizar suplementos alimentares para Visualizar suplemento alimentar.

O relacionamento de generalização foi o último componente inserido no diagrama de casos com o objetivo de englobar de maneira prática e eficiente as permissões e o comportamento do administrador dentro do sistema. Nesta situação, o administrador herda todas as capacidades dos atores usuário comum e educador físico. Ou seja, para fins de manutenção, o administrador é capaz de realizar todas as funcionalidades que caracterizam interação direta através de interfaces [25].

40

Desta maneira, todos os casos de uso presentes no diagrama passaram a ser estudados e documentados individualmente em uma estrutura que compreende todos os aspectos necessários ao entendimento operacional da funcionalidade, e foi realizada em formato de tabela. Inicialmente, a tabela contém um cabeçalho que contempla informações sobre o contexto do caso de uso [25]. São elas:

- “Nome do Caso de Uso”: nomenclatura que está sendo utilizada àquela funcionalidade no diagrama e no documento;
- “Breve Descrição”: explicação da função geral do caso de uso dentro do sistema, para que fim será executado;
- “Ator Principal”: ator ao qual está destinada a execução do caso de uso, ou aquele que tem mais impacto ou participação na funcionalidade;
- “Pré-Condição”: descrição das condições necessárias para desencadear a execução do caso de uso.

Para melhor compreensão do preenchimento destes dados, a documentação elaborada do caso de uso “Visualizar consultas do usuário” será tomada como exemplo. Na Tabela 1 abaixo está representado o cabeçalho da funcionalidade referida, que contempla seus aspectos gerais e garante ao leitor um entendimento prévio do contexto que será aprofundado em detalhes posteriormente [25].

Tabela 1. Cabeçalho de contextualização do caso de uso “Visualizar consultas do usuário” [25]

Nome do caso de uso: Visualizar consultas do usuário	
Breve Descrição:	Este caso de uso deve ocorrer sempre que o usuário comum selecionar a opção “Minhas consultas” no menu, proporcionando ao usuário a visualização de suas consultas.
Ator Principal:	Usuário comum (População).
Pré-Condição:	Estar logado no sistema, conter um Perfil de Treinamento e estar vinculado a um educador físico.

Abaixo deste cabeçalho, inicia-se um levantamento minucioso sob a forma de lista de todos os fluxos dos cenários do caso de uso em questão. Esta lista contém o fluxo principal do caso de uso, seguido de todos os seus fluxos alternativos. Cada fluxo recebe um nome para identificação do cenário referido, e suas etapas operacionais são numeradas e descritas de acordo com as ações do ator e do sistema de forma alternada, para facilitar a compreensão da interação entre as duas partes. Este levantamento será explanado ainda tomando o caso de uso “Visualizar consultas do usuário” como exemplo, a começar pela abordagem de seu fluxo principal, documentado na Tabela 2 abaixo.

O objetivo nesta estrutura é detalhar ao máximo todos os elementos de interação e recursos visuais presentes nas interfaces dos casos de uso, pois estas serão posteriormente compreendidas e desenvolvidas durante a codificação do sistema [25].

Tabela 2. Documentação do fluxo principal do caso de uso “Visualizar consultas do usuário” [25]

Fluxo Principal	
Ações dos Atores:	Ações do Sistema:
1. Selecionar a opção “Minhas consultas” no menu.	2. Abrir a interface de consultas do usuário. A interface conterá uma mensagem dizendo “Parece que você não tem uma consulta marcada” e a opção “Marcar consulta”.
3. Fim do Caso de Uso.	

O fluxo principal exibido acima foi definido pelos envolvidos na análise através da tentativa de identificar o cenário de interação mais prematura do ator com o sistema. Neste caso, este fluxo foi definido através do cenário em que o usuário comum ainda não marcou nenhuma consulta e visualiza esta situação [25].

Após a definição do fluxo principal, sua descrição é dividida através de ações do ator, seguidas do desencadeamento de ações do sistema. As etapas operacionais do fluxo principal se iniciam a partir do método inicial de execução do caso de uso, que nesta ocasião ocorre ao selecionar a opção “Minhas consultas” no menu do usuário. Em seguida, se descreve o que o sistema irá realizar como consequência da interação do ator, e assim se sucede o processo de operações do fluxo [25].

O encerramento do fluxo principal é marcado pela finalização da tarefa do sistema ou o redirecionamento a uma interface descrita em outro caso de uso, e sua representação é feita através de uma última etapa inserida nas ações dos atores, denominada “Fim do Caso de Uso” [25].

Em seguida, começa a se considerar todos os cenários que correspondem a exceções e casos alternativos ao processo operacional descrito no fluxo principal, e isso ocorre através da documentação dos fluxos alternativos. Os três fluxos alternativos do caso de uso “Visualizar consultas do usuário” estão representados na Tabela 3 abaixo [25].

Tabela 3. Documentação dos fluxos alternativos do caso de uso “Visualizar consultas do usuário” [25]

Fluxo Alternativo A - Marcar consulta	
Ações dos Atores:	Ações do Sistema:
3. Selecionar a opção “Marcar consulta” na	4. Fluxo descrito no Caso de Uso 4.9.

interface.	
Fluxo Alternativo B – Interface do usuário com uma consulta já marcada	
1. Selecionar a opção “Minhas consultas” no menu.	2. Abrir a interface de consultas do usuário. Para o caso do usuário que já possui uma consulta marcada, a interface conterá uma janela com uma mensagem informando os dados da consulta. A mensagem diz “Você tem uma consulta dia X (Ex:25/3 (segunda-feira)) às XX:XXhs (Ex:13:00hs) com o educador físico Z (Ex: Rogerinho) no seguinte local: W (Ex: Posto Durval). Link do Google Maps: goo.gl/xxxx” Abaixo da janela, a interface contém as opções “Remarcar consulta” e “Cancelar consulta”.
Fluxo Alternativo C – Remarcar consulta	
3. Selecionar a opção “Remarcar” na interface.	4. Fluxo descrito no Caso de Uso 4.9.
Fluxo Alternativo D – Cancelar consulta	
3. Selecionar a opção “Cancelar” na interface.	4. Fluxo descrito no Caso de Uso 4.9.

Dada a complexidade da descrição da estrutura acima, os fluxos alternativos serão abordados individualmente.

O fluxo alternativo A, denominado “Marcar consulta”, representa, na verdade, uma alternativa de extensão do fluxo principal, e por isso sua etapa inicial contém o número 3, indicando uma continuidade a partir do fim do fluxo principal. Ou seja, um cenário em que o ator realiza o fluxo principal, e em seguida, executa etapas adicionais, é representado como um fluxo alternativo [25].

Entretanto, é possível notar que a opção “Marcar consulta” indica apenas o direcionamento à uma interface que está presente em outro caso de uso. Este direcionamento resulta na utilização do termo “Fluxo descrito no Caso de Uso X”, onde X indica o subtópico onde o caso de uso que descreve o processo referido está localizado no documento. Neste exemplo, as etapas

operacionais para marcar consulta que vêm após o direcionamento da interface estão descritas no caso de uso “Marcar consulta”, que está localizado no subtópico 4.9 do documento de casos de uso em questão [25].

Já o fluxo alternativo B “Interface do usuário com uma consulta já marcada” contempla um cenário totalmente alternativo e distinto ao fluxo principal. Sua diferenciação encontra-se no fato de que nessa ocasião o usuário comum já possui uma consulta marcada, e assim, a interface se comportará de maneira diferente. Como trata-se de uma versão com processos semelhantes ao próprio fluxo principal, sua etapa inicial contém o número 1, seguida de uma etapa do sistema que se dedica à descrição da interface alternativa [25].

Por fim, os fluxos alternativos C e D, “Remarcar consulta” e “Cancelar consulta”, indicam opções de extensão ao fluxo alternativo B, e por isso seguem a mesma dinâmica de descrição do fluxo alternativo A – iniciando na etapa número 3 – que tem como objetivo demarcar o redirecionamento de interfaces ocasionado por ações adicionais. Nesta situação, essas ações adicionais são a seleção das opções “Remarcar” ou “Cancelar” na interface do fluxo alternativo B por parte do usuário comum [25].

A seguir encontra-se mais um exemplo da estrutura de documentação – desta vez exibida de maneira completa – utilizada para os casos de uso. “Marcar consulta” é o caso de uso referido na Tabela 4 [25].

Tabela 4. Estrutura de documentação do caso de uso “Marcar consulta” [25]

Nome do caso de uso: Marcar consulta	
Breve Descrição:	Este caso de uso deve ocorrer sempre que o usuário comum executar o método “Marcar consulta” na interface de visualização de consultas.
Ator Principal:	Usuário comum (População).
Pré-Condição:	Estar logado no sistema, conter um Perfil de Treinamento e estar vinculado a um educador físico.
Fluxo Principal	
Ações dos Atores:	Ações do Sistema:
1. Selecionar a opção “Marcar consulta” na interface de visualização de consultas.	2. Exibir uma tela com a opção “Selecione a data desejada”, onde o usuário deve selecionar o dia, mês e ano no calendário, que corresponde a um certo dia da semana.
3. Selecionar a data desejada e, em seguida,	4. Exibir uma interface com uma lista dos

a opção “Confirmar”.	horários de consulta disponibilizados pelo educador físico naquele dia da semana. Os intervalos de horário com consulta já marcada estarão bloqueados para seleção.
5. Selecionar o horário que deseja preenchendo uma das caixas, e a opção “Confirmar”.	6. A consulta será computada no banco de dados e enviada uma notificação de consulta ao e-mail do educador físico e do usuário comum. Fluxo descrito no Caso de Uso 4.10.
7. Fim do Caso de Uso.	
Fluxo Alternativo A – Remarcar consulta	
1. Selecionar a opção “Remarcar” quando já tiver uma consulta registrada na interface de consulta do usuário.	2. Exibir uma tela com a opção “Selecione a data desejada”, onde o usuário deve selecionar o dia, mês e ano no calendário, que corresponde a um certo dia da semana.
3. Selecionar a data desejada e, em seguida, a opção “Confirmar”.	4. Exibir uma interface com uma lista dos horários de consulta disponibilizados pelo educador físico naquele dia da semana. Os intervalos de horário com consulta já marcada estarão bloqueados para seleção.
5. Selecionar o horário que deseja preenchendo uma das caixas, e a opção “Confirmar”.	6. A consulta anterior será excluída e uma nova consulta, com o novo horário e data, será computada no banco de dados e enviada uma notificação de consulta ao e-mail do educador físico e do usuário comum. Fluxo descrito no Caso de Uso 4.10.
Fluxo Alternativo B – Cancelar consulta	
1. Selecionar a opção “Cancelar” quando já tiver uma consulta registrada na interface de consulta do usuário.	2. Exibir uma janela confirmando a ação do usuário, dizendo “Tem certeza que deseja cancelar a consulta?”

3. Selecionar a opção “Sim”, confirmando o cancelamento.	4. Excluir consulta do banco de dados e retornar à interface de consulta do usuário.
Fluxo Alternativo C – Abortar cancelamento de consulta	
3. Selecionar a opção “Não”, abortando o cancelamento da consulta.	4. Abortar exclusão da consulta e retornar à interface de consulta do usuário.

2.2.3 Explicação da prototipagem prática de um sistema e seus impactos no ambiente de trabalho cotidiano, explicitando a relação dinâmica entre analista e desenvolvedor

Os exemplos dados no subtópico anterior, referentes à estruturação de documentação de casos de uso, tinham como objetivo a compreensão dos fluxos e do contexto do caso de uso por parte do leitor, e por isso, um dos itens da documentação não estava presente, o protótipo de interface homem-máquina, exibido na Tabela 5 abaixo [25].

Tabela 5. Protótipo de interface homem-máquina da documentação do caso de uso “Visualizar consultas do usuário” [25]

Nome do Caso de Uso: Visualizar consultas do usuário
Protótipo de Interface Homem-Máquina:
<div style="text-align: center;"> <p>Parece que você não tem uma consulta marcada.</p> <p>Marcar consulta</p> </div> <div style="border: 1px solid black; padding: 10px; margin: 20px auto; width: 80%;"> <p>Você tem uma consulta dia 25/3 (segunda-feira) às 13:00hs com o educador físico Rogerinho no seguinte local: Postinho do Durval</p> <p>Link do Google Maps: https://goo.gl/maps/1k3vkRMhLY12</p> </div> <div style="text-align: center; margin-top: 20px;"> <p>Remarcar Cancelar</p> </div>

A prototipagem das interfaces torna-se um processo fundamental na migração das etapas finais de análise – simultâneas ao detalhamento e à construção documentativa de casos de uso – rumo ao desenvolvimento, e para que esse processo ocorresse de maneira progressiva e natural, os envolvidos na análise juntaram-se aos desenvolvedores na criação de uma dinâmica ambiental que fosse capaz de alinhar as visões do sistema de ambas as partes, e a partir disso, resultasse nas primeiras representações visuais das funcionalidades do módulo.

Inicialmente, os envolvidos na análise reuniram-se com os desenvolvedores para uma discussão geral que envolvesse o debate e o entendimento da documentação de casos de uso, para garantir aos desenvolvedores uma visão geral de todas as funcionalidades e contribuísse para a concepção de um padrão de interação que perpetuasse em todos os protótipos. Este padrão consiste no processo visual de como os elementos serão apresentados aos usuários, de que maneira ocorrerão as visualizações, as exclusões, o fluxo de dados, entre outros.

A partir desta definição que colocasse todos os participantes na mesma página, os desenvolvedores se dividiram de maneira que cada um dos integrantes ficasse responsável pela prototipagem de um grupo de funcionalidades correlatas. Assim, ganho de conhecimento substancial de cada desenvolvedor com relação ao seu grupo de casos de uso asseguraria a qualidade individual dos protótipos que começariam a ser criados nesta segunda etapa processual da prototipagem, com a orientação dos envolvidos na análise.

Os primeiros protótipos realizados já demonstravam a importância prática de um levantamento eficiente de casos de uso, onde o alto detalhamento das funcionalidades levou a um alinhamento mais preciso da visão dos envolvidos na análise e dos desenvolvedores, reduzindo a abertura para interpretações distintas do sistema. Neste ponto do processo, a prototipagem já traduzia de maneira coesa as etapas de análise realizadas, e apenas necessitava de alguns poucos ajustes a serem feitos que não foram identificados nas primeiras discussões.

Por fim, na última etapa antes do início da codificação das funcionalidades do sistema, um dos aspectos mais benéficos do contato ambiental entre a análise e o desenvolvimento foi a troca de informações e visões distintas do sistema. Dessa maneira, a falta de experiência dos envolvidos na análise foi suprida pela adaptação de diversos casos de uso através da análise das interfaces dos protótipos.

Ou seja, alinhando-se aos desenvolvedores, os envolvidos na análise puderam compreender o sistema que estavam documentando por um ponto de vista alternativo e mais visual, e essa faceta distinta da análise pode auxiliá-los na correção de erros logísticos e permitiu a flexibilização de casos de uso para aprimorar o sistema.

Desta maneira, as visões de sistema das duas partes envolvidas nesse processo foram alinhadas e estabilizadas para finalizar a realização bem-sucedida da prototipagem de todos os fluxos principais e alternativos do sistema que pudessem ser representados visualmente, como no exemplo da Tabela 6 abaixo, que indica os protótipos dos cenários do caso de uso “Marcar consulta” [25].

Tabela 6. Protótipo de interface homem-máquina da documentação do caso de uso “Marcar consulta” [25]

Nome do Caso de Uso: Visualizar consultas do usuário											
Protótipo de Interface Homem-Máquina:											
<div style="text-align: center; margin-bottom: 20px;"> <h3 style="margin: 0;">Marcar consulta</h3> </div> <div style="text-align: center; margin-bottom: 20px;"> <p>Selecione a data desejada <input style="width: 100px;" type="text" value="dd/mm/aaaa"/></p> </div> <div style="text-align: center; margin-bottom: 20px;"> <p>Confirmar</p> </div> <div style="text-align: center; margin-bottom: 20px;"> <p>Selecione o horário desejado</p> <table style="margin: auto; border-collapse: collapse;"> <tbody> <tr> <td style="padding: 5px 10px;">8:00 às 9:00</td> <td style="text-align: center; padding: 5px 10px;"><input type="radio"/></td> </tr> <tr> <td style="padding: 5px 10px;">9:00 às 10:00</td> <td style="text-align: center; padding: 5px 10px;"><input type="radio"/></td> </tr> <tr> <td style="padding: 5px 10px;">10:00 às 11:00</td> <td style="text-align: center; padding: 5px 10px;"><input type="radio"/></td> </tr> <tr> <td style="padding: 5px 10px;">11:00 às 12:00</td> <td style="text-align: center; padding: 5px 10px;"><input type="radio"/></td> </tr> <tr> <td style="padding: 5px 10px;">13:00 às 14:00</td> <td style="text-align: center; padding: 5px 10px;"><input type="radio"/></td> </tr> </tbody> </table> </div> <div style="text-align: center; margin-bottom: 20px;"> <p>Confirmar</p> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 20px;"> <p>Tem certeza que deseja cancelar a consulta? ×</p> <div style="text-align: right; margin-top: 10px;"> <p>Não Sim</p> </div> </div>		8:00 às 9:00	<input type="radio"/>	9:00 às 10:00	<input type="radio"/>	10:00 às 11:00	<input type="radio"/>	11:00 às 12:00	<input type="radio"/>	13:00 às 14:00	<input type="radio"/>
8:00 às 9:00	<input type="radio"/>										
9:00 às 10:00	<input type="radio"/>										
10:00 às 11:00	<input type="radio"/>										
11:00 às 12:00	<input type="radio"/>										
13:00 às 14:00	<input type="radio"/>										

3 Conclusões e Recomendações

Os objetivos que levaram à confecção deste trabalho compreendiam a especificação e a descrição de todos os processos e métodos que foram fundamentais para a elaboração dos Diagramas de Caso de Uso do Módulo 04: Treinos do Projeto Mais Saúde São João e sua posterior documentação, de maneira a explicar ao leitor as capacidades e modalidades do conhecimento utilizadas para a realização de todas as etapas de análise fundamentais aos resultados finais obtidos.

Para alcançar este objetivo geral, foram compreendidos neste documento as necessidades de introduzir ao leitor os conceitos e conhecimentos pertinentes ao tema de casos de uso e suas etapas decorrentes de análise. Além disso, explicitou-se a potencial concretização de uma série de objetivos específicos que focavam-se em explicar os métodos iniciais de abordagem e análise de requisitos do sistema proposto, a complexidade do processo de concepção, diagramação e documentação dos casos de uso a partir da conversão de requisitos, e por fim, a criação das dinâmicas ambientais entre os envolvidos na análise e desenvolvedores, que permitiram a prototipagem dos casos de uso e a concretização de uma base sólida para a posterior codificação do sistema.

Os resultados obtidos da pesquisa proveniente dos processos redigidos neste trabalho incluem:

- o levantamento com alto nível de detalhe de 12 requisitos funcionais e 6 requisitos não funcionais registrados em um Documento de Solicitação dos Principais Envolvidos, que contém 15 páginas e foi realizado através de 4 versões confirmadas em um período de 1 mês [24];
- a identificação de 6 atores, a concepção, diagramação e documentação de 26 casos de uso, e a realização de 36 protótipos finais registrados em um Documento de Casos de Uso, que contém 47 páginas e foi realizado através de 14 versões confirmadas em um período de 6 meses [25];
- a criação de diversos métodos ambientais próprios da dinâmica dos envolvidos na análise e desenvolvedores, contribuindo para o aprimoramento do sistema e da realização de documentos e etapas de análise.

A listagem dos resultados obtidos reflete um processo satisfatório de análise do sistema, que obteve êxito nas modalidades exigidas e representou a constituição de uma base sólida de entendimento para a concepção de um sistema acessível, dinâmico, eficiente e que se encontra dentro das propostas iniciais.

Acerca de todo o processo redigido neste documento, podem ser avaliados os principais pontos positivos e negativos a serem destacados. Todos os envolvidos no Módulo de Treinos demonstraram grande capacidade de lidar com as adversidades ambientais e técnicas e conseguiram entregar um produto de qualidade acima do esperado, e o comprometimento e a maturidade apresentada por grande parte das equipes excederam as expectativas.

Por outro lado, diversas circunstâncias prejudicaram o andamento do projeto, como a infraestrutura precária para suportar a concepção de um sistema de grande porte e com diversos indivíduos trabalhando de maneira conjunta, a falta de experiência de todos os envolvidos com o ambiente empresarial proposto pela disciplina, e o eventual impacto negativo do comportamento de alguns indivíduos que degradaram a produtividade e o ambiente pacífico de trabalho.

Para a realização de trabalhos futuros com propostas ambientais semelhantes, se sugere a necessidade de atestar o comprometimento e a consciência de todos os indivíduos envolvidos na concepção de um sistema de grande porte e com a necessidade de coordenação de alto nível entre equipes. Caso esta capacidade não seja palpável, talvez seja considerada interessante a implantação de uma dinâmica com um nível de integração menor e com uma disposição de atividades mais distribuída entre funções para todos os períodos do ano.

Por fim, com relação a temas referentes a casos de uso e processos de análise, é sugerido uma maior participação dos desenvolvedores nas etapas iniciais de análise para alcançar benefícios provenientes do debate entre visões distintas do sistema. Dessa maneira, todos os envolvidos no sistema adquirem maior propriedade com relação ao objeto de trabalho, e ao mesmo tempo, têm a capacidade de maior engajamento e interesse pela disciplina.

4 Referências Bibliográficas

- [1] MINISTÉRIO DA EDUCAÇÃO. **Instituto Federal de São Paulo**. 20XX. Disponível em: <<https://www.ifsp.edu.br/institucional>>. Acesso em: 14 ago. 2018.
- [2] IBGE. **IBGE | Brasil em Síntese | São Paulo | São João da Boa Vista | Panorama**. 201X. Disponível em: <<https://cidades.ibge.gov.br/brasil/sp/sao-joao-da-boa-vista/panorama>>. Acesso em: 14 ago. 2018.
- [3] ROMANO, B. L. **Prática de Desenvolvimento de Sistemas (PDS) (Técnico Integrado em Informática) - IFSP-SBV - Plano de Ensino**. 2018. Disponível em: <[https://sbv.ifsp.edu.br/wiki/index.php/Pr%C3%A1tica_de_Developolvimento_de_Sistemas_\(PDS\)_\(T%C3%A9cnico_Integrado_em_Inform%C3%A1tica\)](https://sbv.ifsp.edu.br/wiki/index.php/Pr%C3%A1tica_de_Developolvimento_de_Sistemas_(PDS)_(T%C3%A9cnico_Integrado_em_Inform%C3%A1tica))>. Acesso em: 29 ago. 2018.
- [4] ROMANO, B. L. **Macrorequisitos dos Módulos do Projeto Mais Saúde São João**. 2018. Disponível em: <<https://sites.google.com/site/blromano/disciplinas/pds2014>>. Acesso em: 29 ago. 2018.
- [5] Equipe Mais Saúde São João. **Termo de Abertura do projeto Mais Saúde São João**. 2018. Disponível em: <<https://svn.sbv.ifsp.edu.br/svn/pds2018vespertino/trunk/documentacao/comum/TermoAberturaProjeto.doc>>. Acesso em: 29 ago. 2018.
- [6] ROMANO, B. L. **Prática de Desenvolvimento de Sistemas (PDS) – Roteiro da Disciplina / Projeto**. 2018. Disponível em: <<https://sites.google.com/site/blromano/disciplinas/pds2014>>. Acesso em: 29 ago. 2018.
- [7] Equipe Mais Saúde São João. **Documento de Visão do Módulo de Treinos do projeto Mais Saúde São João**. 2018. Disponível em: <<https://svn.sbv.ifsp.edu.br/svn/pds2018vespertino/trunk/documentacao/mod04/Documentos%20finalizados/DocumentoVisão.doc>>. Acesso em: 29 ago. 2018.
- [8] COSTA, D. E. **Importância Da Análise De Sistemas Para o Desenvolvimento De Softwares**. 201X. Disponível em: <<http://www.essentiaeditora.iff.edu.br/index.php/conepe/article/viewFile/7063/4733>>. Acesso em: 29 ago. 2018.
- [9] ROMANO, B. L. **Aula 04: UML – Casos de Uso e Estimativa de Esforços por Pontos de Caso de Uso**. 2017. Disponível em: <<https://sites.google.com/site/blromano/disciplinas/pds2014>>. Acesso em: 30 ago. 2018.

- [10] GAEA CONSULTING. **Ciclo de Vida do desenvolvimento de softwares: o que eu preciso saber?** 201X. Disponível em: <<https://gaea.com.br/ciclo-de-vida-do-desenvolvimento-de-softwares-o-que-eu-preciso-saber/>>. Acesso em: 30 set. 2018.
- [11] DEVMEDIA. **Ciclos de Vida de Software.** 2011. Disponível em: <<https://www.devmedia.com.br/ciclos-de-vida-do-software/21099>>. Acesso em: 30 set. 2018.
- [12] LEITE, J. C. **O Processo de Desenvolvimento de Software.** 2000. Disponível em: <<https://www.dimap.ufrn.br/~jair/ES/c2.html>>. Acesso em: 30 set. 2018.
- [13] INFOJOBS. **Analista de Sistemas.** 20XX. Disponível em: <https://www.infojobs.com.br/artigos/Analista_de_Sistemas__2437.aspx>. Acesso em: 30 set. 2018.
- [14] MEDEIROS, H. **Introdução a Requisitos de Software.** 2013. Disponível em: <<https://www.devmedia.com.br/introducao-a-requisitos-de-software/29580>>. Acesso em: 01 out. 2018.
- [15] VIEGAS, G. **Introdução a UML: Unified Modeling Language ou Linguagem de Modelagem Unificada.** 2009. Disponível em: <<https://www.devmedia.com.br/introducao-a-uml/6928>>. Acesso em: 15 out. 2018.
- [16] RIBEIRO, L. **Diagramas de Caso de Uso: O que é UML?** 2012. Disponível em: <<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>>. Acesso em: 15 out. 2018.
- [17] IBM. **IBM Knowledge Center – Relacionamentos em Diagramas de Caso de Uso.** 20XX. Disponível em: <https://www.ibm.com/support/knowledgecenter/pt-br/SS5JSH_9.1.2/com.ibm.xtools.modeler.doc/topics/crelsme_ucd.html>. Acesso em: 21 out. 2018.
- [18] FIGUEIREDO, E. **Relacionamentos do Diagrama de Casos de Uso.** 20XX. Disponível em: <https://homepages.dcc.ufmg.br/~figueiredo/disciplinas/aulas/uml-casos-de-uso-relacionamentos_v01.pdf>. Acesso em: 21 out. 2018.
- [19] VENTURA, P. **Caso de Uso – Include, Extend e Generalização.** 2014. Disponível em: <<https://www.ateomomento.com.br/caso-de-uso-include-extend-e-generalizacao/>>. Acesso em: 21 out. 2018.
- [20] RUP. **Diretrizes: Generalização de Ator no Modelo de Casos de Uso de Negócios.** 20XX. Disponível em: <http://www.funpar.ufpr.br:8080/rup/process/modguide/md_bacg.htm>. Acesso em: 21 out. 2018.
- [21] FIGUEIREDO, E. **Documentando Casos de Uso com Cenários.** 20XX. Disponível em: <https://homepages.dcc.ufmg.br/~figueiredo/disciplinas/aulas/uml-casos-de-uso-cenarios_v01.pdf>. Acesso em: 21 out. 2018.

[22] VENTURA, P. **Caso de Uso – Fluxo Principal**. 2016. Disponível em: <<https://www.ateomomento.com.br/caso-de-uso-fluxo-principal/>>. Acesso em: 21 out. 2018.

[23] THIAGONASC.COM. **A importância dos protótipos no desenvolvimento de sistemas**. 2013. Disponível em: <<http://thiagonasc.com/desenvolvimento-web/a-importancia-dos-prototipos-no-desenvolvimento-de-sistemas>>. Acesso em: 21 out. 2018.

[24] Equipe Mais Saúde São João. **Solicitação dos Principais Envolvidos do Módulo de Treinos do Projeto Mais Saúde São João**. 2018. Disponível em: <<https://svn.sbv.ifsp.edu.br/svn/pds2018vespertino/trunk/documentacao/mod04/Documentos%20finalizados/DocumentoRequisitos.doc>>. Acesso em: 23 out. 2018.

[25] Equipe Mais Saúde São João. **Modelo de Casos de Uso do Módulo de Treinos do Projeto Mais Saúde São João**. 2018. Disponível em: <<https://svn.sbv.ifsp.edu.br/svn/pds2018vespertino/trunk/documentacao/mod04/Documentos%20em%20atualiza%C3%A7%C3%A3o/CasodeUsoMód04>>. Acesso em: 28 out. 2018.