UNIVERSIDADE DE CAXIAS DO SUL
ÁREA DO CONHECIMENTO DE CIÊNCIAS EXATAS E ENGENHARIAS

LUCCA CASANOVA SGARIA

DESENVOLVIMENTO DE PROTÓTIPO DE SOFTWARE PARA LABORATÓRIO CIENTÍFICO

BENTO GONÇALVES

2022

LUCCA CASANOVA SGARIA

DESENVOLVIMENTO DE PROTÓTIPO DE SOFTWARE PARA LABORATÓRIO CIENTÍFICO

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação na Área do Conhecimento de Ciências Exatas e Engenharias da Universidade de Caxias do Sul.

Orientador: Prof. Me. Alexandre Erasmo Krohn Nascimento

BENTO GONÇALVES

2022

LUCCA CASANOVA SGARIA

DESENVOLVIMENTO DE PROTÓTIPO DE SOFTWARE PARA LABORATÓRIO CIENTÍFICO

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação na Área do Conhecimento de Ciências Exatas e Engenharias da Universidade de Caxias do Sul.

Aprovado em 29/11/2022

BANCA EXAMINADORA

_____

Prof. Me. Alexandre Erasmo Krohn Nascimento
Universidade de Caxias do Sul - UCS

_____

Prof. Dr. Daniel Luis Notari
Universidade de Caxias do Sul - UCS

_____

Profª. Drª. Scheila de Ávila e Silva
Universidade de Caxias do Sul - UCS

RESUMO

Os pesquisadores do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos, da Universidade de Caxias do Sul, registram as informações da execução de cada experimento químico em cadernos físicos. Este procedimento provoca dificuldades relacionados ao armazenamento dos registros e ao compartilhamento e análise dos resultados. Deste modo, este trabalho tem o objetivo de produzir um protótipo funcional de um *software* que possibilite a digitalização dos processos do laboratório. Para isto, foi realizado a modelagem do *software* baseando-se na metodologia ICONIX, sendo gerados artefatos da linguagem *Unified Modeling Language* (UML), como diagramas de classes, de casos de uso e de robustez. Além disso, realizou-se um estudo a respeito do desenvolvimento deste sistema utilizando o *framework* Quarkus. Por fim, o protótipo funcional do *software* foi implementado com a linguagem de programação Java e com o Quarkus na parte do *back-end* e com a biblioteca React no *front-end*.

Palavras-chave: Modelagem de *software*. ICONIX. UML. Quarkus.

ABSTRACT

The researchers of Laboratorio de Biotecnologia de Produtos Naturais e Sintéticos, from Universidade de Caxias do Sul, record the information of each chemistry experiment execution using physical notebooks. This procedure generates harsdships related to the storage of records and to the sharing and analysis of results. Threfore, this paperwork has the goal to produce a functional prototype of a software that allows the lab's processses to be digitalized. To achieve this, the software modelling was made based on ICONIX methodology, with UML artifacts such as class, use case and robustness diagrams being generated. Furthermore, a study was made analysing the development of this software using the Quarkus framework. Finally, the functional prototype was implemented with the programming language Java and with Quarkus in the back-end and with the React library at the front-end.

Keywords: Sofware modelling. ICONIX. UML. Quarkus.

LISTA DE FIGURAS

Figura 1 – Fluxos e artefatos do ICONIX 14
Figura 2 – Anotações de um experimento 21
Figura 3 – Requisitos Funcionais - Usuário 23
Figura 4 – Requisitos Funcionais - Grupo 23
Figura 5 – Requisitos Funcionais - Projeto 23
Figura 6 – Requisitos Funcionais - Composto 24
Figura 7 – Requisitos Funcionais - Experimento 24
Figura 8 – Requisitos Não-Funcionais 24
Figura 9 – Modelo de Domínio 25
Figura 10 – Diagrama de Classes 26
Figura 11 – Classes e suas descrições 27
Figura 12 – Diagrama de Casos de Uso 28
Figura 13 – Casos de Uso - Usuário 29
Figura 14 – Interface Gráfica UC-01 Realizar *Login* 30
Figura 15 – Diagrama de Robustez UC-01 Realizar *Login* 31
Figura 16 – Recorte da Interface Gráfica UC-02 Registrar Usuário 32
Figura 17 – Diagrama de Robustez UC-02 Registrar Usuário 33
Figura 18 – Interface Gráfica UC-03 Finalizar registro de Usuário 34
Figura 19 – Diagrama de Robustez UC-03 Finalizar registro de Usuário 35
Figura 20 – Interface Gráfica UC-04 Listar Usuários 36
Figura 21 – Diagrama de Robustez UC-04 Listar Usuários 37
Figura 22 – Interface Gráfica UC-05 Filtrar Usuários 37
Figura 23 – Recorte da Interface Gráfica UC-06 Gerenciar Usuário 38
Figura 24 – Diagrama de Robustez UC-06 Gerenciar Usuário 39
Figura 25 – Recorte da Interface Gráfica UC-07 Gerenciar o próprio Usuário 40
Figura 26 – Diagrama de Robustez UC-07 Gerenciar o próprio Usuário 41
Figura 27 – Interface Gráfica UC-08 Visualizar perfil de Usuário 42
Figura 28 – Diagrama de Robustez UC-08 Visualizar perfil de Usuário 43
Figura 29 – Casos de Uso - Grupo 43
Figura 30 – Interface Gráfica UC-09 Listar Grupos 44
Figura 31 – Diagrama de Robustez UC-09 Listar Grupos 45
Figura 32 – Interface Gráfica UC-10 Filtrar Grupos 45
Figura 33 – Recorte da Interface Gráfica 01 UC-11 Gerenciar Grupo 46
Figura 34 – Recorte da Interface Gráfica 02 UC-11 Gerenciar Grupo 47
Figura 35 – Diagrama de Robustez 01 UC-11 Gerenciar Grupo 49
Figura 36 – Casos de Uso - Composto 49

Figura 37 – Interface Gráfica UC-15 Listar Compostos 50
Figura 38 – Diagrama de Robustez UC-15 Listar Compostos 51
Figura 39 – Interface Gráfica UC-16 Filtrar Compostos 51
Figura 40 – Recorte da Interface Gráfica UC-17 Gerenciar Composto 52
Figura 41 – Diagrama de Robustez UC-17 Gerenciar Composto 54
Figura 42 – Casos de Uso - Projeto 54
Figura 43 – Interface Gráfica UC-12 Listar Projetos 55
Figura 44 – Diagrama de Robustez UC-12 Listar Projetos 56
Figura 45 – Interface Gráfica UC-13 Filtrar Projetos 56
Figura 46 – Recorte da Interface Gráfica 01 UC-14 Gerenciar Projeto 58
Figura 47 – Recorte da Interface Gráfica 02 UC-14 Gerenciar Projeto 58
Figura 48 – Diagrama de Robustez UC-14 Gerenciar Projeto 61
Figura 49 – Casos de Uso - Experimento 61
Figura 50 – Interface Gráfica UC-18 Listar Experimentos 62
Figura 51 – Diagrama de Robustez UC-18 Listar Experimentos 63
Figura 52 – Interface Gráfica UC-19 Filtrar Experimentos 63
Figura 53 – Diagrama de Robustez UC-19 Filtrar Experimentos 64
Figura 54 – Recorte da Interface Gráfica UC-20 Gerenciar Experimento 65
Figura 55 – Diagrama de Robustez UC-20 Gerenciar Experimento 67
Figura 56 – Recorte da Interface Gráfica UC-21 Gerenciar Comentários de Experimento 68
Figura 57 – Diagrama de Robustez UC-21 Gerenciar Comentários de Experimento . . . 70
Figura 58 – Recorte da Interface Gráfica UC-22 Gerenciar Anexos de Experimento . . . 70
Figura 59 – Diagrama de Robustez UC-22 Gerenciar Anexos de Experimento 72
Figura 60 – Recorte da Interface Gráfica UC-23 Gerenciar Linhas de Experimento . . . 72
Figura 61 – Diagrama de Robustez UC-23 Gerenciar Linhas de Experimento 73
Figura 62 – Interface Gráfica UC-24 Executar um Experimento 74
Figura 63 – Diagrama de Robustez UC-24 Executar um Experimento 76
Figura 64 – Interface Gráfica UC-25 Exportar o conteúdo de um Experimento 76

LISTA DE QUADROS

Quadro 1 – Detalhamento UC-01 Realizar *Login* 31
Quadro 2 – Detalhamento UC-02 Registrar Usuário 32
Quadro 3 – Detalhamento UC-03 Finalizar registro de Usuário 35
Quadro 4 – Detalhamento UC-04 Listar Usuários 36
Quadro 5 – Detalhamento UC-05 Filtrar Usuários 38
Quadro 6 – Detalhamento UC-06 Gerenciar Usuário 39
Quadro 7 – Detalhamento UC-07 Gerenciar o próprio Usuário 41
Quadro 8 – Detalhamento UC-08 Visualizar perfil de Usuário 42
Quadro 9 – Detalhamento UC-09 Listar Grupos 44
Quadro 10 – Detalhamento UC-10 Filtrar Grupos 46
Quadro 11 – Detalhamento UC-11 Gerenciar Grupo 48
Quadro 12 – Detalhamento UC-15 Listar Compostos 50
Quadro 13 – Detalhamento UC-16 Filtrar Compostos 52
Quadro 14 – Detalhamento UC-17 Gerenciar Composto 53
Quadro 15 – Detalhamento UC-12 Listar Projetos 55
Quadro 16 – Detalhamento UC-13 Filtrar Projetos 57
Quadro 17 – Detalhamento UC-14 Gerenciar Projeto 60
Quadro 18 – Detalhamento UC-18 Listar Experimentos 62
Quadro 19 – Detalhamento UC-19 Filtrar Experimentos 64
Quadro 20 – Detalhamento UC-20 Gerenciar Experimento 66
Quadro 21 – Detalhamento UC-21 Gerenciar Comentários de Experimento 69
Quadro 22 – Detalhamento UC-22 Gerenciar Anexos de Experimento 71
Quadro 23 – Detalhamento UC-23 Gerenciar Linhas de Experimento 73
Quadro 24 – Detalhamento UC-24 Executar um Experimento 75
Quadro 25 – Detalhamento UC-25 Exportar o conteúdo de um Experimento 77

LISTA DE ABREVIATURAS E SIGLAS

AOT *Ahead-Of-Time*

API *Application Programming Interface*

ARPANET *Advanced Research Projects Agency Network*

AWS Amazon Web Services

ELN *Electronic Laboratory Notebooks*

HTML *HyperText Markup Language*

HTTP *Hypertext Transfer Protocol*

JAR *Java ARchive*

JBOSS *JavaBeans Open Source Software*

JRE Ambiente de Tempo de Execução Java

Java EE *Java Enterprise Edition*

SaaS *Software as a service*

TCP/IP *Transfer Control Protocol/Internetwork Protocol*

UCS Universidade de Caxias do Sul

UML *Unified Modeling Language*

URL *Uniform Resource Locator*

W3C *World Wide Web Consortium*

WWW *World Wide Web*

SUMÁRIO

1 INTRODUÇÃO 11
1.1 OBJETIVO GERAL 12
1.1.1 OBJETIVOS ESPECÍFICOS 12
1.2 ESTRUTURA DO TRABALHO 12

2 METODOLOGIA 13

3 TECNOLOGIAS DE DESENVOLVIMENTO 15
3.1 DESENVOLVIMENTO WEB 15
3.1.1 Internet 15
3.1.2 Modelo cliente-servidor 16
3.2 Quarkus 17
3.2.1 Surgimento 17
3.2.2 Funcionalidades 18

4 PROJETO 21
4.1 REQUISITOS 22
4.2 MODELO DE DOMÍNIO 24
4.3 DIAGRAMA DE CLASSES 25
4.4 DIAGRAMA DE CASOS DE USO 27
4.5 DETALHAMENTO DE CASOS DE USO 29
4.5.1 DETALHAMENTO DE CASOS DE USO - USUÁRIO 29
4.5.1.1 Caso de Uso UC-01 Realizar *Login* 29
4.5.1.2 Caso de Uso UC-02 Registrar Usuário 31
4.5.1.3 Caso de Uso UC-03 Finalizar registro de Usuário 33
4.5.1.4 Caso de Uso UC-04 Listar Usuários 36
4.5.1.5 Caso de Uso UC-05 Filtrar Usuários 37
4.5.1.6 Caso de Uso UC-06 Gerenciar Usuário 38
4.5.1.7 Caso de Uso UC-07 Gerenciar o próprio Usuário 40
4.5.1.8 Caso de Uso UC-08 Visualizar perfil de Usuário 42
4.5.2 DETALHAMENTO DE CASOS DE USO - GRUPO 43
4.5.2.1 Caso de Uso UC-09 Listar Grupos 43
4.5.2.2 Caso de Uso UC-10 Filtrar Grupos 45
4.5.2.3 Caso de Uso UC-11 Gerenciar Grupo 46
4.5.3 DETALHAMENTO DE CASOS DE USO - COMPOSTO 49
4.5.3.1 Caso de Uso UC-15 Listar Compostos 49

4.5.3.2 Caso de Uso UC-16 Filtrar Compostos 51
4.5.3.3 Caso de Uso UC-17 Gerenciar Composto 52
4.5.4 DETALHAMENTO DE CASOS DE USO - PROJETO 54
4.5.4.1 Caso de Uso UC-12 Listar Projetos 54
4.5.4.2 Caso de Uso UC-13 Filtrar Projetos 56
4.5.4.3 Caso de Uso UC-14 Gerenciar Projeto 57
4.5.5 DETALHAMENTO DE CASOS DE USO - EXPERIMENTO 61
4.5.5.1 Caso de Uso UC-18 Listar Experimentos 61
4.5.5.2 Caso de Uso UC-19 Filtrar Experimentos 63
4.5.5.3 Caso de Uso UC-20 Gerenciar Experimento 64
4.5.5.4 Caso de Uso UC-21 Gerenciar Comentários de Experimento 67
4.5.5.5 Caso de Uso UC-22 Gerenciar Anexos de Experimento 70
4.5.5.6 Caso de Uso UC-23 Gerenciar Linhas de Experimento 72
4.5.5.7 Caso de Uso UC-24 Executar um Experimento 74
4.5.5.8 Caso de Uso UC-25 Exportar o conteúdo de um Experimento 76

5 CONSIDERAÇÕES FINAIS 78

REFERÊNCIAS 80

1 INTRODUÇÃO

A utilização de *softwares* está presente em diversas áreas do cotidiano do ser-humano, como na comunicação, no entretenimento e no comércio. Além disso, o emprego de *softwares* também ocorre em ambientes profissionais e acadêmicos. É possível caracterizá-los entre aplicações *web*, de *desktop* ou móveis, por exemplo. Para demonstrar a escala da relevância destas ferramentas digitais, pode-se citar o levantamento de que, em janeiro de 2022, 4.95 bilhões de pessoas utilizaram a internet (KEMP, 2022), responsável pela entrega de aplicações *web* ao usuário final.

O registro de processos realizados por integrantes de laboratórios associados a pesquisas cientificas de áreas como a química, por exemplo, também podem ser realizadas utilizando *softwares*. Mais especificamente, os *Electronic Laboratory Notebooks*s (ELNs), aplicações oferecidas principalmente a partir da década de 1980, possibilitam a gravação, acesso e compartilhamento de registros e dados de laboratórios a partir de um meio digital. Este recurso permite que os integrantes infiram significados e resultados de forma rápida e ajuda a facilitar a reprodutibilidade dos experimentos (HIGGINS *et al.*, 2022).

A maioria destes *softwares* são oferecidos de forma comercial e disponibilizados via assinaturas, através do modelo de negócios de *Software as a service* (SaaS) (HIGGINS *et al.*, 2022). Os recursos financeiros necessários para o acesso à estas aplicações em muitos casos impossibilitam o seu uso por pesquisadores de laboratórios de pequeno e médio porte.

Atualmente, os pesquisadores do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos, da Universidade de Caxias do Sul (UCS), utilizam cadernos físicos para registrar as informações dos experimentos químicos executados. A realização deste processo de forma manual acarreta em variadas dificuldades e problemas.

Entre estas dificuldades está a necessidade de manter inúmeros cadernos físicos, uma vez que os experimentos se multiplicam com o passar do tempo. Além disso, o acesso e a consulta aos registros caracterizam-se como processos lentos, dado ao fato de que nenhum tipo de filtro de pesquisa é empregado. Do mesmo modo, o uso de cadernos físicos também prejudica o compartilhamentos entre pesquisadores das informações e dos resultados dos experimentos.

Dentro desse contexto, no presente trabalho foi desenvolvido o protótipo de um *software* que busca solucionar os empecilhos citados, disponibilizando uma aplicação *web*, de fácil acesso via Internet, para realização das atividades dos pesquisadores do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos da UCS. Além disso, o *software* proposto permite a existência de um controle geral de todos os processos e integrantes do laboratório por parte do Prof. Dr. Thiago Barcellos da Silva e outros supervisores.

Por fim, destaca-se que, neste caso, a utilização da aplicação implementada neste traba-

lho é vantajosa em relação aos ELNs disponibilizados pelo mercado. Isto é justificado devido a alguns fatos, como a gratuidade do *software* e o desenvolvimento das interfaces do mesmo totalmente na língua portuguesa. Assim como, ao fato de que as funcionalidades foram desenhadas considerando as necessidades dos integrantes deste laboratório.

1.1 OBJETIVO GERAL

Construir o protótipo de um *software* para registros de experimentos com a finalidade de auxiliar integrantes do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos da UCS.

1.1.1 OBJETIVOS ESPECÍFICOS

1. Realizar a modelagem do *software*.
2. Analisar as tecnologias possíveis para o desenvolvimento da aplicação.
3. Implementar um protótipo funcional do *software*.

1.2 ESTRUTURA DO TRABALHO

O presente trabalho está organizado da seguinte forma:

- O Capítulo 2 apresenta os conceitos do processo denominado ICONIX, empregado na etapa de modelagem do *software*. É realizado uma explicação resumida deste método e das etapas de modelagem que o mesmo propõe.
- No Capítulo 3 as ferramentas tecnológicas associadas ao desenvolvimento do software são descritas. Inicialmente é realizado uma contextualização ligada ao desenvolvimento *web* e são levantados fatos do ambiente do *front-end* e do *back-end*. Por fim, o *framework* Quarkus é apresentado, sendo destacado seu conceito, seu propósito, suas características e suas vantagens e desvantagens.
- No Capítulo 4 todas as etapas da modelagem do protótipo do *software* são realizadas. Em um primeiro momento, as principais características do projeto são exibidas. Após isso, ocorrem o levantamento de requisitos e as etapas relacionadas ao processo ICONIX, entre elas a produção de diagramas de domínio, de classes e de casos de usos, além do detalhamento de todos casos de uso definidos.
- O Capítulo 5 apresenta as considerações finais do trabalho e analisa o processo de desenvolvimento do protótipo do *software*, comparando o planejamento com o cenário que ocorreu na prática.

2 METODOLOGIA

Como citado anteriormente, este trabalho possui a premissa de construir um protótipo funcional de um *software* que digitalize o registro das informações dos experimentos executados por pesquisadores do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos da UCS.

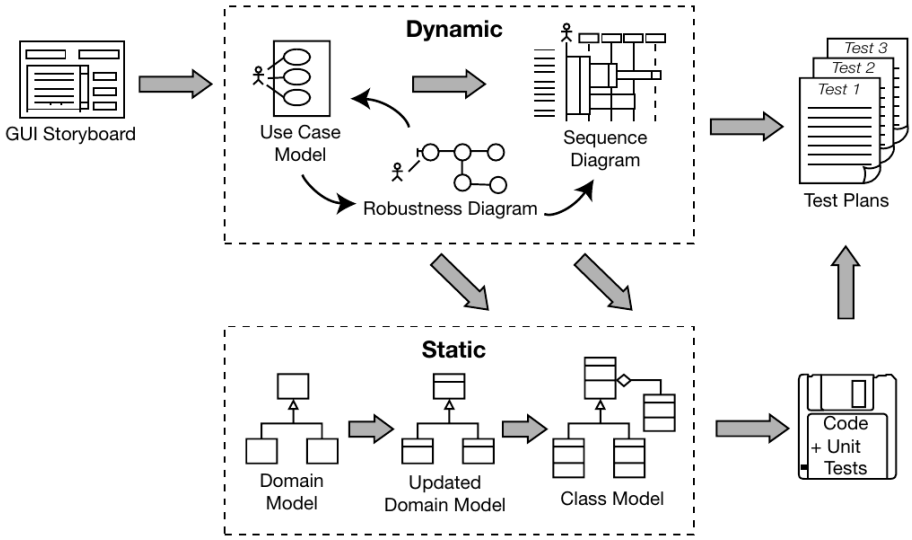
Com o intuito de atingir este objetivo, foi desenvolvido uma aplicação *web*. Nesta aplicação foi necessário o emprego de ferramentas associados ao *front-end* e ao *back-end*. Com relação ao *back-end*, o *framework* Quarkus, da linguagem Java, foi analisado como possível ferramenta a ser utilizada. Após conclusão da viabilidade do mesmo, o *framework* foi utilizado como ferramenta central do *back-end* do protótipo. No âmbito do *front-end*, o instrumento de desenvolvimento empregado foi o React, definido a partir dos resultados obtidos por (MARTINOTTO, 2021).

As etapas da metodologia ICONIX serão utilizadas para o processo de modelagem e desenvolvimento da aplicação. O ICONIX dividi-se em dois fluxos, sendo eles o estático e o dinâmico. O fluxo estático é responsável pela geração dos artefatos ligados aos dados do sistema, sendo nele produzido o modelo de domínio e o diagrama de classes. Por outro lado, o fluxo dinâmico caracteriza-se pela definição de processos associados a interação dos usuários com o software, gerando então os diagramas de casos de uso, de robustez e de sequência. Estes diagrama são representados através da linguagem UML (ROSENBERG; STEPHENS, 2007).

Ainda pode-se destacar o fato de que os fluxos podem ser desenvolvidos de forma paralela e que os modelos do processo estático são refinados conforme os artefatos do fluxo dinâmico são construídos (ROSENBERG; STEPHENS, 2007). Deste modo, o ICONIX foi escolhido, pois proporciona um processo prático e iterativo, além de produzir todos os artefatos necessários para a etapa da implementação, facilitando assim, o trabalho do desenvolvedor.

A Figura 1 apresenta os fluxos da metodologia ICONIX e os artefatos gerados em cada um deles.

Figura 1 – Fluxos e artefatos do ICONIX



Fonte: (ROSENBERG; STEPHENS, 2007)

3 TECNOLOGIAS DE DESENVOLVIMENTO

Este capítulo apresentará uma descrição das tecnologias relacionadas a implementação do projeto. Será realizado uma contextualização a respeito do Internet e do Desenvolvimento *web* na Seção 3.1. Na Seção 3.2, o *framework* Quarkus será conceituado e suas principais características e funcionalidades serão apresentadas para justificar seu uso no projeto do trabalho.

3.1 DESENVOLVIMENTO WEB

Nesta seção serão descritos conceitos relacionados desenvolvimento de aplicações para *web*. Inicialmente, na Seção 3.1.1, aspectos da Internet serão apresentados. Posteriormente, na Seção 3.1.2, o conceito do modelo cliente-servidor, muito presente na arquitetura atual da *web*, será descrito.

3.1.1 Internet

Atualmente a Internet está presente no cotidiano da vida de grande da população mundial. O relatório apresentado por Kemp (2022) divulgou que o número de usuários globais da internet em janeiro de 2022 foi de 4.95 bilhões, o que representa aproximadamente 69% da população humana. O relatório também destaca os motivos de utilização da internet pela população, podendo-se destacar a busca de informações, contato com familiares e amigos e o conhecimento de notícias como os maiores destes motivos.

Durante a década de 1960, o governo dos Estados Unidos fundou e manteve a *Advanced Research Projects Agency Network* (ARPANET). Esta rede de computadores implementava o método de comunicação chamado de comutação de pacotes e existia com objetivo acadêmicos e científicos. Em torno da década dos anos de 1980, a ARPANET conectava centenas de universidades, instituições de pesquisa e empresas. Neste período outras redes surgiram, como a USENET e a X.25, (CONNOLLY; HOAR, 2015).

Com o propósito de promover a criação de outras redes e unificar a comunicação entre as mesmas, protocolos de comunicação foram desenhados e definidos. Em 1983 a ARPANET adotou oficialmente o protocolo *Transfer Control Protocol/Internetwork Protocol* (TCP/IP), marcando o início da adesão deste protocolo por todo o mundo e o surgimento da Internet (CONNOLLY; HOAR, 2015). Os conceitos e elementos que servem de base para o ambiente da Internet que temos contato atualmente foram apresentados no inicio da década de 1990 com a invenção da *World Wide Web* (WWW), também denominada *Web*, por Tim Berners-Lee. Estes elementos, que ainda são fundamentais, foram:

- O *Uniform Resource Locator* (URL), responsável por identificar um recurso único na

WWW.

- O protocolo *Hypertext Transfer Protocol* (HTTP), com o objetivo de descrever como as requisições e as respostas operam.
- A linguagem de marcação *HyperText Markup Language* (HTML), para publicação de documentos.
- Um software para responder as requisições HTTP, posteriormente denominado servidor *web*.
- Um navegador web responsável por realizar as requisições HTTP a partir de *Uniform Resource Locator* e mostrar o conteúdo do HTML recebido (CONNOLLY; HOAR, 2015).

Ao longo dos anos diversos outras ferramentas, *softwares* e convenções foram apresentadas que moldaram a *web*. Entre estas ferramentas estão o primeiro navegador gráfico, o Mosaic, disponibilizado em 1993 por estudantes da Universidade de Illinois, e outro navegador popular da década de 1990, o Netscape, desenvolvido posteriormente por parte destes estudantes (FELKE-MORRIS, 2019). Também é necessário citar a organização *World Wide Web Consortium* (W3C), fundada em 1994 (CONNOLLY; HOAR, 2015), que mantém um papel importante referente a padronização de tecnologias utilizadas na *Web* (FELKE-MORRIS, 2019).

3.1.2 Modelo cliente-servidor

O modelo cliente-servidor no âmbito computacional pode representar um processo no qual um agente, o cliente, efetua requisições á um outro agente denominado servidor, que por sua vez, cumpre a demanda destas requisições e transmite os resultados para o cliente através da rede (FELKE-MORRIS, 2019). É possível identificar este modelo arquitetura típica da *web*.

No contexto da comunicação que ocorre na *web*, o *software* Cliente, normalmente, é um navegador *web*, que possui a responsabilidade de requisitar um recurso específico utilizando uma URL, aguardar a resposta e exibi-lá da maneira adequada para o usuário. As máquinas hospedando o software cliente podem ser, *laptops* ou *smartphones*, por exemplo (CONNOLLY; HOAR, 2015).

Por outro lado o Servidor pode ser entendido como um repositório central do modelo, podendo hospedar aplicações *web* e armazenar dados, por exemplo. Sua característica principal é aguardar o recebimento de requisições, realizar o processamento necessário e responder com algum recurso, como um arquivo HTML. Existem diversos tipos de servidores e entre eles estão os servidores *web*, de aplicação, de banco de dados e de *e-mail* (CONNOLLY; HOAR, 2015).

A partir deste modelo é possível diferenciar as ferramentas, conceitos, necessidades e *frameworks* no desenvolvimento de aplicações *web* em duas áreas principais: o *Back-End* e o *Front-End*.

A área do *Front-End* está associada ao Cliente e seus artefatos normalmente executam em um navegador *web*. A escolha da ferramenta ligada ao *Front-End* que será utilizada na implementação do *software* proposto neste trabalho acontecerá baseando-se nos resultados obtidos por Martinotto (2021). Em contrapartida, o *Back-End* relaciona-se, tipicamente, com os elementos do Servidor. Como citado anteriormente, o *framework* Quarkus será apresentado na Seção 3.2, para que seja possível verificar a viabilidade de seu emprego como ferramenta centralizadora do *Back-End* do projeto.

3.2 QUARKUS

O Quarkus é um *framework* para o desenvolvimento de aplicações da linguagem de programação Java, que permite a criação de *softwares* modernos prontos para o ambiente da computação nativa da nuvem (QUARKUS, 2022).

A fácil integração com o Kubernetes, sistema de orquestração de contêineres, e com a plataforma Docker caracterizam-se como alguns de seus principais benefícios. Além disso é possível citar a baixa utilização de memória, a rápida inicialização e uma grande produtividade dos desenvolvedores (BUENO; PORTER, 2020).

O *framework* é um projeto de código aberto e licenciado sob a Licença Apache 2.0 (QUARKUS, 2022). A empresa americana Red Hat conduz o seu desenvolvimento e o apresentou ao público pela primeira vez no ano de 2019 (GONCALVES, 2020). O código-fonte da ferramenta é hospedado na plataforma GitHub e o lançamento da primeira versão ocorreu no dia 20 de março de 2019 nomeada 0.12.0.

Através de seu site oficial, os responsáveis pelo Quarkus destacam o objetivo de construir o futuro de aplicações Java e apresentam uma comunidade aberta a discussões e ideias. Os mesmos ainda reforçam o status absoluto de código aberto que a ferramenta possui (QUARKUS, 2022).

3.2.1 Surgimento

A Red Hat adquiriu a empresa JBoss no ano de de 2006, até então responsável pelo desenvolvimento do servidor de aplicação a *JavaBeans Open Source Software* (JBOSS), que passaria ser conhecido como JBOSS EAP, um sistema de tempo de execução que suporta todas as especificações da plataforma *Java Enterprise Edition* (Java EE). Ao longo dos anos a empresa continuou contribuindo com ferramentas para a comunidade de servidores de aplicação (GONCALVES, 2020).

Segundo Goncalves (2020), a mudança de foco para o ambiente da computação da nuvem passou a fazer sentido para a empresa, devido ao fato de que a mesma já comercializava o seu sistema operacional Linux corporativo, nomeado de *Red Hat Enterprise Linux*, e de já contribuir com o Java HotSpot, máquina virtual da linguagem Java focado em melhorias de

desempenho. Além disso, a empresa ainda possuía sua plataforma para implantação e gerenciamento de *softwares* baseados em contêiner, o *OpenShift Container Platform*.

A partir disso, a Red Hat precisava de uma alternativa aos servidores de aplicação, deficientes em alguns fatores importantes para o ambiente da nuvem, como inicialização tardia e a alocação de uma quantidade grande de memória. Neste contexto, ocorreu o surgimento do Quarkus (GONCALVES, 2020).

3.2.2 Funcionalidades

O Quarkus possui diversas características e apresenta uma vasta gama de funcionalidades, desde focar no prazer do usuário do *framework* durante o ciclo de desenvolvimento até o fato de disponibilizar extensões em bibliotecas conhecidas do mundo Java.

Uma funcionalidade, presente no ciclo de desenvolvimento ao utilizar o este *framework*, que deve ser citada é a possibilidade de compilar o *software* e disponibilizá-lo em um arquivo executável nativo e específico para um sistema operacional. Este recurso é possível devido ao suporte do *framework* ao GraalVM, uma distribuição JDK, Kit de Desenvolvimento Java, de alta performance que busca acelerar a execução de aplicações escritas em Java e outras linguagens compiladas para a máquina virtual Java como Kotlin e Groovy, além de suportar outras como JavaScript, Ruby e Python (GRAALVM, 2022).

O GraalVM utiliza a técnica de compilação *AOT* para gerar um arquivo binário nativo ao sistema operacional, que contém todo o conteúdo do programa e pode ser executado imediatamente (GONCALVES, 2020). Esta técnica de compilação melhora a performance do *software* drasticamente, diminuindo o tempo de inicialização significativamente e diminui a utilização de memória. O desempenho do Java neste tipo de compilação pode ser comparada a de linguagens como C++ (KOLEOSO, 2020).

O código compilado para uma imagem nativa utilizando o *Ahead-Of-Time* (AOT) busca manter exatamente apenas o necessário para executá-lo. Devido a este fato, o processo de compilação e *build*, ou seja, a construção do artefato final, ocorre em um período superior se comparado ao tempo em que um desenvolvedor Java está familiarizado normalmente (KOLEOSO, 2020). Levando isto em consideração, o ideal é utilizar a compilação para uma imagem nativa somente quando for necessário disponibilizar a aplicação para um ambiente de produção.

O Quarkus também disponibiliza o empacotamento da aplicação no chamado modo JVM, com a geração de arquivos no formato *Java ARchive* (JAR) tradicional, que posteriormente devem ser executados em um Ambiente de Tempo de Execução Java (JRE). Neste modo, é possível configurar o *framework* para empacotar a aplicação em um arquivo JAR, um arquivo Fast-JAR ou ainda um Uber-JAR (KOLEOSO, 2020)

A alegria do desenvolvedor, também denominada como *developer joy*, é apresentada como uma prioridade para o Quarkus (CLINGAN; FINNIGAN, 2022). Além disso, os responsáveis

pela manutenção do *framework* enfatizam que a utilização do mesmo, acima da alta produtividade, deve ser agradável (QUARKUS, 2022). Com o intuito de concretizar estas afirmações, a ferramenta disponibiliza funcionalidades como o *Live Coding* e a Configuração Unificada.

O recurso de *Live Coding*, presente somente quando o Quarkus está inicializado no modo específico para desenvolvimento, detecta mudanças em arquivos de código-fonte, arquivos de configuração e recursos estáticos, recompila e disponibiliza as modificações quase que instantaneamente, sem a necessidade de reiniciar a aplicação (CLINGAN; FINNIGAN, 2022). Esta funcionalidade economiza tempo e possibilita uma dinâmica mais ágil durante o ciclo de desenvolvimento.

Outra funcionalidade que o Quarkus implementa e proporciona com o objetivo de facilitar e simplificar o processo de desenvolvimento é a configuração unificada. O Quarkus suporta diversas ferramentas, bibliotecas e conceitos de diferentes ecossistemas como Java EE, Eclipse Vert.x e Spring, que separadamente necessitam de configurações especificas, geralmente em arquivos próprios. No entanto, o *framework* unifica todas estas configurações em apenas em arquivo de propriedades denominado "application.properties"(CLINGAN; FINNIGAN, 2022). Ainda a ferramenta garante que todas opções configuráveis apresentam um valor padrão funcional, para que seja possível iniciar o processo com o mínimo de parametrização (QUARKUS, 2022).

Os testes automatizados também são suportados pelo Quarkus. Segundo Goncalves (2020), o framework permite que o processo de testes da aplicação seja rápido e fácil, citando o fato de que os testes possuem a capacidade de inciarem sua execução rapidamente, logo após a inicialização da aplicação. O Quarkus possui integração com o JUnit 5, framework de código aberto conhecido no ecossistema Java que proporciona a escrita e execução de testes (GONCALVES, 2020). Além disso, o Quarkus disponibiliza a sua própria extensão denominada quarkus-junit5. que apresenta todas as funcionalidade existentes do JUnit 5 e oferece funcionalidades de testes exclusivas (KOLEOSO, 2020).

O suporte a diversos outras ferramentas conhecidas do mundo do *Back-End* também destaca-se como uma importante característica do Quarkus. É possível usufruir das especificações da plataforma Java EE, do Eclipse MicroProfile e do Spring Framework. O *framework* ainda implementa e disponibiliza robustas extensões a bibliotecas conhecidas, como uma extensão do Hibernate ORM, denominada Panache (KOLEOSO, 2020).

O Quarkus também qualifica-se como um *framework* amigável e ideal para aplicações voltadas para ambientes *serverless*, assim como para ambientes conteinerizados. O modelo de implantação *serverless*, oferecido por provedores de nuvem, como a Amazon Web Services (AWS), remove a necessidade de se preocupar com um servidor de aplicação, somente se fazendo necessário a entrega de um pacote de implantação. Alguns destes provedores cobram pela quantidade de tempo que a aplicação roda. Devido a isto, é de extrema necessidade que o programa inicialize no menor tempo possível. O suporte a compilação para imagens nativas proporciona a inicialização da aplicação em baixíssimo período de tempo. Esta característica

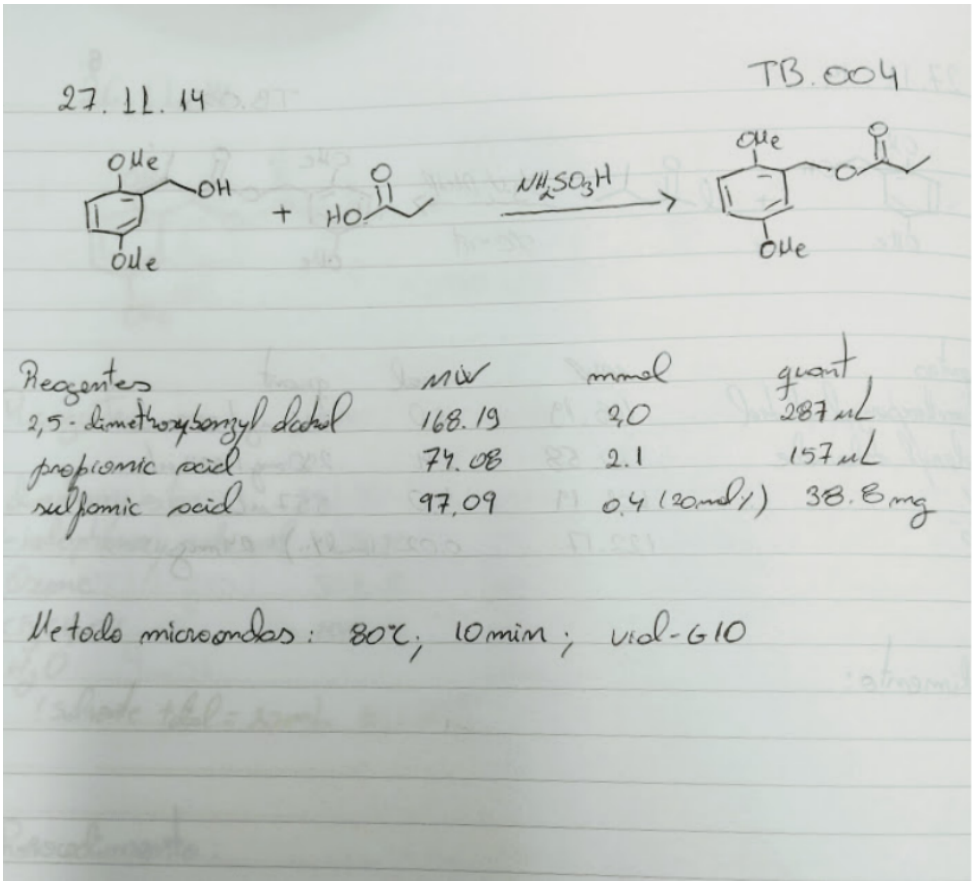
também define aplicações construídas com o Quarkus ideais também para ambientes em contêineres, já que a escalabilidade rápida da aplicação é desejada (KOLEOSO, 2020).

4 PROJETO

Como citado nos capítulos anteriores, a proposta de solução deste trabalho é o desenvolvimento de um *software* que auxilie os pesquisadores do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos da UCS na gravação e acesso as informações de cada experimento executado.

Atualmente, os pesquisadores utilizam cadernos físicos para anotar os dados destes experimentos. Dentre estes dados, estão a data de execução, uma numeração, a representação gráfica da reação química, a listagem dos reagentes e informações relacionados ao método utilizado. A Figura 2 apresenta um registro manual da execução de um experimento no laboratório.

Figura 2 – Anotações de um experimento



Fonte: (ROSSATO; SILVA, 2019)

A aplicação desenvolvida tem como funcionalidade central a execução de um experimento através do ambiente digital, possibilitando a gravação, o acesso e a exportação dos dados de forma funcional e fácil. No entanto, também foram implementadas outras funcionalidades que permitem que o software se caracterize realmente como uma representação no mundo digital do laboratório, proporcionando uma visão geral e também detalhada de todos processos e

integrantes do laboratório.

Dentre estas outras funcionalidades estão o cadastro e a manutenção de usuários, de grupos de usuários, de compostos químicos e de projetos, por exemplo. Além disso, é importante destacar que o acesso a plataforma somente é possível através de um endereço de *e-mail* e senha previamente cadastrados por outro usuário, o qual deve pertencer a um grupo que possua a permissão correta. Esta funcionalidade promove um aspecto de segurança importante para a aplicação e seus usuários.

O *framework* Quarkus foi utilizado para o desenvolvimento da parte do *software* associada ao *back-end*, responsável principalmente pelas regras de negócio e pela comunicação com um servidor de banco de dados. A escolha desta tecnologia ocorre devido as características e vantagens expostas no Seção 3.2, assim como relaciona-se com a familiaridade do autor do trabalho com a linguagem de programação da ferramenta, o Java. Também ocorreu a implementação de uma *Application Programming Interface* (API) para possibilitar o acesso dos dados por parte do *front-end*.

Como ferramenta para construção da aplicação associada ao *front-end*, utilizou-se o *framework* React, devido ao fato de que os resultados obtidos pela pesquisa de Martinotto (2021), demonstram a sua superioridade, tanto na avaliação geral quanto na avaliação em todos critérios, em relação ao VueJS e ao Angular.

Nas próximas seções as etapas do processo da metodologia ICONIX serão realizadas e apresentadas. Inicialmente, na Seção 4.1, ocorrerá o levantamento de requisitos, o que delimitará as necessidades que o software deve possuir. Posteriormente, na Seção 4.2 e na Seção 4.3, serão apresentados, respectivamente, o modelo de domínio e o diagrama de classes, produzidos, na metodologia ICONIX, durante o fluxo de trabalho denominado estático. A partir da Seção 4.5, os artefatos gerados durante a etapa do fluxo dinâmico serão exibidos. Estes artefatos são o diagrama de casos de uso e os diagramas de robustez. Além disso, nestas seções, cada caso de uso será detalhado com o design da tela implementada e uma tabela explicativa.

4.1 REQUISITOS

Os requisitos funcionais definem os diferentes comportamentos que o sistema deve apresentar, além de descreverem o que deve ser implementado para que os usuários realizem suas ações. Por outro lado os requisitos não-funcionais podem referir-se a características, propriedades e restrições do sistema relacionados a sua disponibilidade, usabilidade, performance e segurança, por exemplo (WIEGERS; BEATTY, 2013).

Os requisitos funcionais do sistema foram divididos em 5 tabelas, com cada uma referenciando os requisitos de um certo domínio. O modelo de domínios está mapeado na Seção 4.2.

A Figura 3 apresenta os requisitos funcionais relacionados ao domínio Usuário.

Figura 3 – Requisitos Funcionais - Usuário

Usuário	
Código	**Descrição**
RF-01	O sistema deve possibilitar o login de usuários.
RF-02	O sistema deve possibilitar o cadastro de usuários.
RF-03	O sistema deve possibilitar a edição e visualização de propriedades do próprio usuário.
RF-04	O sistema deve possibilitar a edição e visualização de propriedades de um usuário.
RF-05	O sistema deve possibilitar a listagem de usuários.
RF-06	O sistema deve possibilitar a realização de um filtro dos usuários.
RF-07	O sistema deve possibilitar a visualização do perfil de um usuário.

Fonte: O Autor (2022).

A Figura 4 apresenta os requisitos funcionais relacionados ao domínio Grupo.

Figura 4 – Requisitos Funcionais - Grupo

Grupo	
Código	**Descrição**
RF-08	O sistema deve possibilitar a criação, edição e visualização de um grupo.
RF-09	O sistema deve possibilitar a edição e visualização das permissões de um grupo.
RF-10	O sistema deve possibilitar a edição e visualização dos usuário de um grupo.
RF-11	O sistema deve possibilitar a listagem de grupos.
RF-12	O sistema deve possibilitar a realização de um filtro dos grupos.

Fonte: O Autor (2022).

A Figura 5 apresenta os requisitos funcionais relacionados ao domínio Projeto.

Figura 5 – Requisitos Funcionais - Projeto

Projeto	
Código	**Descrição**
RF-13	O sistema deve possibilitar a criação, edição e visualização de um projeto.
RF-14	O sistema deve possibilitar a edição e visualização dos usuários de um projeto.
RF-15	O sistema deve possibilitar a listagem de projetos.
RF-16	O sistema deve possibilitar a realização de um filtro dos projetos.

Fonte: O Autor (2022).

A Figura 6 apresenta os requisitos funcionais relacionados ao domínio Composto.

Figura 6 – Requisitos Funcionais - Composto

Composto	
Código	**Descrição**
RF-17	O sistema deve possibilitar a criação, edição e visualização de um composto.
RF-18	O sistema deve possibilitar a listagem de compostos.
RF-19	O sistema deve possibilitar a realização de um filtro de compostos.

Fonte: O Autor (2022).

A Figura 7 apresenta os requisitos funcionais relacionados ao domínio Experimento.

Figura 7 – Requisitos Funcionais - Experimento

Experimento	
Código	**Descrição**
RF-20	O sistema deve possibilitar a criação, edição e visualização de um experimento.
RF-21	O sistema deve possibilitar a criação, edição e visualização de anexos de um experimento.
RF-22	O sistema deve possibilitar a criação, edição e visualização de comentários de um experimento.
RF-23	O sistema deve possibilitar a criação, edição e visualização de linhas de experimentos.
RF-24	O sistema deve possibilitar a listagem dos experimentos.
RF-25	O sistema deve possibilitar a realização de um filtro dos experimentos.
RF-26	O sistema deve possibilitar a execução de um experimento.
RF-27	O sistema deve possibilitar que um experimento seja exportado.

Fonte: O Autor (2022).

A Figura 8 apresenta os requisitos não-funcionais do sistema.

Figura 8 – Requisitos Não-Funcionais

Código	Descrição
RNF-01	O dispositivo de acesso deve possuir conexão com a Internet.
RNF-02	O sistema não deve permitir o acesso a funções em que o usuário não possua permissão.

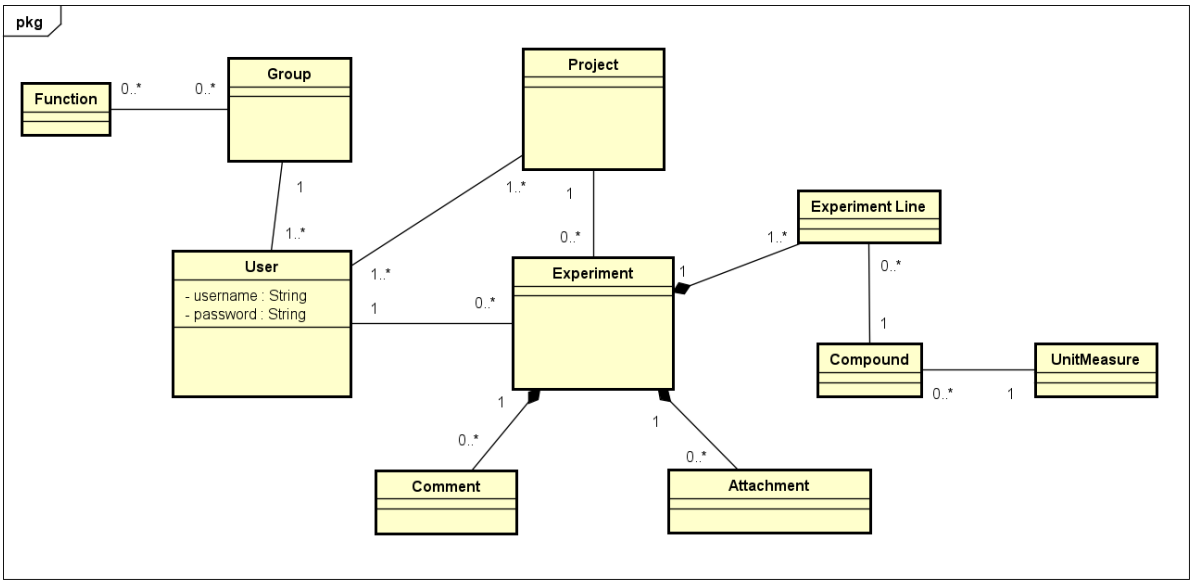
Fonte: O Autor (2022).

4.2 MODELO DE DOMÍNIO

O modelo de domínio é o primeiro artefato gerado no fluxo estático do ICONIX. Tem o objetivo de definir o escopo geral do projeto, apresentado os principais termos a serem utilizados no *software*. O seu refinamento dará origem ao diagrama de classes (ROSENBERG; STEPHENS, 2007).

A Figura 9 apresenta o diagrama de domínio do projeto.

Figura 9 – Modelo de Domínio



Fonte: O Autor (2022).

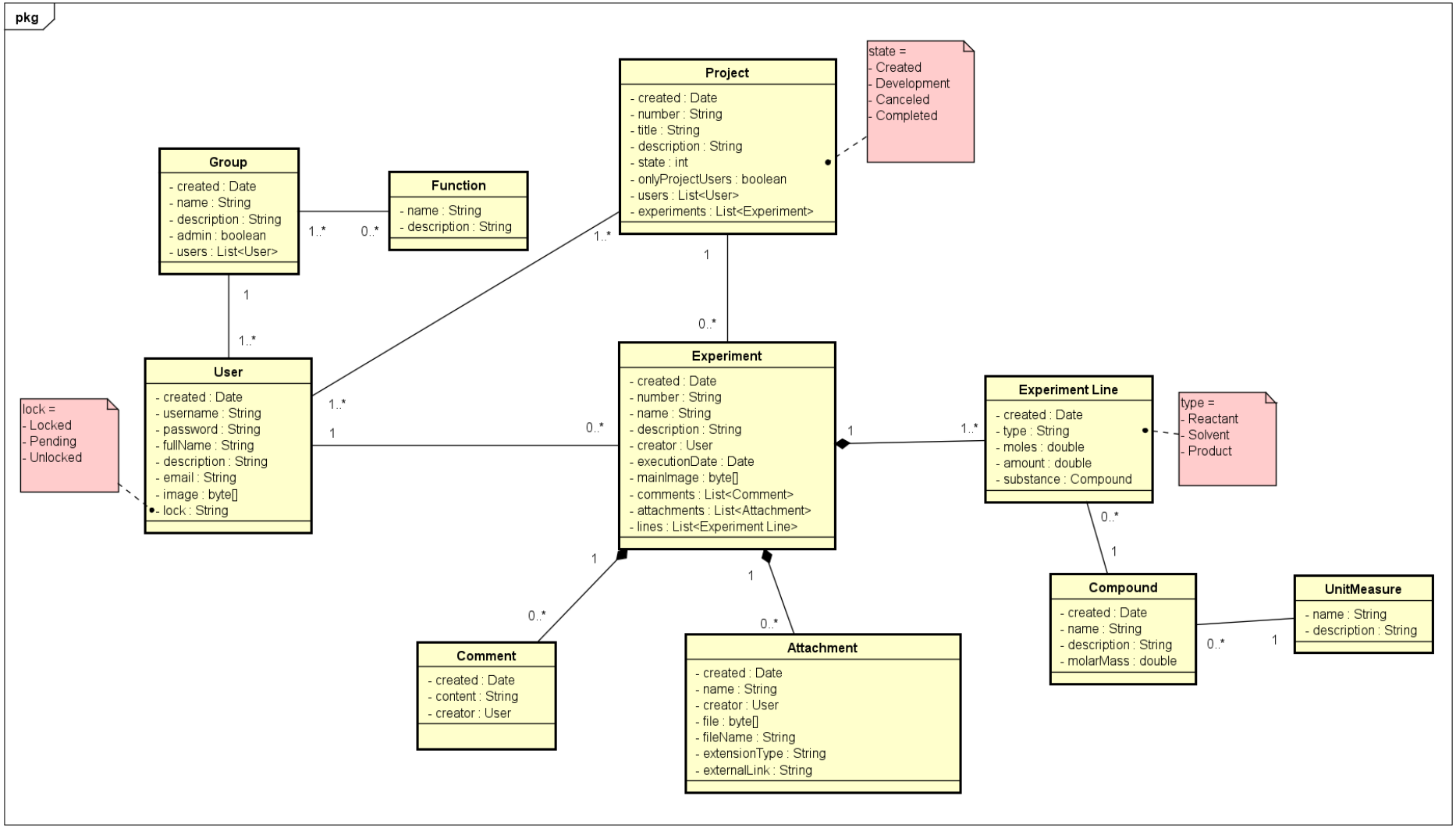
4.3 DIAGRAMA DE CLASSES

O diagrama de classes tem como propósito definir as classes presentes no sistema, além de mapear os relacionamentos entre as mesmas. Uma classe pode representar o modelo para um item do mundo-real que será incorporado na estrutura do sistema (WIEGERS; BEATTY, 2013).

No processo ICONIX o diagrama de classe faz parte do fluxo estático e é construído a partir do refinamento do modelo de domínio. Estes diagramas são alterados e revisados durante as diferentes etapas do processo (ROSENBERG; STEPHENS, 2007).

A Figura 10 apresenta o diagrama de classes do sistema, detalhado a partir do modelo de domínio definido na Seção 4.2.

Figura 10 – Diagrama de Classes



Fonte: O Autor (2022).

A Figura 11 apresenta uma tabela com as classes mapeadas, assim como uma descrição relacionada a cada uma delas.

Figura 11 – Classes e suas descrições

Classe	Descrição
User	Representa um usuário cadastrado no sistema. Deve estar relacionado a um Group.
Group	Representa um agrupamento de usuários que possuem as mesmas permissões. Pode estar relacionado a uma lista de Function. Exemplo : grupo denominado Professores e outro Pesquisadores.
Function	Permissão do sistema.
Project	Representa um projeto de pesquisa ou estudo de um ou mais usuários. Pode estar relacionado a uma lista de Experiment e uma lista de User.
Experiment	Representa um experimento, neste caso uma reação química. Pode estar relacionado a uma lista de Comment e Attachment. Deve estar relacionado a um User.
Comment	Representa um comentário de um usuário em referência a um Experiment. Deve estar relacionado a um User.
Attachment	Representa um anexo ou link externo. Deve estar relacionado a um Experiment.
ExperimentLine	Representa um registro que pode ser do tipo Reagente, Solvente ou Produto. Deve estar relacionado a um Compound.
Compound	Representa um composto químico. Deve estar relacionado a um UnitMeasure.
UnitMeasure	Representa uma unidade de medida.

Fonte: O Autor (2022).

4.4 DIAGRAMA DE CASOS DE USO

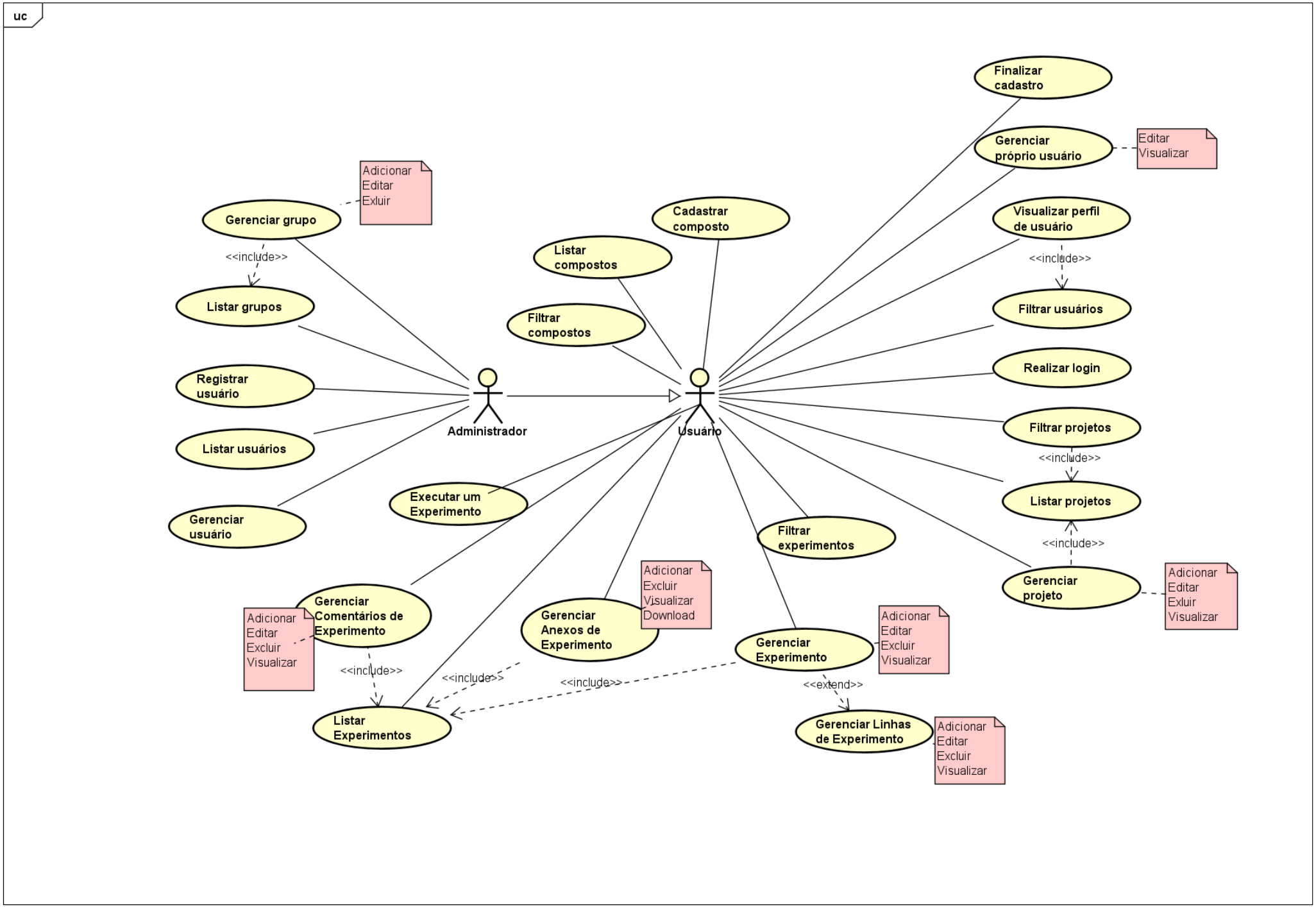
Os casos de uso definem as ações que serão realizadas entre um ator e o sistema. Os atores são agentes externos, como pessoas ou outros sistemas, que interagem com o sistema executando um caso de uso (WIEGERS; BEATTY, 2013).

Para este projeto, foram mapeados os atores Usuário e o Administrador. O Usuário representa qualquer usuário cadastrado e o Administrador é um usuário que pertence a um grupo de administradores. Este grupo, por sua vez, é caracterizado pelo fato de não possuir restrições em relação à permissões do sistema.

Os casos de uso podem ser agrupados em um diagrama de casos de uso. Este diagrama apresenta o título dos casos de uso mapeados, relacionado-os com um ou mais atores (ROSENBERG; STEPHENS, 2007).

A Figura 12 apresenta o diagrama de casos de uso do projeto.

Figura 12 – Diagrama de Casos de Uso



Fonte: O Autor (2022).

4.5 DETALHAMENTO DE CASOS DE USO

Nesta seção, os casos de usos apresentados na Figura 10 serão detalhados como previsto no processo ICONIX.

Os casos de uso foram separados em grupos levando em consideração com qual domínio os mesmos se relacionam. Deste modo, os detalhamentos também serão apresentados em diferente subseções.

4.5.1 DETALHAMENTO DE CASOS DE USO - USUÁRIO

Nesta seção, serão detalhados os caso de uso relacionados principalmente ao domínio User.

A Figura 13 lista os códigos e títulos destes casos de uso.

Figura 13 – Casos de Uso - Usuário

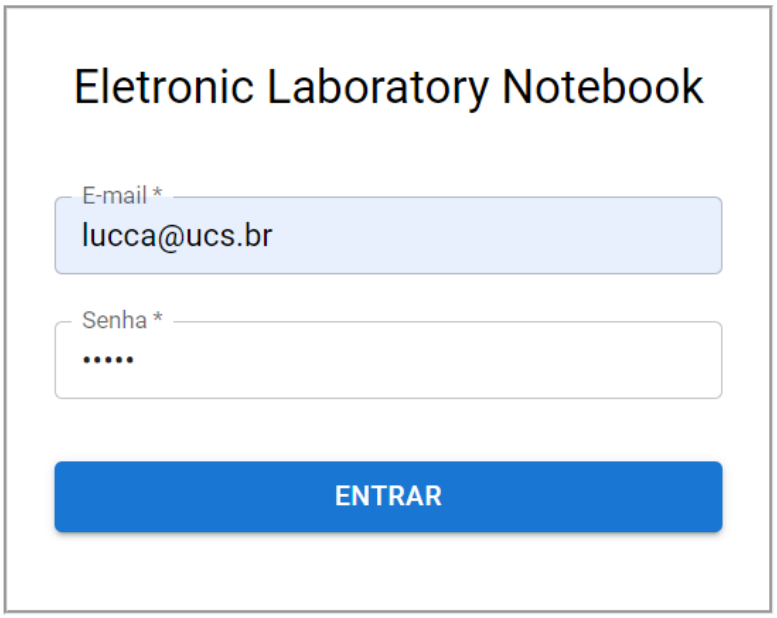
Código	Título
UC-01	Realizar Login
UC-02	Registrar Usuário
UC-03	Finalizar registro de Usuário
UC-04	Listar Usuários
UC-05	Filtrar Usuários
UC-06	Gerenciar Usuário
UC-07	Gerenciar o próprio Usuário
UC-08	Visualizar perfil de Usuário

Fonte: O Autor (2022).

4.5.1.1 Caso de Uso UC-01 Realizar *Login*

O caso de uso UC-01 Realizar *Login* atende ao requisito funcional RF-01. A Figura 14 apresenta a interface gráfica deste caso de uso.

Figura 14 – Interface Gráfica UC-01 Realizar *Login*



Fonte: O Autor (2022).

O Quadro 1 apresenta o detalhamento deste caso de uso.

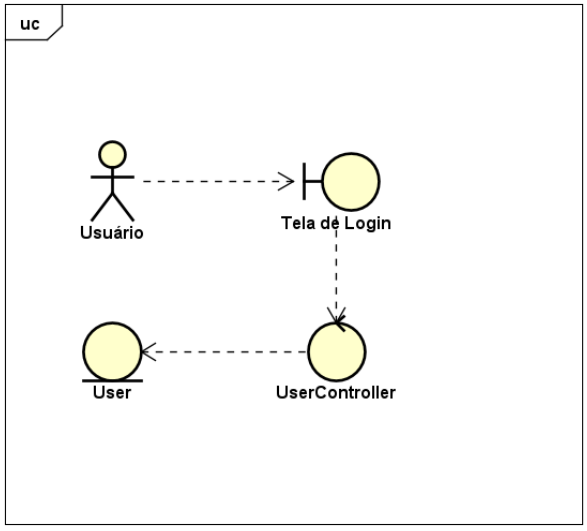
Quadro 1 – Detalhamento UC-01 Realizar *Login*

Código e Descrição	UC-01 Realizar *Login*
Ator	Usuário
Descrição	Permite que um usuário realize a autenticação no sistema.
RF Associados	RF-01
Pré-Condições	O usuário deve estar cadastrado no sistema e com bloqueio 'Unlocked'
Pós-Condições	Usuário autenticado e redirecionado para tela inicial.
Fluxo principal	1. Usuário acessa tela 'Login'. 2. Usuário informa e-mail e senha. 3. Usuário pressiona botão 'Entrar'. 4. Sistema valida dados. 5. Sistema autentica usuário. 6. Sistema redireciona usuário para tela inicial.
Fluxos alternativos	**Validação de Dados** 4.a) Dados inválidos. 1.Sistema informa que usuário não existe. 2.Sistema informa que senha está incorreta. 3.Sistema informa que usuário está bloqueado. 4.b) Dados obrigatórios. 1.Sistema informa que e-mail é obrigatório. 2.Sistema informa que senha é obrigatória. **Usuário com estado 'Pending'** 6. Sistema redireciona para o caso de uso 'Finalizar Registro'

Fonte: O Autor (2021).

A Figura 15 apresenta o diagrama de robustez deste caso de uso.

Figura 15 – Diagrama de Robustez UC-01 Realizar *Login*



Fonte: O Autor (2022).

4.5.1.2 Caso de Uso UC-02 Registrar Usuário

O caso de uso UC-02 Registrar Usuário atende ao requisito funcional RF-02. A Figura 16 apresenta a interface gráfica deste caso de uso.

Figura 16 – Recorte da Interface Gráfica UC-02 Registrar Usuário



Fonte: O Autor (2022).

O Quadro 2 apresenta o detalhamento deste caso de uso.

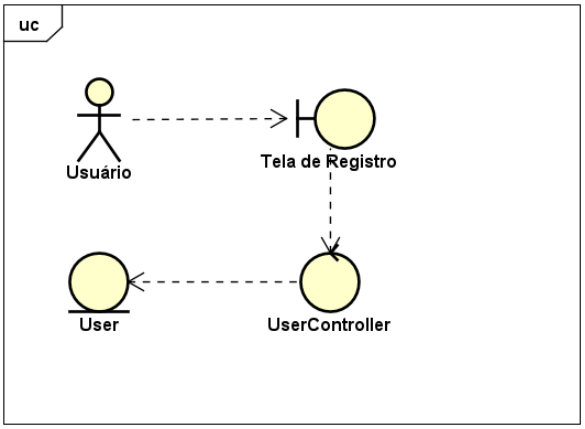
Quadro 2 – Detalhamento UC-02 Registrar Usuário

Código e Descrição	UC-02 Registrar Usuário
Ator	Usuário
Descrição	Permite que um usuário registre um novo usuário no sistema.
RF Associados	RF-02
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Usuários' e 'Registrar Usuário'.
Pós-Condições	Novo usuário cadastrado.
Fluxo principal	1. Usuário selecione a entrada 'Usuários' no menu lateral. 2. Sistema exibe a tela 'Usuários'. 3. Usuário pressiona botão 'Adicionar', representado pelo ícone de adição. 4. Sistema exibe a janela 'Registrar Usuário'. 5. Usuário informa e-mail e nome do grupo do novo usuário. 6. Usuário pressiona botão 'Registrar'. 7. Sistema valida dados. 8. Sistema cria registro de novo usuário com bloqueio 'Pending'. 9. Sistema envia e-mail com senha temporária para finalização do cadastro. 10. Sistema fecha a janela 'Registrar Usuário'.
Fluxos alternativos	**Cancelamento** 6. Usuário pressiona botão 'Cancelar'. 7. Sistema fecha janela 'Registrar Usuário'. **Validação de Dados** 7.a) Dados inválidos. 1.Sistema informa que já existe usuário com e-mail informado. 7.b) Dados obrigatórios. 1.Sistema informa que email é obrigatório. 2.Sistema informa que grupo é obrigatório. **Falha ao enviar e-mail** 9. Sistema informa que ocorreu uma falha ao enviar e-mail.

Fonte: O Autor (2021).

A Figura 17 apresenta o diagrama de robustez deste caso de uso.

Figura 17 – Diagrama de Robustez UC-02 Registrar Usuário



Fonte: O Autor (2022).

4.5.1.3 Caso de Uso UC-03 Finalizar registro de Usuário

O caso de uso UC-03 Finalizar registro de Usuário atende ao requisito funcional RF-02. A Figura 18 apresenta a interface gráfica deste caso de uso.

Figura 18 – Interface Gráfica UC-03 Finalizar registro de Usuário



Fonte: O Autor (2022).

O Quadro 3 apresenta o detalhamento deste caso de uso.

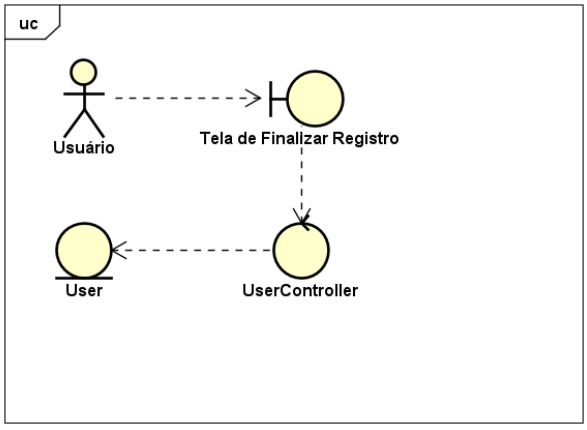
Quadro 3 – Detalhamento UC-03 Finalizar registro de Usuário

Código e Descrição	UC-03 Finalizar registro de Usuário
Ator	Usuário
Descrição	Permite que um usuário finalize o seu cadastro no sistema.
RF Associados	RF-02
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. Realização do caso de uso 'UC-02 Registrar Usuário' por outro usuário. 3. O usuário deve estar com estado 'Pending'.
Pós-Condições	1.Bloqueio do usuário alterado para 'Unlocked'. 2.Registro do usuário editado com dados fornecidos.
Fluxo principal	1. Usuário realiza caso de uso 'UC-01 Realizar Login'. 2. Sistema exibe a tela 'Finalização Cadastro - Usuário'. 3. Usuário informa 'Nome Completo', 'Username', 'Senha' e 'Descrição'. 3. Usuário pressiona botão 'Finalizar Cadastro'. 4. Sistema valida dados. 5. Sistema altera bloqueio do usuário para 'Unlocked' 6. Sistema altera registro do Usuário com dados informados. 7. Sistema redireciona usuário para tela 'Login'.
Fluxos alternativos	**Validação de Dados** 4.a) Dados inválidos. 1. Sistema informa que a senha deve possuir pelo menos um dígito e pelo menos um caractere especial. 4.b) Dados obrigatórios. 1. Sistema informa que o nome completo é obrigatório. 2. Sistema informa que o username é obrigatório. 3. Sistema informa que a senha é obrigatória.

Fonte: O Autor (2021).

A Figura 19 apresenta o diagrama de robustez deste caso de uso.

Figura 19 – Diagrama de Robustez UC-03 Finalizar registro de Usuário

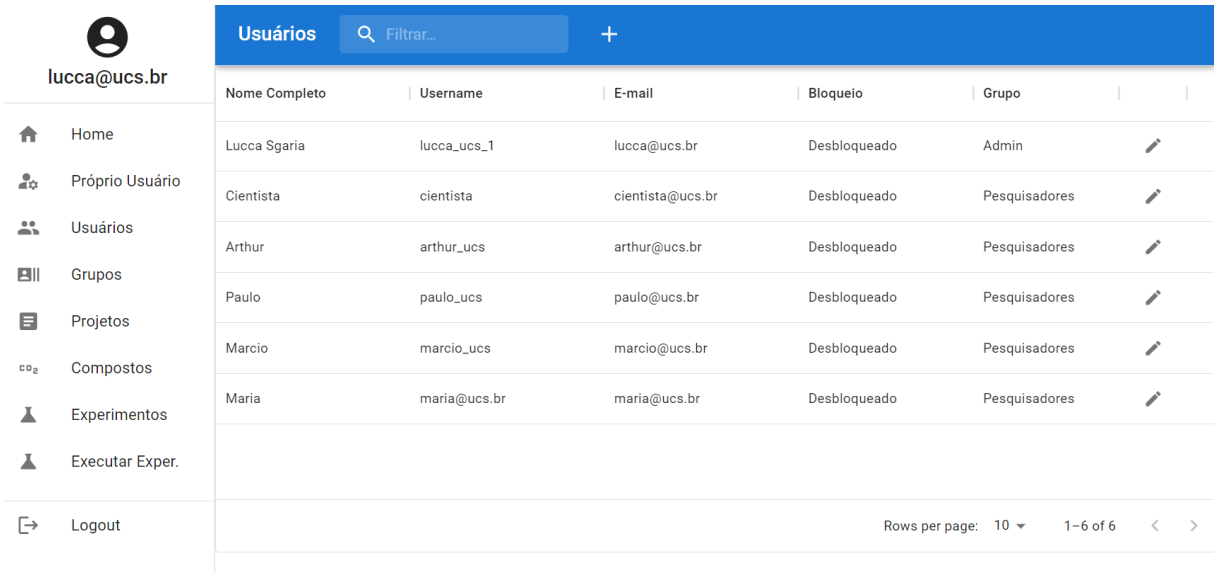


Fonte: O Autor (2022).

4.5.1.4 Caso de Uso UC-04 Listar Usuários

O caso de uso UC-04 Listar Usuários atende ao requisito funcional RF-05. A Figura 20 apresenta a interface gráfica deste caso de uso.

Figura 20 – Interface Gráfica UC-04 Listar Usuários



Fonte: O Autor (2022).

O Quadro 4 apresenta o detalhamento deste caso de uso.

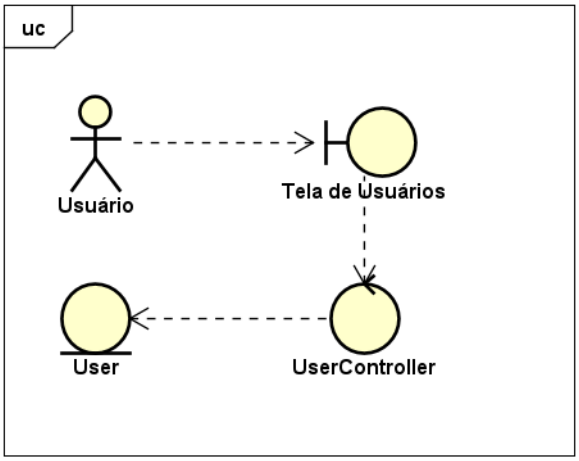
Quadro 4 – Detalhamento UC-04 Listar Usuários

Código e Descrição	UC-04 Listar Usuários
Ator	Usuário
Descrição	Permite que um usuário realize uma listagem dos usuários cadastrados no sistema
RF Associados	RF-05
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Usuários'.
Pós-Condições	Usuários do sistema listados.
Fluxo principal	1. Usuário selecione a entrada 'Usuários' no menu lateral. 2. Sistema exibe a tela 'Usuários'. 3. Sistema realiza busca dos usuários cadastros. 4. Sistema exibe tabela dos usuários cadastrados com as colunas 'Nome Completo', 'Username', E-mail', 'Bloqueio' e 'Grupo".
Fluxos alternativos	

Fonte: O Autor (2021).

A Figura 21 apresenta o diagrama de robustez deste caso de uso.

Figura 21 – Diagrama de Robustez UC-04 Listar Usuários

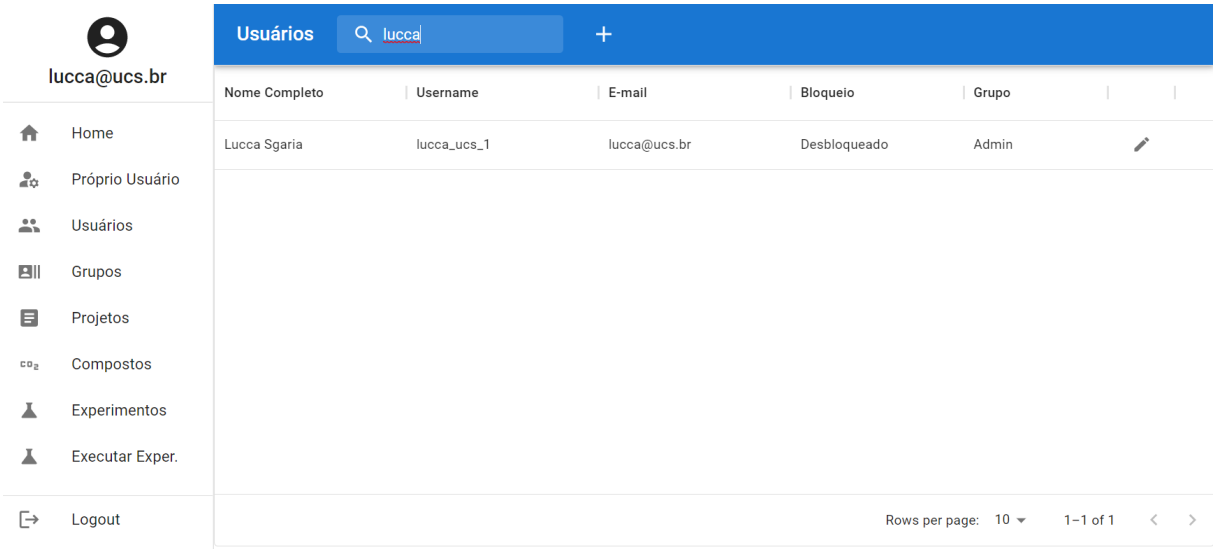


Fonte: O Autor (2022).

4.5.1.5 Caso de Uso UC-05 Filtrar Usuários

O caso de uso UC-05 Filtrar Usuários atende ao requisito funcional RF-06. A Figura 22 apresenta a interface gráfica deste caso de uso.

Figura 22 – Interface Gráfica UC-05 Filtrar Usuários



Fonte: O Autor (2022).

O Quadro 5 apresenta o detalhamento deste caso de uso.

Quadro 5 – Detalhamento UC-05 Filtrar Usuários

Código e Descrição	UC-05 Filtrar Usuários
Ator	Usuário
Descrição	Permite que um usuário realize uma filtragem dos usuários cadastrados no sistema
RF Associados	RF-06
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Usuários' e 'Filtrar Usuários'. 3. Realização do caso de uso 'UC-04 Listar Usuários'.
Pós-Condições	Usuários que correspondem ao filtro listados.
Fluxo principal	1. Usuário informa um filtro na caixa de pesquisa. 2. Usuário pressiona o Enter. 3. Sistema realiza busca dos usuários cadastros que possuam o 'Username', 'Nome Completo' ou 'E-mail' similar ao do filtro informado ou que participem de Grupo com 'Nome' similar ao do filtro. 4. Sistema exibe tabela dos usuários filtrados com as colunas 'Nome Completo', 'Username', 'E-mail', 'Bloqueio' e 'Grupo".
Fluxos alternativos	

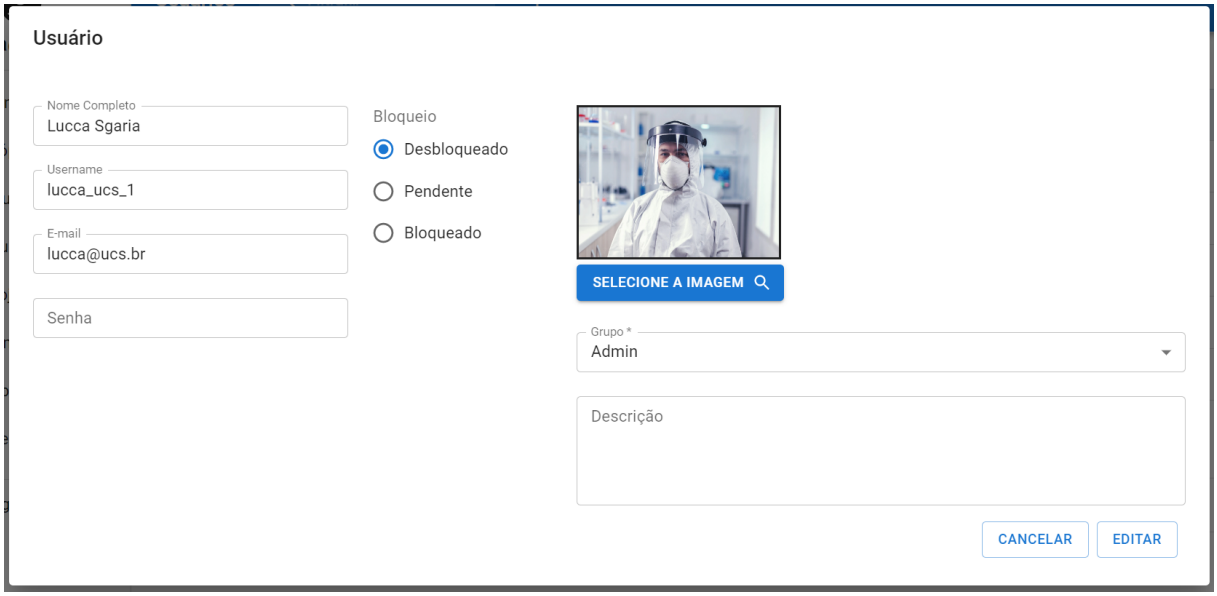
Fonte: O Autor (2021).

O caso de uso possui o mesmo diagrama de robustez apresentado na Figura 21.

4.5.1.6 Caso de Uso UC-06 Gerenciar Usuário

O caso de uso UC-06 Gerenciar Usuário atende ao requisito funcional RF-04. A Figura 23 apresenta a interface gráfica deste caso de uso.

Figura 23 – Recorte da Interface Gráfica UC-06 Gerenciar Usuário



Fonte: O Autor (2022).

O Quadro 6 apresenta o detalhamento deste caso de uso.

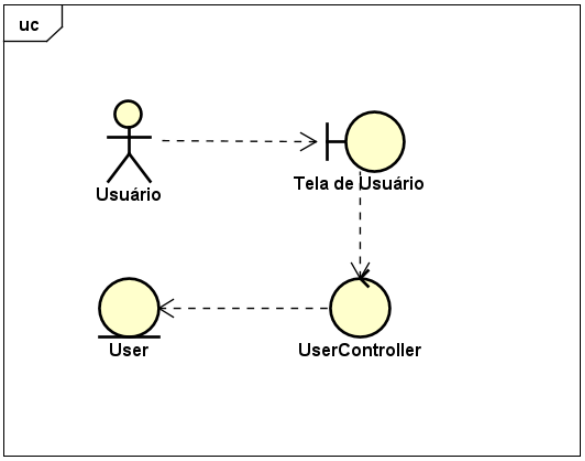
Quadro 6 – Detalhamento UC-06 Gerenciar Usuário

Código e Descrição	UC-06 Gerenciar Usuário
Ator	Usuário
Descrição	Permite que um usuário visualize e altere as propriedades de um usuário.
RF Associados	RF-04
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Usuários' e 'Gerenciar Usuário'. 3. Realização do caso de uso 'UC-04 Listar Usuários'
Pós-Condições	Propriedades de um usuário exibidas e/ou alteradas.
Fluxo principal	1. Usuário clica no ícone de edição do usuário que deseja visualizar e/ou alterar. 2. Sistema exibe janela 'Usuário' com as propriedades do usuário selecionado: 'Nome Completo', 'Username', 'E-mail', 'Senha', 'Bloqueio', 'Imagem', 'Grupo' e 'Descrição'. 3. Usuário altera propriedades que deseja. 4. Usuário pressiona botão 'Editar'. 5. Sistema valida dados. 6. Sistema altera registro do usuário com propriedades informadas. 7. Sistema fecha janela 'Usuário'.
Fluxos alternativos	**Apenas Visualização** 3. Usuário pressiona botão 'Cancelar'. 4. Sistema fecha janela 'Usuário'. **Validação de Dados** 5.a) Dados inválidos. 1.Sistema informa que o e-mail já está cadastrado. 5.b) Dados obrigatórios. 1.Sistema informa que o username é obrigatório. 2.Sistema informa que o nome completo é obrigatório. 3.Sistema informa que senha é obrigatória. 4.Sistema informa que e-mail é obrigatório.

Fonte: O Autor (2021).

A Figura 24 apresenta o diagrama de robustez deste caso de uso.

Figura 24 – Diagrama de Robustez UC-06 Gerenciar Usuário

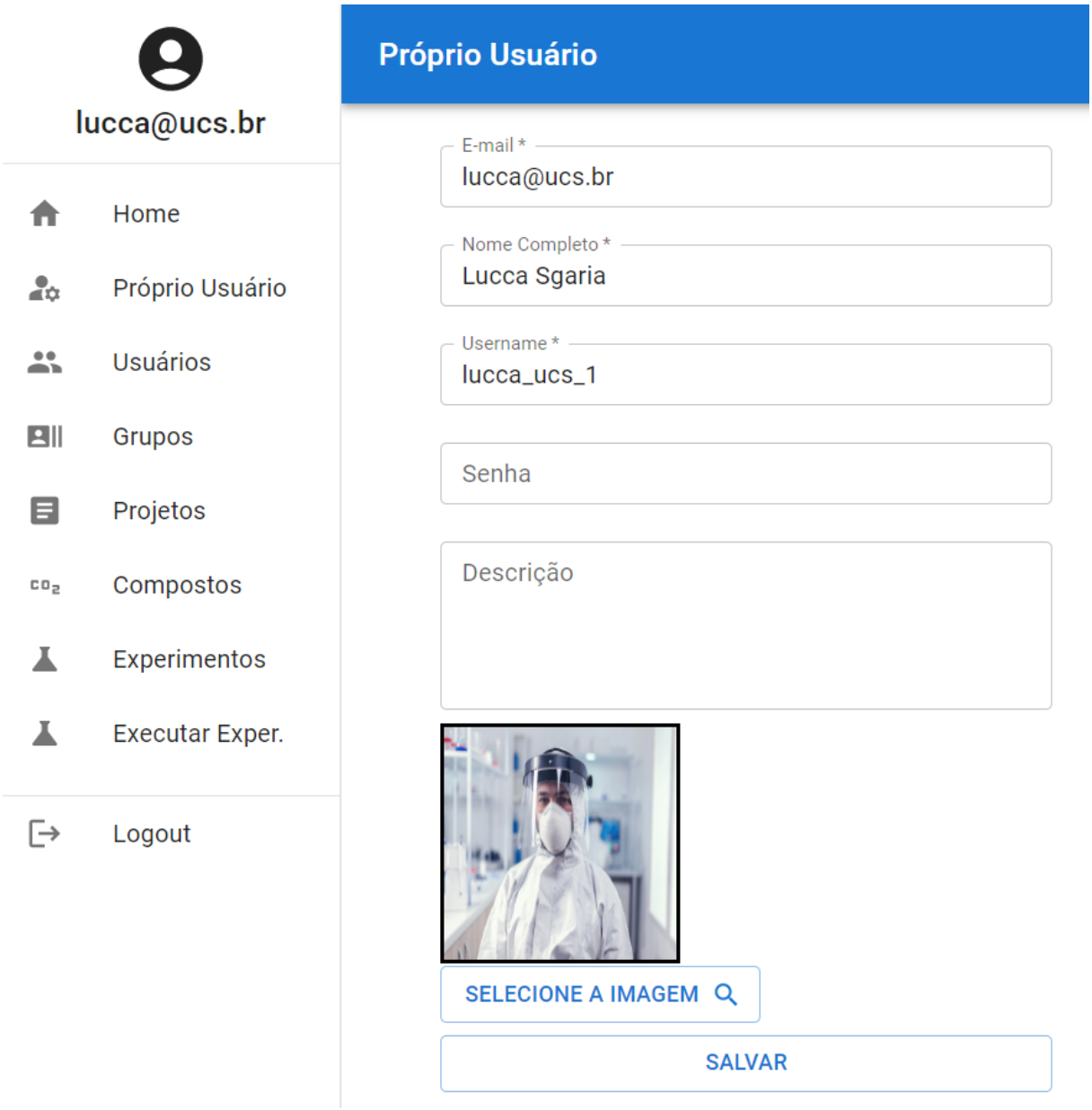


Fonte: O Autor (2022).

4.5.1.7 Caso de Uso UC-07 Gerenciar o próprio Usuário

O caso de uso UC-06 Gerenciar Usuário atende ao requisito funcional RF-03. A Figura 25 apresenta a interface gráfica deste caso de uso.

Figura 25 – Recorte da Interface Gráfica UC-07 Gerenciar o próprio Usuário



Fonte: O Autor (2022).

O Quadro 7 apresenta o detalhamento deste caso de uso.

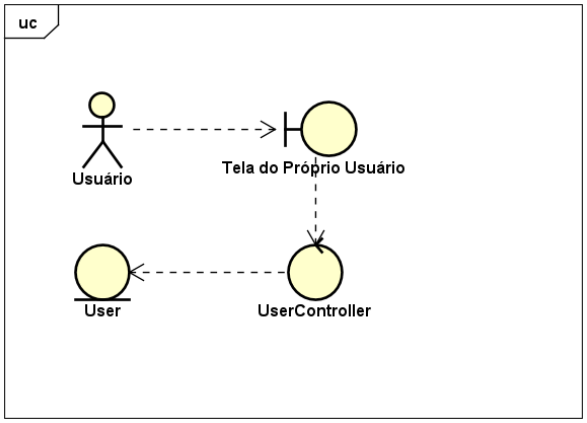
Quadro 7 – Detalhamento UC-07 Gerenciar o próprio Usuário

Código e Descrição	UC-07 Gerenciar o próprio Usuário
Ator	Usuário
Descrição	Permite que um usuário visualize e altere as propriedades de seu próprio registro.
RF Associados	RF-03
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Gerenciar Próprio Usuário'.
Pós-Condições	Propriedades do próprio usuário exibidas e/ou alteradas.
Fluxo principal	1. Usuário selecione a entrada 'Próprio Usuário' no menu lateral. 2. Sistema exibe a tela 'Próprio Usuário' com as propriedades do usuário autenticado : 'Nome Completo', 'Username', 'E-mail', 'Senha', 'Imagem' e 'Descrição'. 3. Usuário altera propriedades que deseja. 4. Usuário pressiona botão 'Salvar'. 5. Sistema valida dados. 6. Sistema altera registro do usuário autenticado com propriedades informadas.
Fluxos alternativos	**Validação de Dados** 5.a) Dados inválidos. 1. Sistema informa que o e-mail já está cadastrado. 2. Sistema informa que a senha deve possuir pelo menos um dígito e pelo menos um caractere especial. 5.b) Dados obrigatórios. 1.Sistema informa que o username é obrigatório. 2.Sistema informa que o nome completo é obrigatório. 3.Sistema informa que senha é obrigatória. 4.Sistema informa que e-mail é obrigatório.

Fonte: O Autor (2021).

A Figura 26 apresenta o diagrama de robustez deste caso de uso.

Figura 26 – Diagrama de Robustez UC-07 Gerenciar o próprio Usuário

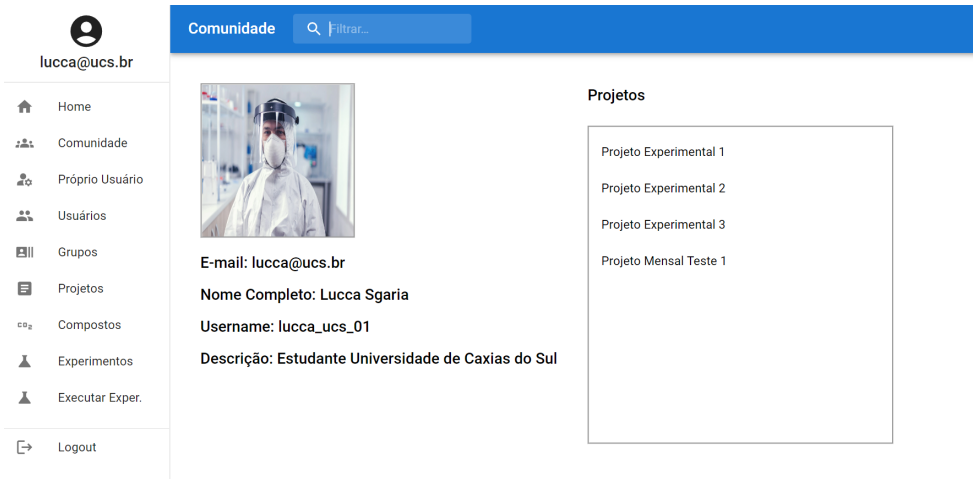


Fonte: O Autor (2022).

4.5.1.8 Caso de Uso UC-08 Visualizar perfil de Usuário

O caso de uso UC-08 Visualizar perfil de Usuário atende ao requisito funcional RF-07. A Figura 27 apresenta a interface gráfica deste caso de uso.

Figura 27 – Interface Gráfica UC-08 Visualizar perfil de Usuário



Fonte: O Autor (2022).

O Quadro 8 apresenta o detalhamento deste caso de uso.

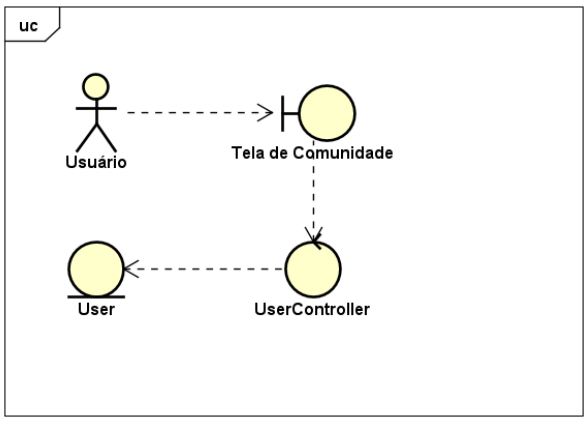
Quadro 8 – Detalhamento UC-08 Visualizar perfil de Usuário

Código e Descrição	UC-08 Visualizar perfil de Usuário
Ator	Usuário
Descrição	Permite que um usuário visualize o perfil de um usuário.
RF Associados	RF-07
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão ‘Comunidade’.
Pós-Condições	Perfil de usuário exibido.
Fluxo principal	1. Usuário seleciona a entrada ‘Comunidade’ no menu lateral. 2. Sistema exibe a tela ‘Comunidade’ com o perfil do usuário autenticado. O perfil possui as propriedades : ‘Nome Completo’, ‘Username’, ‘E-mail’, ’Imagem’ e ‘Descrição’, assim como uma listagem dos projetos deste usuário. 3. Usuário informa um filtro na caixa de pesquisa. 4. Usuário pressiona o Enter. 5. Sistema realiza busca dos usuários cadastros que possuam o username, nome completo ou e-mail igual ao do filtro informado ou que participem de grupo com nome igual ao do filtro . 6. Sistema exibe listagem dos nomes dos usuários filtrados. 7. Usuário seleciona o item da lista desejado. 8. Sistema exibe perfil do usuário selecionado. 9. Caso de Uso é encerrado.
Fluxos alternativos	**Repetição** 9. Usuário repete os passos a partir do item 3.

Fonte: O Autor (2021).

A Figura 28 apresenta o diagrama de robustez deste caso de uso.

Figura 28 – Diagrama de Robustez UC-08 Visualizar perfil de Usuário



Fonte: O Autor (2022).

4.5.2 DETALHAMENTO DE CASOS DE USO - GRUPO

Nesta seção, serão detalhados os caso de uso relacionados principalmente aos domínios Group e Function.

A Figura 29 lista os códigos e títulos destes casos de uso.

Figura 29 – Casos de Uso - Grupo

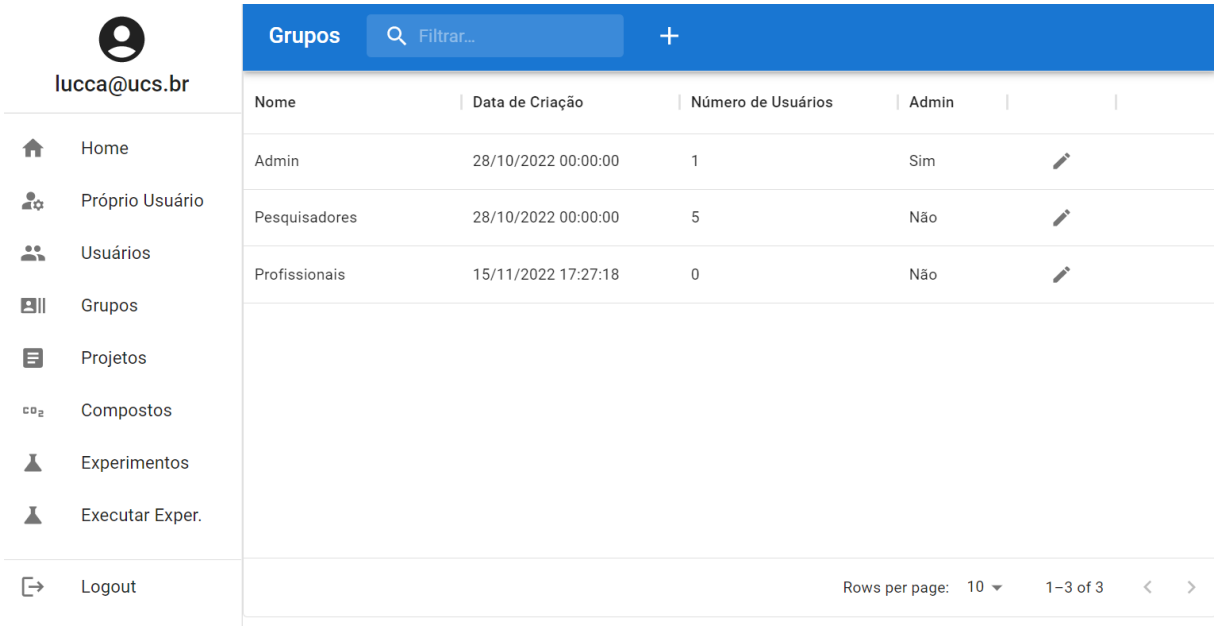
Código	Descrição
UC-09	Listar Grupos
UC-10	Filtrar Grupos
UC-11	Gerenciar Grupo

Fonte: O Autor (2022).

4.5.2.1 Caso de Uso UC-09 Listar Grupos

O caso de uso ’UC-09 Listar Grupos’ atende ao requisito funcional RF-11. A Figura 30 apresenta a interface gráfica deste caso de uso.

Figura 30 – Interface Gráfica UC-09 Listar Grupos



Fonte: O Autor (2022).

O Quadro 9 apresenta o detalhamento deste caso de uso.

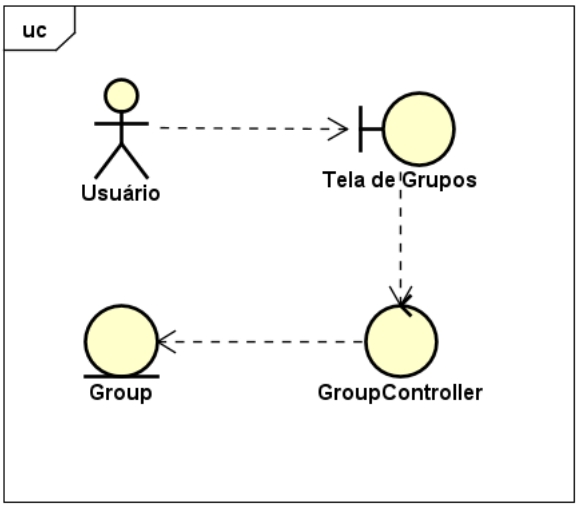
Quadro 9 – Detalhamento UC-09 Listar Grupos

Código e Descrição	UC-09 Listar Grupos
Ator	Usuário
Descrição	Permite que um usuário realize uma listagem dos grupos cadastrados no sistema
RF Associados	RF-11
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Grupos'.
Pós-Condições	Grupos do sistema listados.
Fluxo principal	1. Usuário selecione a entrada 'Grupos' no menu lateral. 2. Sistema exibe a tela 'Grupos'. 3. Sistema realiza busca dos grupos cadastros. 4. Sistema exibe tabela dos grupos cadastrados com as colunas 'Nome', 'Data de Criação', 'Número de Usuários' e 'Admin'.
Fluxos alternativos	

Fonte: O Autor (2022).

A Figura 31 apresenta o diagrama de robustez deste caso de uso.

Figura 31 – Diagrama de Robustez UC-09 Listar Grupos

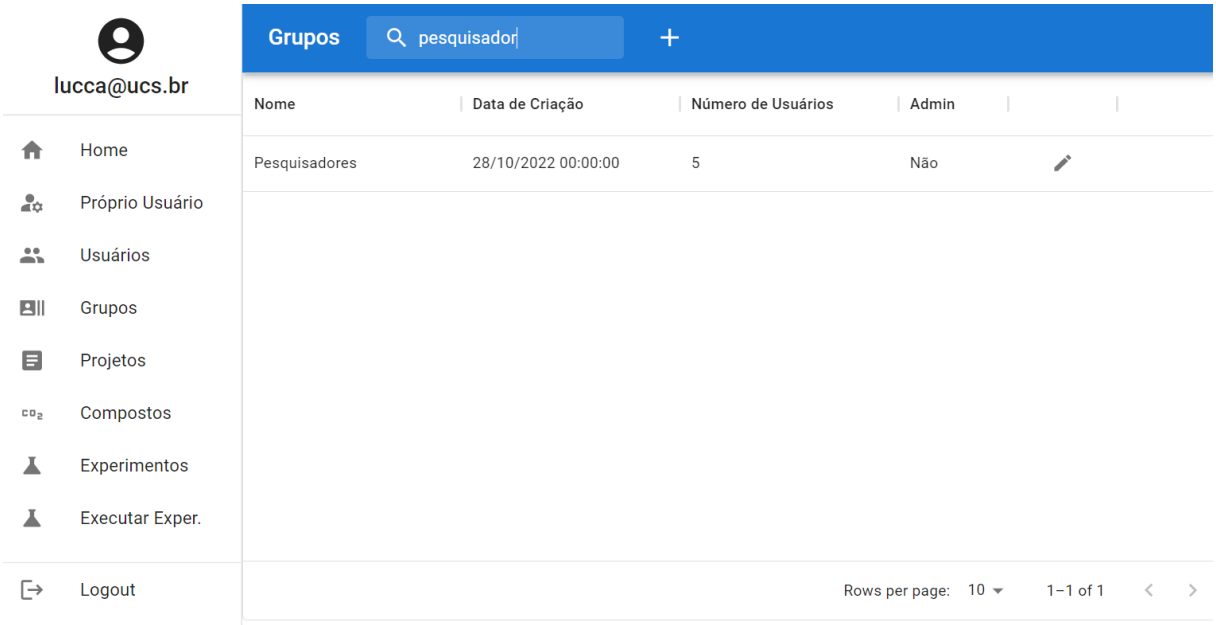


Fonte: O Autor (2022).

4.5.2.2 Caso de Uso UC-10 Filtrar Grupos

O caso de uso 'UC-10 Filtrar Grupos' atende ao requisito funcional RF-12. A Figura 32 apresenta a interface gráfica deste caso de uso.

Figura 32 – Interface Gráfica UC-10 Filtrar Grupos



Fonte: O Autor (2022).

O Quadro 10 apresenta o detalhamento deste caso de uso.

Quadro 10 – Detalhamento UC-10 Filtrar Grupos

Código e Descrição	UC-10 Filtrar Grupos
Ator	Usuário
Descrição	Permite que um usuário realize uma filtragem dos grupos cadastrados no sistema
RF Associados	RF-12
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Grupos' e 'Filtrar Grupos'. 3. Realização do caso de uso 'UC-10 Listar Grupos'.
Pós-Condições	Grupos que correspondem ao filtro listados.
Fluxo principal	1. Usuário informa um filtro na caixa de pesquisa. 2. Usuário pressiona o Enter. 3. Sistema realiza busca dos grupos cadastros que possuam o 'Nome' similar ao do filtro informado. 4. Sistema exibe tabela dos grupos filtrados com as colunas 'Nome', 'Data de Criação', 'Número de Usuários' e 'Admin'.
Fluxos alternativos	

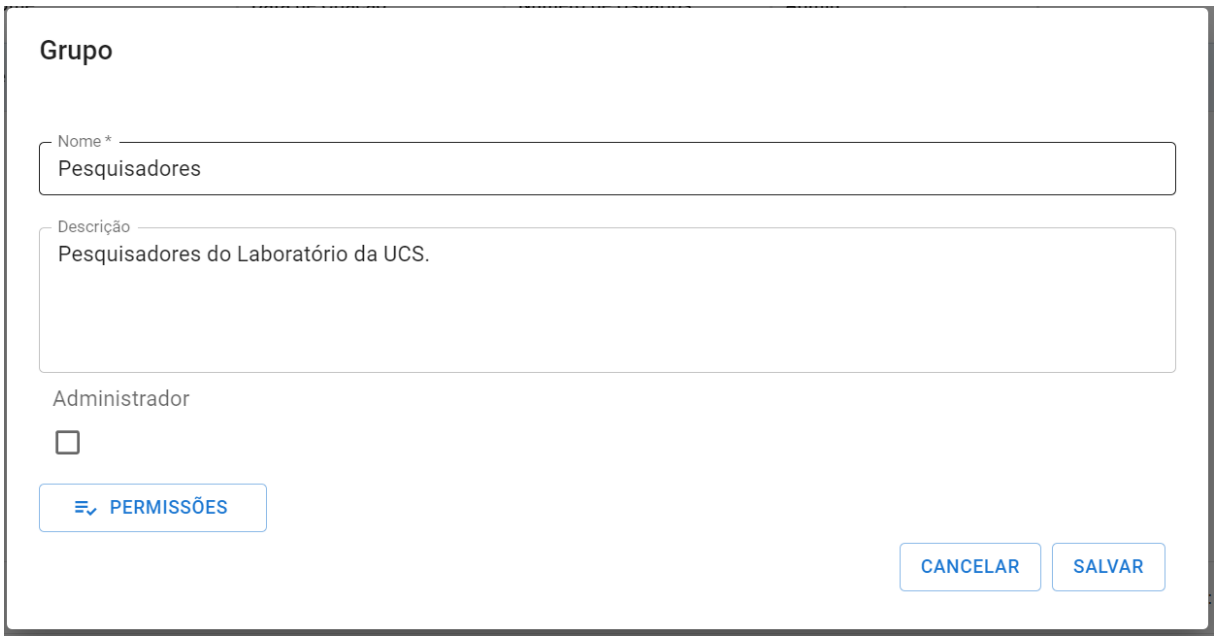
Fonte: O Autor (2022).

O caso de uso possui o mesmo diagrama de robustez apresentado na Figura 31.

4.5.2.3 Caso de Uso UC-11 Gerenciar Grupo

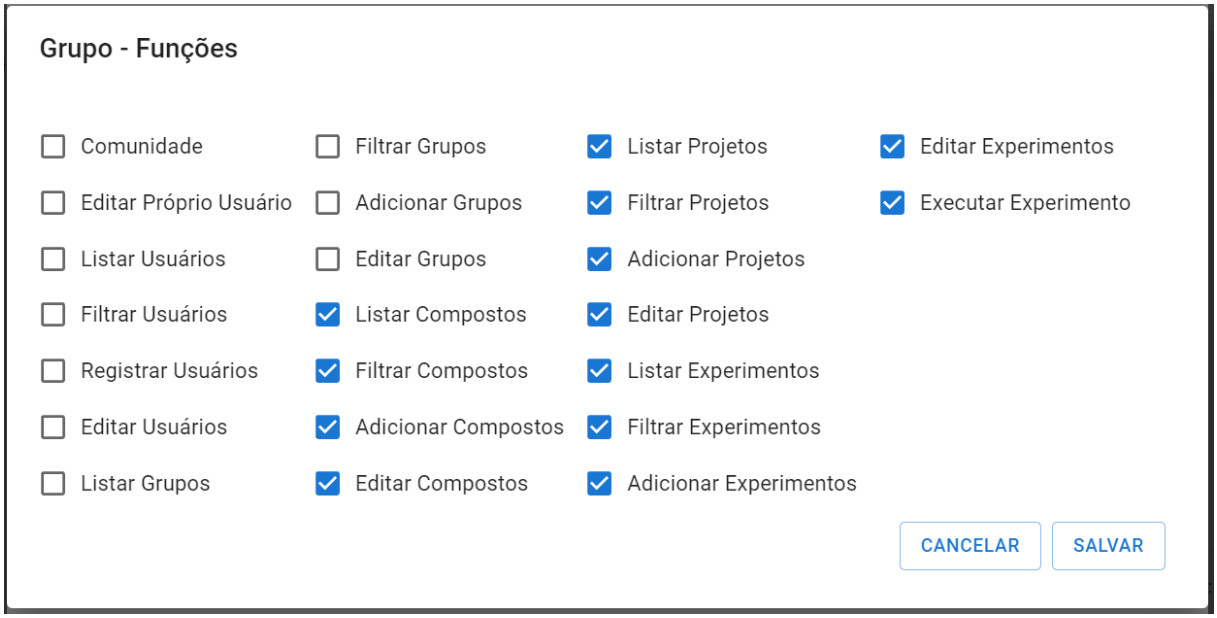
O caso de uso 'UC-11 Gerenciar Grupo' atende ao requisito funcional RF-08,RF-09 e RF-10. A Figura 33 e a Figura 34 apresentam as interface gráficas deste caso de uso.

Figura 33 – Recorte da Interface Gráfica 01 UC-11 Gerenciar Grupo



Fonte: O Autor (2022).

Figura 34 – Recorte da Interface Gráfica 02 UC-11 Gerenciar Grupo



Fonte: O Autor (2022).

O Quadro 11 apresenta o detalhamento deste caso de uso.

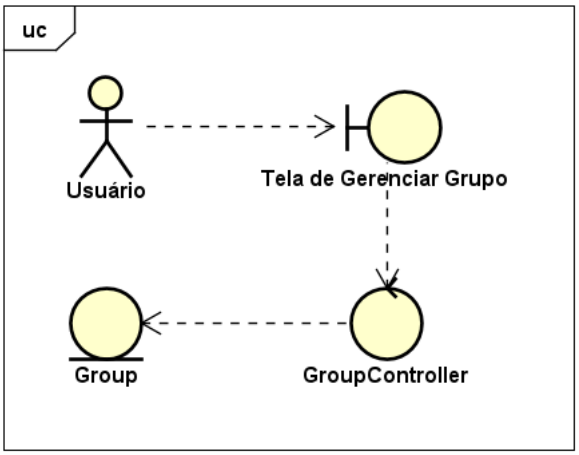
Quadro 11 – Detalhamento UC-11 Gerenciar Grupo

Código e Descrição	UC-11 Gerenciar Grupo
Ator	Usuário
Descrição	Permite que um usuário crie um grupo, altere e visualize suas propriedades e altere e visualize suas permissões.
RF Associados	RF-08, RF-09 e RF-10
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir as permissões 'Listar Grupos' e 'Gerenciar Grupo'. 3. Realização do caso de uso 'UC-09 Listar Grupos'.
Pós-Condições	**Criação** Novo grupo cadastrado no sistema. **Edição e Visualização** Propriedades de um grupo exibidas e/ou alteradas.
Fluxo principal	1. Usuário seleciona a entrada 'Grupos' no menu lateral. 2. Sistema exibe a tela 'Grupos'. **Criação** 3. Usuário pressiona botão de adição. 4. Sistema exibe janela 'Grupo' com propriedades a serem definidas: 'Nome', 'Administrador' e 'Descrição'. 5. Usuário preenche 'Nome', 'Administrador', 'Descrição'. 6. Caso 'Administrador' não estiver selecionado o sistema exibe o botão 'Permissões'. 6.1. Usuário clica no botão 'Permissões'. 6.2. Sistema exibe janela 'Grupo - Funções'. 6.3. Usuário seleciona as permissões/funções desejadas. 6.4. Usuário pressiona o botão 'Salvar'. 6.5 Sistema fecha janela 'Grupo - Funções'. 7. Usuário pressiona o botão 'Salvar'. 8. Sistema valida dados. 9. Sistema realiza criação de registro de novo grupo. 10. Sistema fecha janela 'Grupo'. **Edição e Visualização** 1. Usuário clica no ícone de edição do grupo que deseja visualizar e/ou alterar. 2. Sistema exibe janela 'Grupo' com propriedades da entidade : 'Nome', 'Administrador' e 'Descrição'. 3. Usuário altera as propriedades que desejar entre 'Nome', 'Administrador' e 'Descrição'. 4. Caso o Grupo não seja administrador, o sistema exibe o botão 'Permissões'. 4.1. Usuário clica no botão 'Permissões'. 4.2. Sistema exibe janela 'Grupo - Funções'. 4.3. Usuário seleciona as permissões/funções desejadas. 4.4 Usuário pressiona o botão 'Salvar'. 4.5 Sistema fecha janela 'Grupo - Funções'. 5. Usuário pressiona o botão 'Salvar'. 6. Sistema valida dados. 7. Sistema realiza edição de registro do grupo. 8. Sistema fecha janela 'Grupo'.
Fluxos alternativos	**Apenas Visualização ou Cancelamento** 1. Na janela 'Grupo' e/ou 'Grupo - Funções' o usuário pressiona o botão cancelar. 2. Sistema fecha a respectiva janela. **Validação de Dados** 1.a) Dados repetidos : Sistema informa que o nome já está cadastrado. 1.b) Dados obrigatórios : Sistema informa que o nome é obrigatório.

Fonte: O Autor (2022).

A Figura 35 apresenta o diagrama de robustez deste caso de uso.

Figura 35 – Diagrama de Robustez 01 UC-11 Gerenciar Grupo



Fonte: O Autor (2022).

4.5.3 DETALHAMENTO DE CASOS DE USO - COMPOSTO

Nesta seção, serão detalhados os caso de uso relacionados principalmente ao domínio Compound.

A Figura 36 lista os códigos e títulos destes casos de uso.

Figura 36 – Casos de Uso - Composto

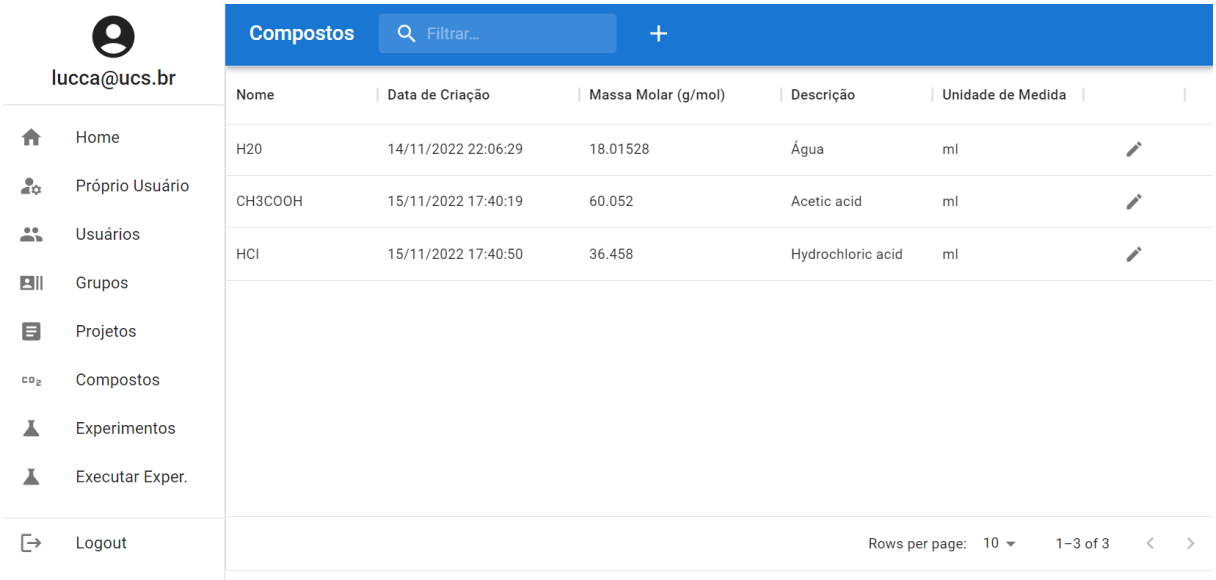
Código	Título
UC-15	Listar Compostos
UC-16	Filtrar Compostos
UC-17	Gerenciar Composto

Fonte: O Autor (2022).

4.5.3.1 Caso de Uso UC-15 Listar Compostos

O caso de uso ’UC-15 Listar Compostos’ atende ao requisito funcional RF-18. A Figura 37 apresenta a interface gráfica deste caso de uso.

Figura 37 – Interface Gráfica UC-15 Listar Compostos



Fonte: O Autor (2022).

O Quadro 12 apresenta o detalhamento deste caso de uso.

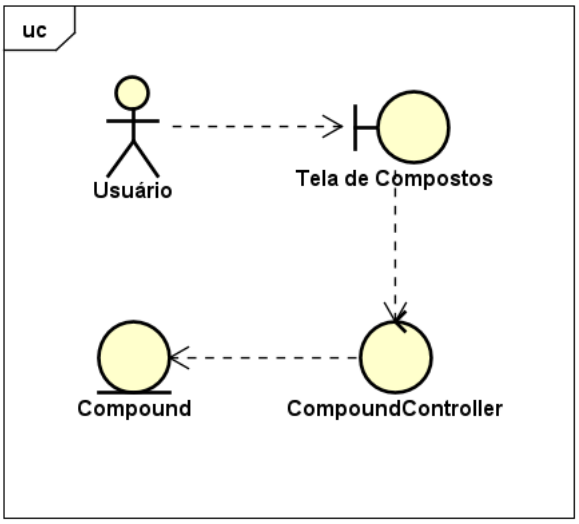
Quadro 12 – Detalhamento UC-15 Listar Compostos

Código e Descrição	UC-15 Listar Compostos
Ator	Usuário
Descrição	Permite que um usuário realize uma listagem dos compostos cadastrados no sistema.
RF Associados	RF-18
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Compostos.
Pós-Condições	Compostos do sistema listados.
Fluxo principal	1. Usuário selecione a entrada 'Compostos' no menu lateral. 2. Sistema exibe a tela 'Compostos'. 3. Sistema realiza busca dos compostos cadastros. 4. Sistema exibe tabela dos compostos cadastrados com as colunas 'Nome', 'Data de Criação', 'Massa Molar (g/mol)', 'Descrição' e 'Unidade de Media'.
Fluxos alternativos	

Fonte: O Autor (2022).

A Figura 38 apresenta o diagrama de robustez deste caso de uso.

Figura 38 – Diagrama de Robustez UC-15 Listar Compostos

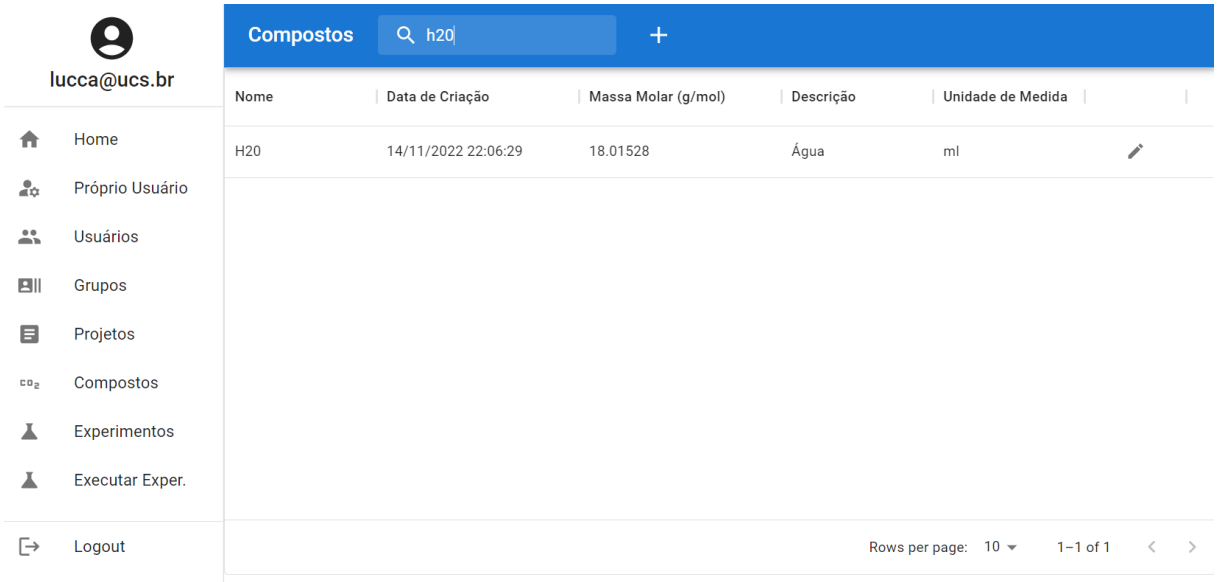


Fonte: O Autor (2022).

4.5.3.2 Caso de Uso UC-16 Filtrar Compostos

O caso de uso 'UC-10 Filtrar Grupos' atende ao requisito funcional RF-19. A Figura 39 apresenta a interface gráfica deste caso de uso.

Figura 39 – Interface Gráfica UC-16 Filtrar Compostos



Fonte: O Autor (2022).

O Quadro 13 apresenta o detalhamento deste caso de uso.

Quadro 13 – Detalhamento UC-16 Filtrar Compostos

Código e Descrição	UC-16 Filtrar Compostos
Ator	Usuário
Descrição	Permite que um usuário realize uma filtragem dos compostos cadastrados no sistema
RF Associados	RF-19
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Compostos' e 'Filtrar Compostos'. 3. Realização do caso de uso 'UC-15 Listar Compostos'.
Pós-Condições	Compostos que correspondem ao filtro listados.
Fluxo principal	1. Usuário informa um filtro na caixa de pesquisa. 2. Usuário pressiona o Enter. 3. Sistema realiza busca dos compostos cadastros que possuam o 'nome' ou 'descrição' semelhante ao filtro informado. 4. Sistema exibe tabela dos compostos filtrados com as colunas 'Nome', 'Data de Criação', 'Massa Molar (g/mol)' e 'Descrição'.
Fluxos alternativos	

Fonte: O Autor (2022).

O caso de uso possui o mesmo diagrama de robustez apresentado na Figura 38.

4.5.3.3 Caso de Uso UC-17 Gerenciar Composto

O caso de uso 'UC-17 Gerenciar Composto' atende ao requisito funcional RF-17. A Figura 40 apresenta a interface gráfica deste caso de uso.

Figura 40 – Recorte da Interface Gráfica UC-17 Gerenciar Composto



Fonte: O Autor (2022).

O Quadro 14 apresenta o detalhamento deste caso de uso.

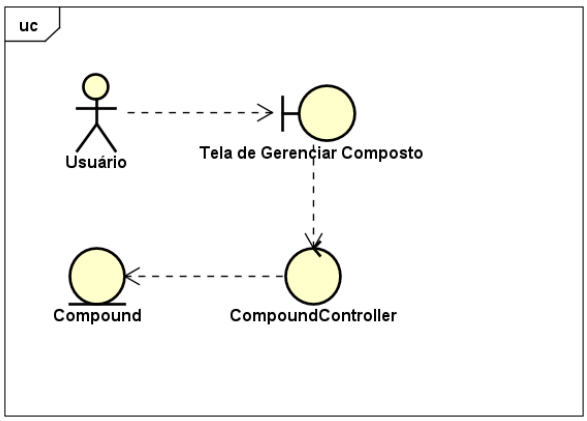
Quadro 14 – Detalhamento UC-17 Gerenciar Composto

Código e Descrição	UC-17 Gerenciar Composto
Ator	Usuário
Descrição	Permite que um usuário crie um composto, além de permitir que o mesmo altere e visualize suas propriedades.
RF Associados	RF-17
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Compostos' e 'Gerenciar Compostos'. 3. Realização do caso de uso 'UC-15 Listar Compostos'.
Pós-Condições	**Criação** Novo composto cadastrado no sistema. **Edição e Visualização** Propriedades de um composto exibidas e/ou alteradas.
Fluxo principal	**Criação** 1. Usuário seleciona a entrada 'Compostos' no menu lateral. 2. Sistema exibe a tela 'Compostos'. 3. Usuário pressiona botão de adição. 4. Sistema exibe janela 'Composto' com propriedades a serem definidas : 'Nome', 'Massa Molar', 'Descrição' e 'Unidade de Medida'. 5. Usuário preenche/seleciona as propriedades. 6. Caso o usuário deseje adicionar uma Unidade de Medida: 6.1 Usuário pressiona o botão de adição. 6.2 Sistema exibe janela 'Unidade de Medida' com um único campo a ser preenchido. 6.3 Usuário preenche o campo com valor da 'Unidade de Medida'. 6.4 Usuário pressiona o botão Adicionar'. 6.5 Sistema cria registro de 'Unidade de Medida'. 6.6 Sistema fecha janela 'Unidade de Medida'. 7. Usuário pressiona o botão 'Salvar'. 8. Sistema valida dados. 9. Sistema realiza criação de registro de novo composto. 10. Sistema fecha janela 'Composto'. **Edição e Visualização** 1. Usuário clica no ícone de edição do composto que deseja visualizar e/ou alterar. 2. Sistema exibe janela 'Compostos' com propriedades do grupo : "Nome', 'Massa Molar', 'Descrição' e Unidade de Medida'. 3. Usuário altera as propriedades que desejar. 4. Usuário pressiona o botão 'Salvar'. 5. Sistema valida dados. 6. Sistema realiza edição de registro do composto. 7. Sistema fecha janela 'Composto'.
Fluxos alternativos	**Validação de Dados** 1.a) Dados inválidos. 1.1. Sistema informa que o nome já está cadastrado. 1.2. Sistema informa que a massa molar deve ser um numeral. 1.b) Dados obrigatórios. 1.1. Sistema informa que o nome é obrigatório. 1.2. Sistema informa que a massa molar é obrigatória. 1.3. Sistema informa que unidade de medida é obrigatória.

Fonte: O Autor (2022).

A Figura 41 apresenta o diagrama de robustez deste caso de uso.

Figura 41 – Diagrama de Robustez UC-17 Gerenciar Composto



Fonte: O Autor (2022).

4.5.4 DETALHAMENTO DE CASOS DE USO - PROJETO

Nesta seção, serão detalhados os caso de uso relacionados principalmente ao domínio Project.

A Figura 42 lista os códigos e títulos destes casos de uso.

Figura 42 – Casos de Uso - Projeto

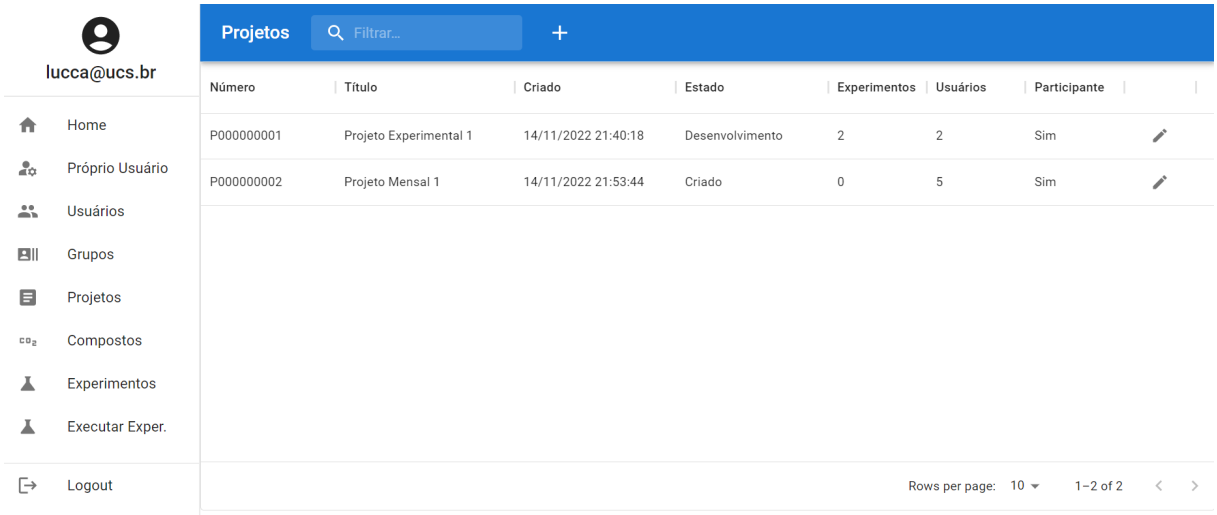
Código	Título
UC-11	Listar Projetos
UC-12	Filtrar Projetos
UC-13	Gerenciar Projeto

Fonte: O Autor (2022).

4.5.4.1 Caso de Uso UC-12 Listar Projetos

O caso de uso 'UC-12 Listar Projetos' atende ao requisito funcional RF-15. A Figura 43 apresenta a interface gráfica deste caso de uso.

Figura 43 – Interface Gráfica UC-12 Listar Projetos



Fonte: O Autor (2022).

O Quadro 15 apresenta o detalhamento deste caso de uso.

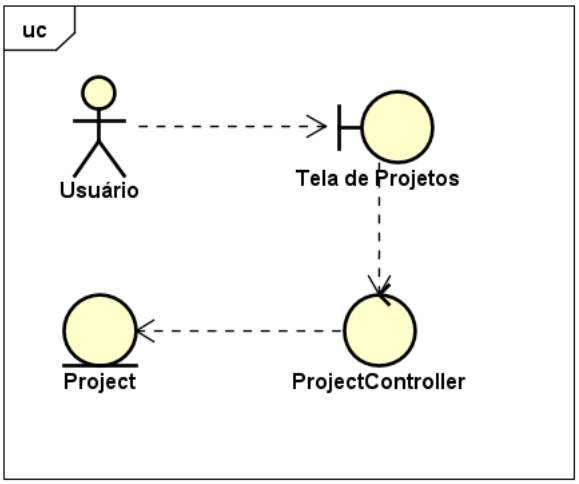
Quadro 15 – Detalhamento UC-12 Listar Projetos

Código e Descrição	UC-12 Listar Projetos
Ator	Usuário
Descrição	Permite que um usuário realize uma listagem dos projetos cadastrados no sistema que o mesmo possua permissão de acesso.
RF Associados	RF-15
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Projetos'.
Pós-Condições	Projetos acessíveis pelo usuário listados
Fluxo principal	1. Usuário selecione a entrada 'Projetos' no menu lateral. 2. Sistema exibe a tela 'Projetos'. 3. Sistema realiza busca dos projetos cadastros visíveis para o usuário autenticado. 4. Sistema exibe tabela dos projetos cadastrados com as colunas 'Número', 'Título', 'Criado', 'Estado', 'Experimentos', 'Usuários' e 'Participante'.
Fluxos alternativos	

Fonte: O Autor (2022).

A Figura 44 apresenta o diagrama de robustez deste caso de uso.

Figura 44 – Diagrama de Robustez UC-12 Listar Projetos

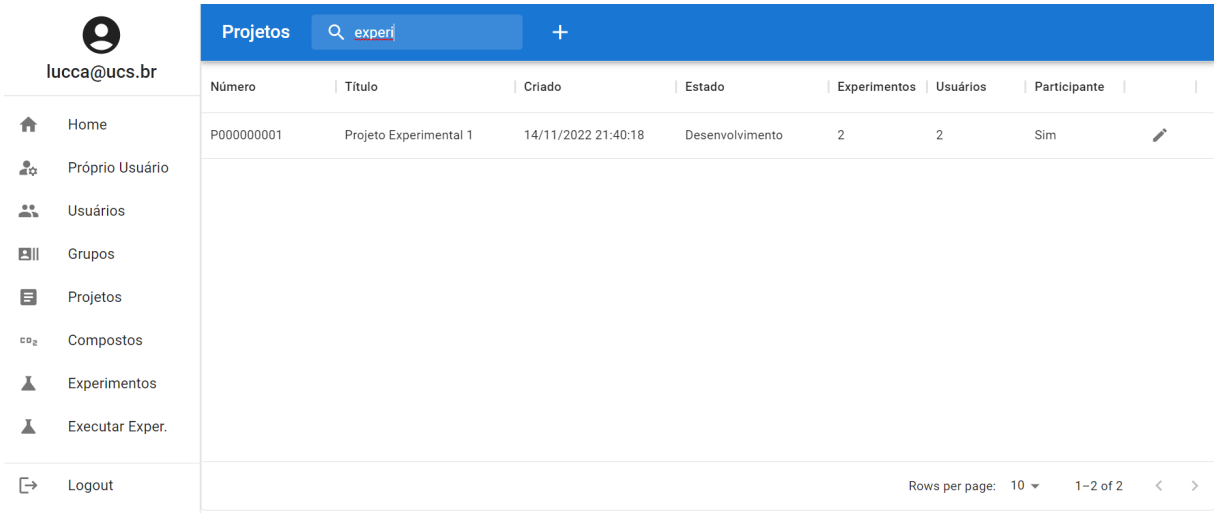


Fonte: O Autor (2022).

4.5.4.2 Caso de Uso UC-13 Filtrar Projetos

O caso de uso 'UC-13 Filtrar Projetos' atende ao requisito funcional RF-16. A Figura 45 apresenta a interface gráfica deste caso de uso.

Figura 45 – Interface Gráfica UC-13 Filtrar Projetos



Fonte: O Autor (2022).

O Quadro 16 apresenta o detalhamento deste caso de uso.

Quadro 16 – Detalhamento UC-13 Filtrar Projetos

Código e Descrição	UC-13 Filtrar Projetos
Ator	Usuário
Descrição	Permite que um usuário realize uma filtragem dos projetos cadastrados no sistema que o mesmo possua permissão de acesso.
RF Associados	RF-16
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Projetos' e 'Filtrar Projetos'. 3. Realização do caso de uso 'UC-12 Listar Projetos'.
Pós-Condições	Projetos que correspondem ao filtro listados.
Fluxo principal	1. Usuário informa um filtro na caixa de pesquisa. 2. Usuário pressiona o Enter. 3. Sistema realiza busca dos projetos cadastros que possuam 'número' ou 'título' semelhante ao do filtro informado e que sejam acessíveis pelo usuário. 4. Sistema exibe tabela dos projetos filtrados com as colunas 'Número', 'Título', 'Criado', 'Estado', 'Experimentos', 'Usuários' e 'Participante'.
Fluxos alternativos	

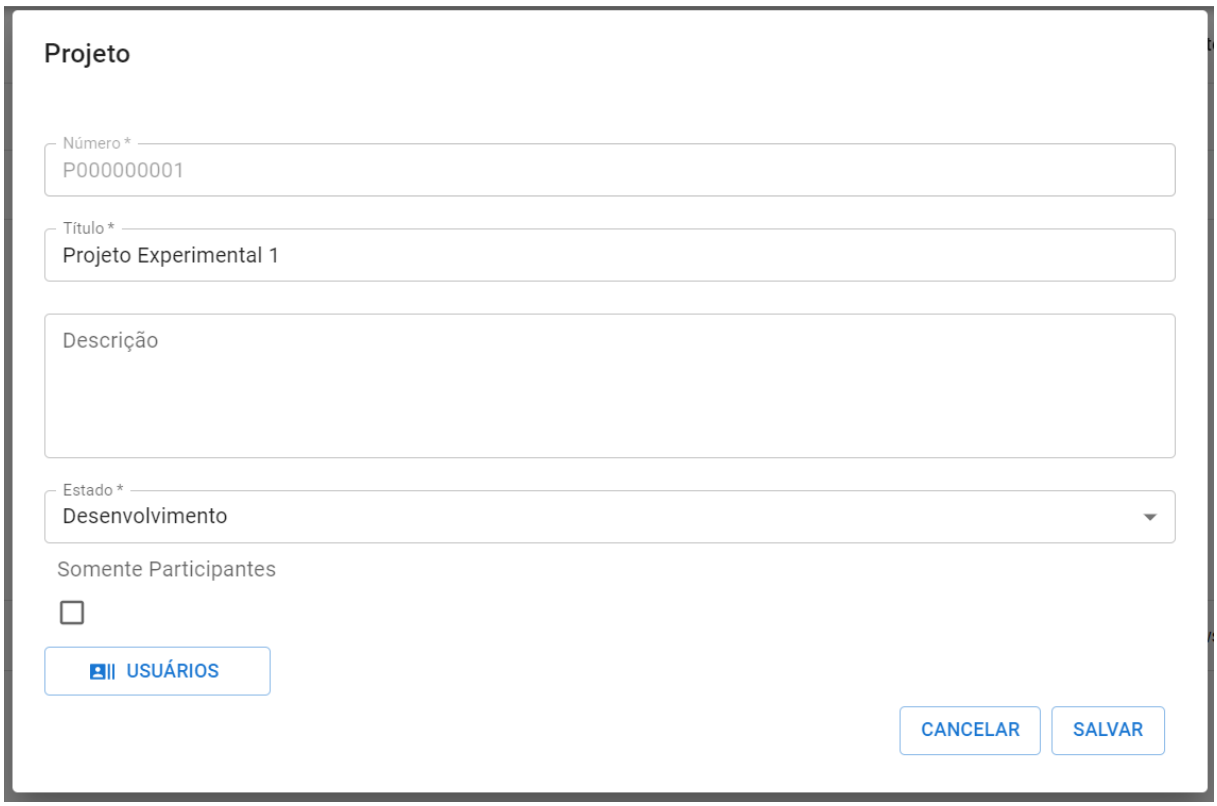
Fonte: O Autor (2022).

O caso de uso possui o mesmo diagrama de robustez apresentado na Figura 44.

4.5.4.3 Caso de Uso UC-14 Gerenciar Projeto

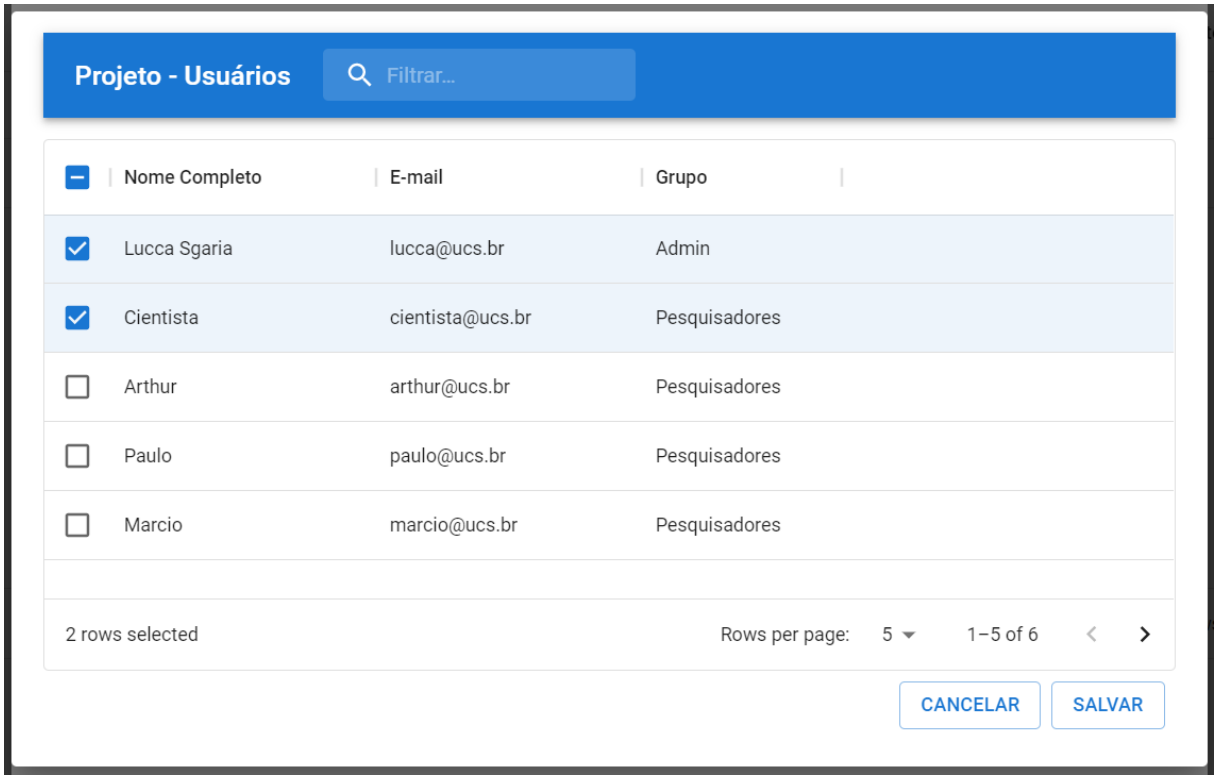
O caso de uso 'UC-14 Gerenciar Projeto' atende ao requisito funcional RF-13 E RF-14. A Figura 46 e a Figura 47 apresentam as interface gráficas deste caso de uso.

Figura 46 – Recorte da Interface Gráfica 01 UC-14 Gerenciar Projeto



Fonte: O Autor (2022).

Figura 47 – Recorte da Interface Gráfica 02 UC-14 Gerenciar Projeto



Fonte: O Autor (2022).

O Quadro 17 apresenta o detalhamento deste caso de uso.

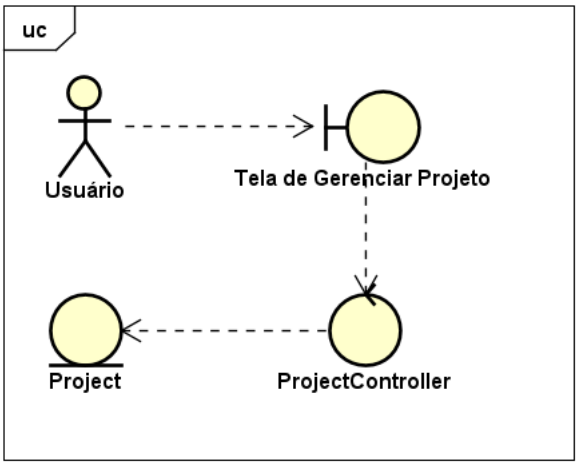
Quadro 17 – Detalhamento UC-14 Gerenciar Projeto

Código e Descrição	UC-17 Gerenciar Composto
Ator	Usuário
Descrição	Permite que um usuário crie, altere e visualize um projeto. Assim como permite que um usuário edite e visualize os usuários de um projeto.
RF Associados	RF-13 e RF-14
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O projeto do usuário deve possuir a permissão 'Listar Projetos' e 'Gerenciar Projeto'. 3. Realização do caso de uso 'UC-12 Listar Projetos'.
Pós-Condições	**Criação** Novo projeto cadastrado no sistema. **Edição e Visualização** Propriedades de um projeto exibidas e/ou alteradas.
Fluxo principal	**Criação** 1. Usuário seleciona a entrada 'Projetos' no menu lateral. 2. Sistema exibe a tela 'Projetos'. 3. Usuário pressiona botão de adição. 4. Sistema exibe janela 'Projeto' com propriedades: 'Número', 'Título', 'Descrição', 'Estado' e 'Somente Participantes' e o botão 'Usuários'. 5. Usuário define as propriedades. 6. Usuário clica no botão 'Usuários'. 6.1 O sistema exibe a janela 'Projeto - Usuários'. 6.2 Usuário seleciona os usuários desejados. 6.3 Usuário pressiona o botão 'Salvar'. 6.4 Sistema fecha janela 'Projeto - Usuários'. 7. Usuário pressiona o botão 'Salvar'. 8. Sistema valida dados. 9. Sistema realiza criação de registro de novo projeto. 10. Sistema fecha janela 'Projeto'. **Edição e Visualização** 1. Usuário clica no ícone de edição. 2. Sistema exibe janela 'Projeto' com propriedades do projeto : 'Número', 'Título', 'Descrição', 'Estado' e 'Somente Participantes' e o botão 'Usuários'. 3. Usuário altera as propriedades desejadas. 4. Usuário clica no botão 'Usuários'. 4.1 O sistema exibe a janela 'Projeto - Usuários'. 4.2 Usuário seleciona os usuários desejados. 4.3 Usuário pressiona o botão 'Salvar'. 4.4 Sistema fecha janela 'Projeto - Usuários'. 5. Usuário pressiona o botão 'Salvar'. 6. Sistema valida dados. 7. Sistema realiza edição de registro do projeto. 8. Sistema fecha janela 'Projeto'.
Fluxos alternativos	**Apenas Visualização ou Cancelamento** 1. Na janela 'Projeto' e/ou Projeto - Usuários' o usuário pressiona o botão cancelar. 2. Sistema fecha a respectiva janela. **Validação de Dados** 1.a) Dados repetidos. 1.1. Sistema informa que o título já está cadastrado. 1.b) Dados obrigatórios. 1.1. Sistema informa que o título é obrigatório.

Fonte: O Autor (2022).

A Figura 48 apresenta o diagrama de robustez deste caso de uso.

Figura 48 – Diagrama de Robustez UC-14 Gerenciar Projeto



Fonte: O Autor (2022).

4.5.5 DETALHAMENTO DE CASOS DE USO - EXPERIMENTO

Nesta seção, serão detalhados os caso de uso relacionados principalmente aos domínios Experiment, ExperimentLines, Comment e Attachment.

A Figura 49 lista os códigos e títulos destes casos de uso.

Figura 49 – Casos de Uso - Experimento

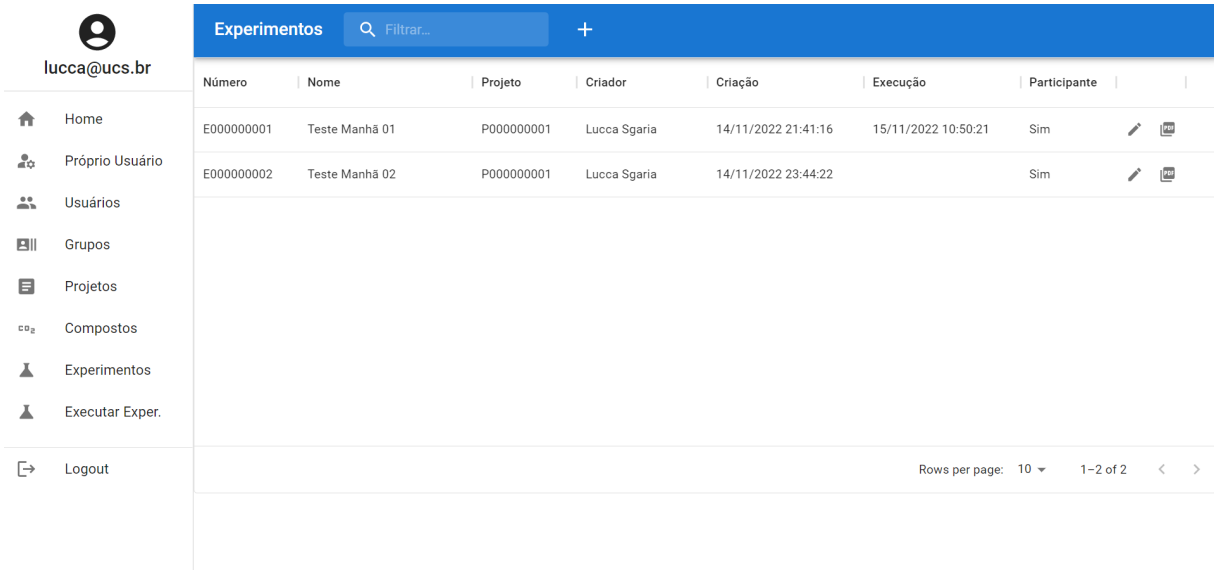
Código	Título
UC-18	Listar Experimentos
UC-19	Filtrar Experimentos
UC-20	Gerenciar Experimento
UC-21	Gerenciar Comentários de Experimento
UC-22	Gerenciar Anexos de Experimento
UC-23	Gerenciar Linhas de Experimento
UC-24	Executar um Experimento
UC-25	Exportar o conteúdo de um Experimento

Fonte: O Autor (2022).

4.5.5.1 Caso de Uso UC-18 Listar Experimentos

O caso de uso ’UC-18 Listar Experimentos’ atende ao requisito funcional RF-24. A Figura 50 apresenta a interface gráfica deste caso de uso.

Figura 50 – Interface Gráfica UC-18 Listar Experimentos



Fonte: O Autor (2022).

O Quadro 18 apresenta o detalhamento deste caso de uso.

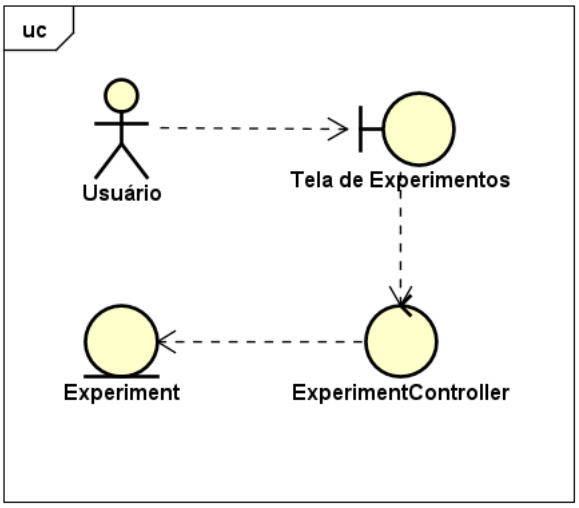
Quadro 18 – Detalhamento UC-18 Listar Experimentos

Código e Descrição	UC-18 Listar Experimentos
Ator	Usuário
Descrição	Permite que um usuário realize uma listagem dos experimentos cadastrados no sistema que o mesmo possua permissão de acesso.
RF Associados	RF-24
Pré-Condições	1.O usuário deve estar autenticado no sistema. 2.O grupo do usuário deve possuir a permissão ’Listar Experimentos’.
Pós-Condições	Experimentos acessíveis pelo usuário listados.
Fluxo principal	1. Usuário selecione a entrada ‘Experimentos’ no menu lateral. 2. Sistema exibe a tela ‘Experimentos’. 3. Sistema realiza busca dos experimentos cadastros visíveis para o usuário autenticado . 4. Sistema exibe tabela dos experimentos cadastrados com as colunas ‘Número’, ’Nome’, ’Projeto’, ’Criador’, ’Criação’, ’Execução’ e ’Participante’.
Fluxos alternativos	

Fonte: O Autor (2022).

A Figura 51 apresenta o diagrama de robustez deste caso de uso.

Figura 51 – Diagrama de Robustez UC-18 Listar Experimentos

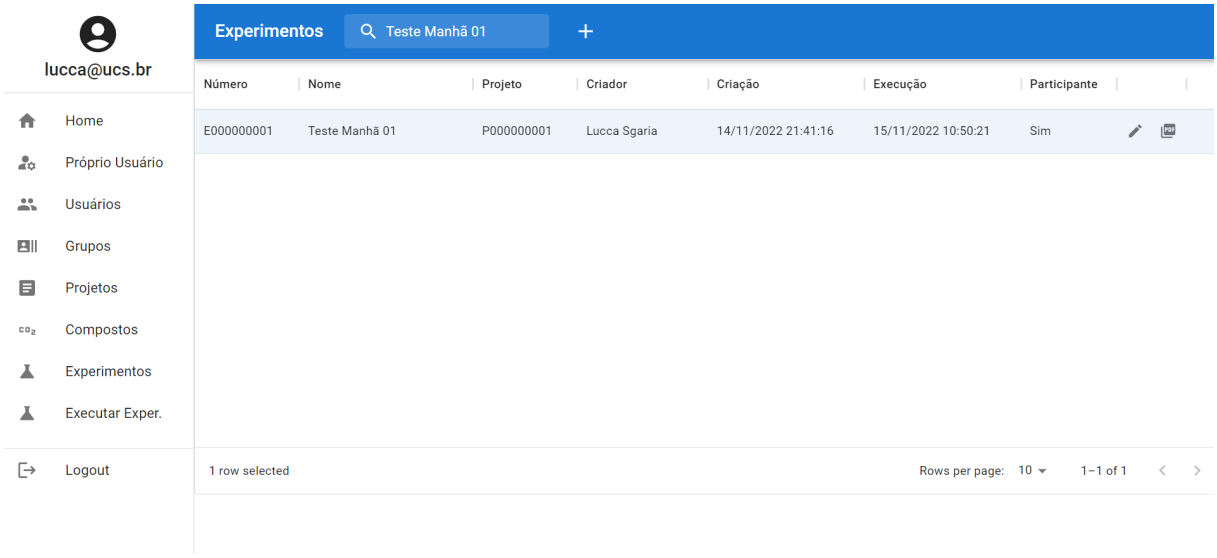


Fonte: O Autor (2022).

4.5.5.2 Caso de Uso UC-19 Filtrar Experimentos

O caso de uso 'UC-19 Filtrar Experimentos' atende ao requisito funcional RF-25. A Figura 52 apresenta a interface gráfica deste caso de uso.

Figura 52 – Interface Gráfica UC-19 Filtrar Experimentos



Fonte: O Autor (2022).

O Quadro 19 apresenta o detalhamento deste caso de uso.

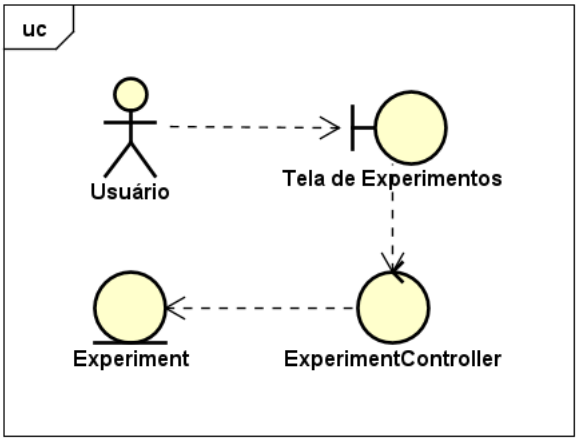
Quadro 19 – Detalhamento UC-19 Filtrar Experimentos

Código e Descrição	UC-19 Filtrar Experimentos
Ator	Usuário
Descrição	Permite que um usuário realize uma filtragem dos experimentos cadastrados no sistema que o mesmo possua permissão de acesso.
RF Associados	RF-25
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Experimentos' e 'Filtrar Experimentos'. 3. Realização do caso de uso 'UC-18 Listar Experimentos'
Pós-Condições	Experimentos que correspondem ao filtro listados.
Fluxo principal	1. Usuário informa um filtro na caixa de pesquisa. 2. Usuário pressiona o Enter. 3. Sistema realiza busca dos experimentos cadastros que possuam número ou nome semelhante ao do filtro informado e que sejam acessíveis pelo usuário. 4. Sistema exibe tabela dos experimentos cadastrados com as colunas 'Número', 'Nome', 'Projeto', 'Criador', 'Criação', 'Execução' e 'Participante'.
Fluxos alternativos	

Fonte: O Autor (2022).

A Figura 53 apresenta o diagrama de robustez deste caso de uso.

Figura 53 – Diagrama de Robustez UC-19 Filtrar Experimentos

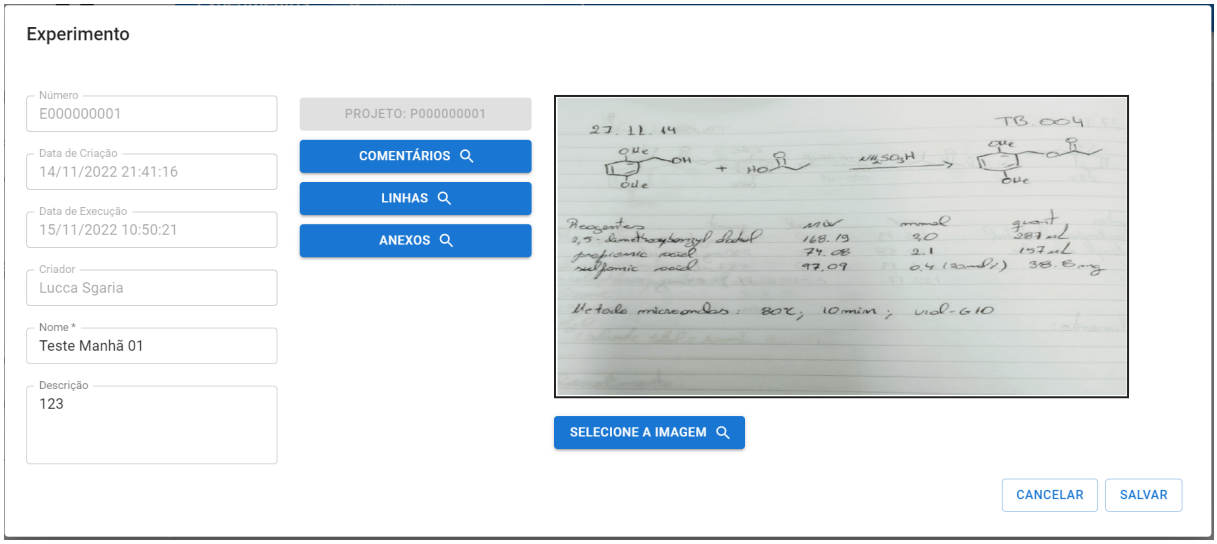


Fonte: O Autor (2022).

4.5.5.3 Caso de Uso UC-20 Gerenciar Experimento

O caso de uso 'UC-20 Gerenciar Experimento' atende ao requisito funcional RF-20. A Figura 54 apresenta a interface gráfica deste caso de uso.

Figura 54 – Recorte da Interface Gráfica UC-20 Gerenciar Experimento



Fonte: O Autor (2022).

O Quadro 20 apresenta o detalhamento deste caso de uso.

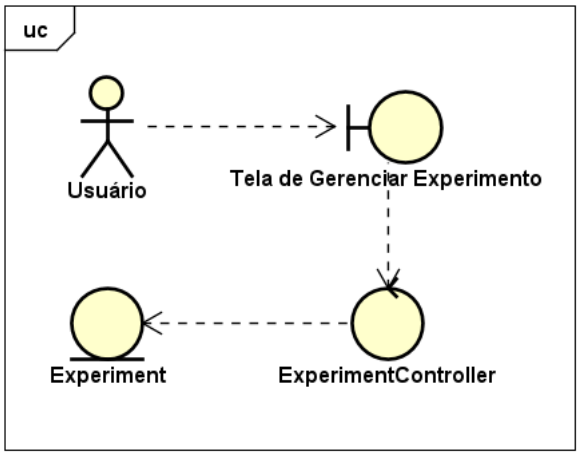
Quadro 20 – Detalhamento UC-20 Gerenciar Experimento

Código e Descrição	UC-20 Gerenciar Experimento
Ator	Usuário
Descrição	Permite que um usuário crie um projeto, além de permitir que o mesmo altere e visualize suas propriedades.
RF Associados	RF-20
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Experimentos' e 'Gerenciar Experimento'. 3. Realização do caso de uso 'UC-18 Listar Experimentos'
Pós-Condições	**Criação** Novo experimento cadastrado no sistema **Visualização e Edição** Propriedades de um experimento exibidas e/ou alteradas.
Fluxo principal	**Criação** 1. Usuário seleciona a entrada 'Experimentos' no menu lateral. 2. Sistema exibe a tela 'Experimentos'. 3. Usuário pressiona botão de adição. 4. Sistema exibe janela 'Experimento' com propriedades a serem definidas: 'Nome', 'Descrição' e 'Imagem 'Principal' e botão 'Projeto'. 5. Usuário define as propriedades. 6. Usuário pressiona botão 'Projeto'. 6.1. Sistema exibe a janela 'Experimento - Projeto'. 6.2. Sistema exibe tabela com projetos que o usuário faz parte. 6.3. Usuário seleciona o projeto do Experimento. 6.4. Usuário pressiona botão 'Salvar'. 6.5. Sistema fecha janela 'Experimento - Projeto'. 7. Usuário pressiona o botão 'Salvar' 8. Sistema valida dados 9. Sistema realiza criação de registro de novo experimento. 10. Sistema fecha janela 'Experimento' **Visualização e Edição** 1. Usuário clica no ícone de edição do experimento que deseja visualizar e/ou alterar. 2. Sistema exibe janela 'Experimento' com propriedades a serem definidas: 'Nome', 'Descrição' e 'Imagem 'Principal' e exibe o 'Número Gerado', 'Data de Criação', 'Data de Execução' e 'Criador' do experimento. 3. Usuário altera as propriedades que desejar entre 'Nome', 'Descrição' e Imagem 'Principal'. 4. Sistema exibe botão "Anexos" acompanhado de um ícone de lupa. Permite a execução do caso de uso UC-22 Gerenciar Anexos de Experimento. 5. Sistema exibe botão "Comentários" acompanhado de um ícone de lupa. Permite a execução do caso de uso UC-21 Gerenciar Comentários de Experimento. 6. Sistema exibe botão "Linhas" acompanhado de um ícone de lupa. Permite a execução do caso de uso UC-23 Gerenciar Linhas de Experimento. 7. Usuário pressiona o botão 'Salvar' 8. Sistema valida dados 9. Sistema realiza edição de registro do experimento. 10. Sistema fecha janela 'Experimento'
Fluxos alternativos	**Apenas Visualização ou Cancelamento** 1. Na janela 'Experimento' e/ou Experimento - Projeto' o usuário pressiona o botão cancelar. 2. Sistema fecha a respectiva janela. **Validação de Dados** 1.a) Dados obrigatórios. 1.1. Sistema informa que o título é obrigatório. 1.2. Sistema informa que o projeto é obrigatório.

Fonte: O Autor (2022).

A Figura 55 apresenta o diagrama de robustez deste caso de uso.

Figura 55 – Diagrama de Robustez UC-20 Gerenciar Experimento

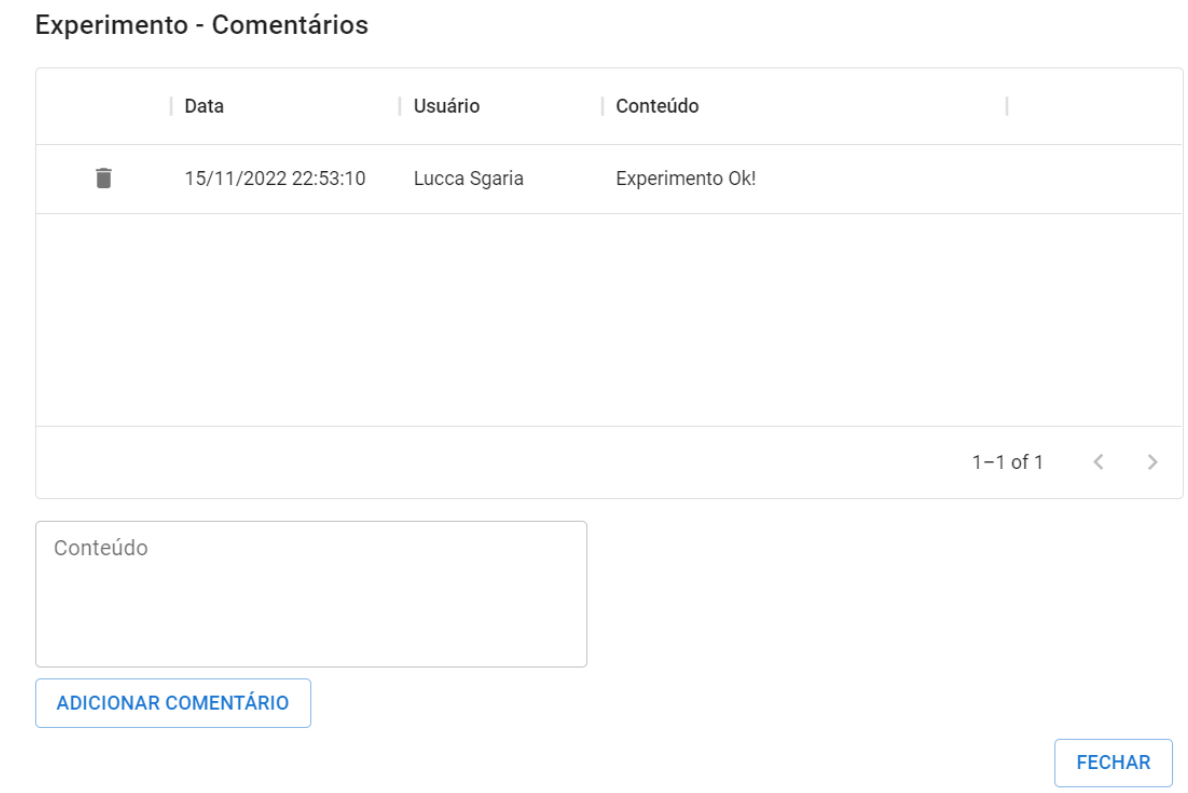


Fonte: O Autor (2022).

4.5.5.4 Caso de Uso UC-21 Gerenciar Comentários de Experimento

O caso de uso ’UC-21 Gerenciar Comentários de Experimento’ atende ao requisito funcional RF-22. A Figura 56 apresenta a interface gráfica deste caso de uso.

Figura 56 – Recorte da Interface Gráfica UC-21 Gerenciar Comentários de Experimento



Fonte: O Autor (2022).

O Quadro 15 apresenta o detalhamento deste caso de uso.

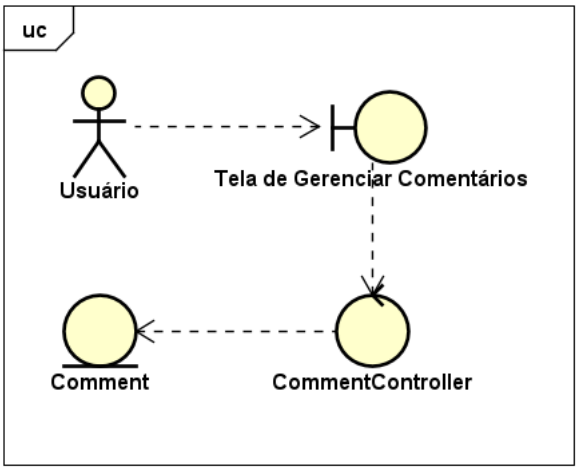
Quadro 21 – Detalhamento UC-21 Gerenciar Comentários de Experimento

Código e Descrição	UC-21 Gerenciar Comentários de Experimento
Ator	Usuário
Descrição	Permite que um usuário crie comentários de experimento, além de permitir que o mesmo visualize suas propriedades.
RF Associados	RF-22
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. Realização do caso de uso 'UC-20 Gerenciar Experimento' ou 'UC-24 Executar um Experimento' 3. O grupo do usuário deve possuir a permissão 'Gerenciar Comentários de Experimento'.
Pós-Condições	**Criação** Novo comentário de experimento cadastrado no sistema. **Listagem** Comentários do Experimento listados. **Exclusão** Comentário do Experimento excluído.
Fluxo principal	1. Usuário clica no botão 'Comentários'. 2. Sistema exibe janela 'Experimento - Comentários' 3. Sistema exibe tabela de comentários do experimento com as colunas "Data", "Usuário" e "Conteúdo" e uma coluna com o ícone de exclusão. 4. Sistema exibe uma caixa de texto intitulada 'Conteúdo' e um botão 'Adicionar Comentário'. **Criação** 5.1. Usuário escreve o comentário desejado na caixa de texto. 5.2. Usuário pressiona o botão 'Adicionar Comentário'. 5.3. Sistema valida dados. 5.4. Sistema cria registro de novo comentário de experimento. 5.5. Sistema atualiza a listagem de comentários. **Exclusão** 5.1. Usuário clica no ícone de exclusão. 5.5. Sistema remove comentário de experimento. 6. Usuário pressiona botão 'Fechar'. 7. Sistema fecha janela 'Experimento - Comentários'.
Fluxos alternativos	**Validação de Dados** 1.a) Dados obrigatórios. 1. Sistema informa que o conteúdo é obrigatório.

Fonte: O Autor (2022).

A Figura 57 apresenta o diagrama de robustez deste caso de uso.

Figura 57 – Diagrama de Robustez UC-21 Gerenciar Comentários de Experimento

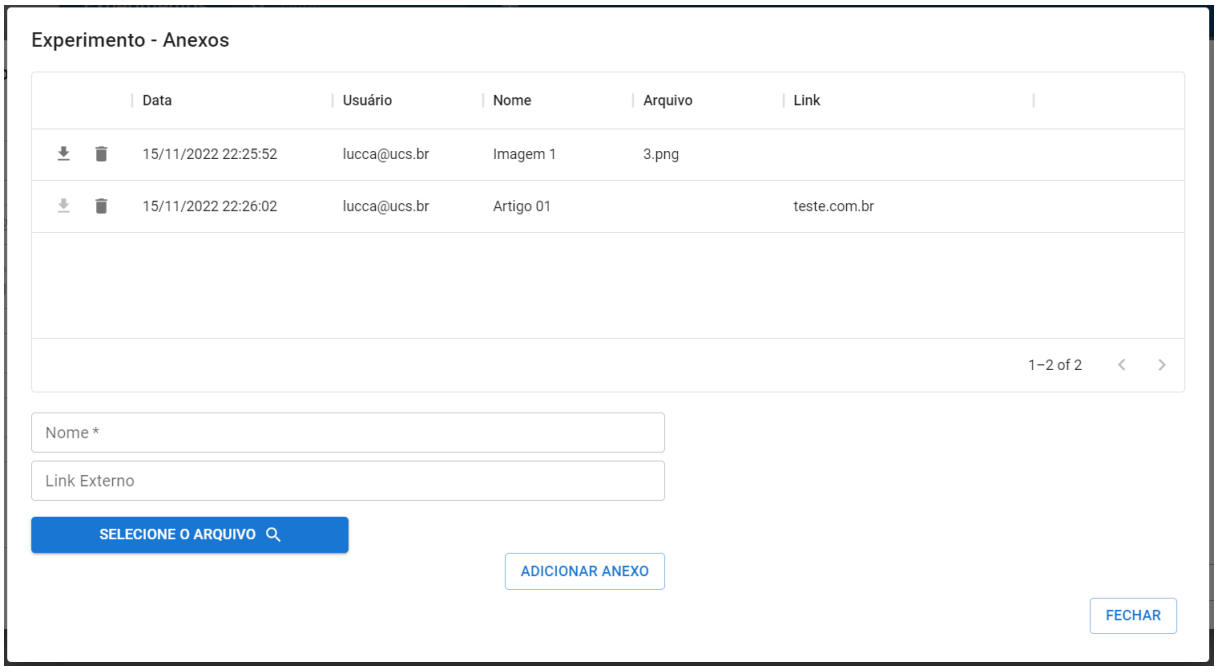


Fonte: O Autor (2022).

4.5.5.5 Caso de Uso UC-22 Gerenciar Anexos de Experimento

O caso de uso 'UC-22 Gerenciar Anexos de Experimento' atende ao requisito funcional RF-21. A Figura 58 apresenta a interface gráfica deste caso de uso.

Figura 58 – Recorte da Interface Gráfica UC-22 Gerenciar Anexos de Experimento



Fonte: O Autor (2022).

O Quadro 22 apresenta o detalhamento deste caso de uso.

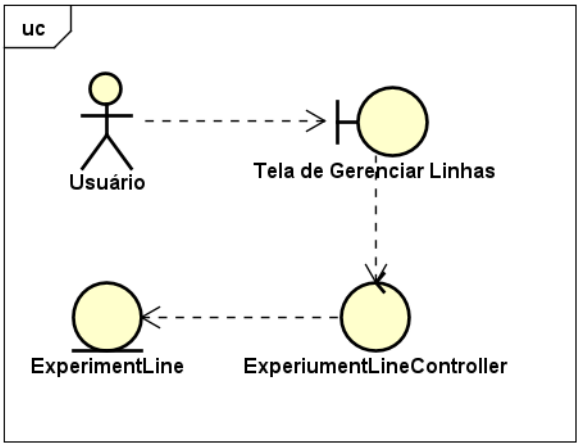
Quadro 22 – Detalhamento UC-22 Gerenciar Anexos de Experimento

Código e Descrição	UC-22 Gerenciar Anexos de Experimento
Ator	Usuário
Descrição	Permite que um usuário crie anexos de um experimento, realize o download do arquivo, além de permitir que o mesmo visualize suas propriedades.
RF Associados	RF-21
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. Realização do caso de uso ‘UC-20 Gerenciar Experimento’ ou ‘UC-24 Executar um Experimento’. 3. O grupo do usuário deve possuir a permissão ‘Gerenciar Anexos de Experimento’.
Pós-Condições	**Criação** Novo anexo de experimento cadastrado no sistema. **Listagem** Anexos do Experimento listados. **Exclusão** Anexo de experimento removido do sistema.
Fluxo principal	1. Usuário clica no botão ‘Anexos’. 2. Sistema exibe janela ‘Experimento - Anexos’. 3. Sistema exibe tabela de anexos do experimento com as colunas "Data", "Usuário", “Nome”, “Arquivo“ e "Link"e coluna com ícones de exclusão e download. 4. Sistema exibe campos ’Nome’, ’Link Externo’, botão ’Selecione o Arquivo’ e botão ’Adicionar Anexo’. **Criação** 5.1. Usuário define as propriedades ‘Nome’ e ‘Link Externo’ e se desejar seleciona um arquivo. 5.2. Usuário pressiona o botão ‘Adicionar anexo’. 5.3. Sistema valida dados. 5.4. Sistema cria registro de novo anexo de experimento. 5.5. Sistema atualiza a listagem de anexos. **Exclusão** 5.1. Usuário clica no ícone de exclusão. 5.2. Sistema remove anexo de experimento. **Download** 5.1. Usuário clica no ícone de download. 5.2. Sistema realiza download do arquivo. 6. Usuário pressiona botão ‘Fechar’. 7. Sistema fecha janela ‘Experimento - Anexos’.
Fluxos alternativos	**Validação de Dados** 1.a) Dados obrigatórios. 1. Sistema informa que o nome é obrigatório.

Fonte: O Autor (2022).

A Figura 59 apresenta o diagrama de robustez deste caso de uso.

Figura 59 – Diagrama de Robustez UC-22 Gerenciar Anexos de Experimento

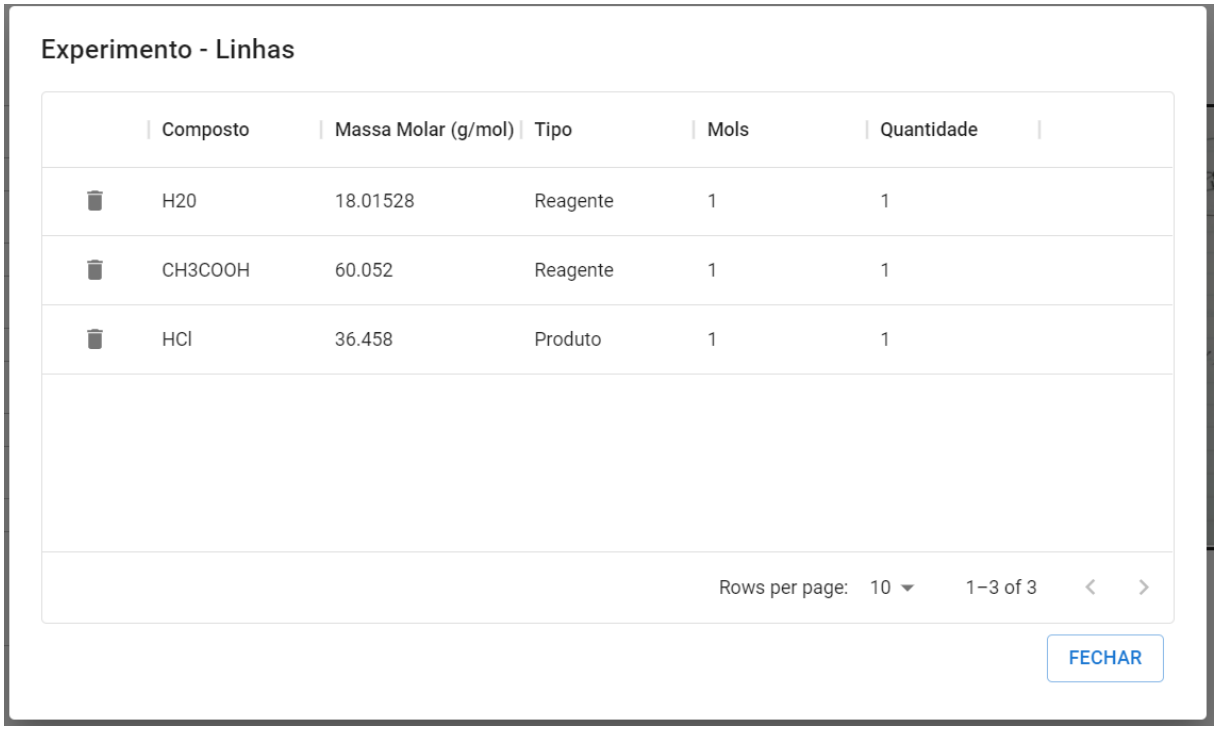


Fonte: O Autor (2022).

4.5.5.6 Caso de Uso UC-23 Gerenciar Linhas de Experimento

O caso de uso 'UC-23 Gerenciar Linhas de Experimento' atende ao requisito funcional RF-23. A Figura 60 apresenta a interface gráfica deste caso de uso.

Figura 60 – Recorte da Interface Gráfica UC-23 Gerenciar Linhas de Experimento



Fonte: O Autor (2022).

O Quadro 23 apresenta o detalhamento deste caso de uso.

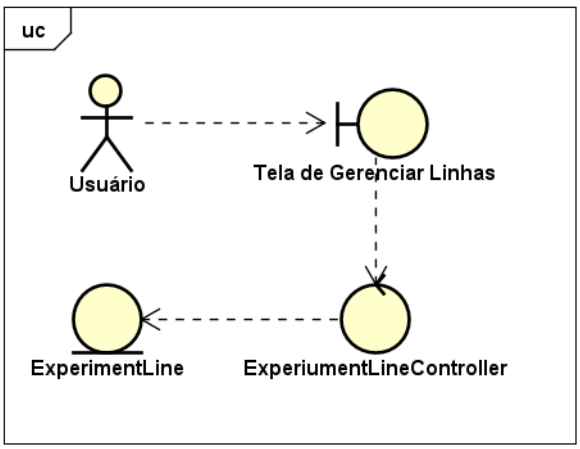
Quadro 23 – Detalhamento UC-23 Gerenciar Linhas de Experimento

Código e Descrição	UC-23 Gerenciar Linhas de Experimento
Ator	Usuário
Descrição	Permite que um usuário visualize, edite e exclua as linhas de um experimento.
RF Associados	RF-23
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. Realização do caso de uso ‘UC-20 Gerenciar Experimento’. 3. O grupo do usuário deve possuir a permissão ‘Gerenciar Linhas de Experimento’.
Pós-Condições	**Visualização e Edição** Propriedades de uma linha de experimento exibidas e/ou alteradas. **Exclusão** Linha de experimento cadastrado no sistema.
Fluxo principal	1. Usuário clica no botão ‘Linhas’. 2. Sistema exibe janela ‘Experimento - Linhas’. 3. Sistema exibe tabela de linhas do experimento com as colunas "Composto", "Massa Molar (g/mol)", “Tipo”, “Mols“ e "Quantidade"e coluna com ícone de exclusão. **Edição** 4.1. Usuário clica na célula ’Mols’ e/ou ’Quantidade’. 4.2. Sistema desbloqueia edição destas células. 4.3. Usuário define novos valores. 4.4. Usuário pressiona ’Enter’. 4.5. Sistema grava novos valores da linha. **Exclusão** 4.1. Usuário clica no ícone de exclusão. 4.2. Sistema remove linha de experimento. 5. Usuário pressiona botão ‘Fechar’. 6. Sistema fecha janela ‘Experimento - Linhas’.
Fluxos alternativos	

Fonte: O Autor (2022).

A Figura 61 apresenta o diagrama de robustez deste caso de uso.

Figura 61 – Diagrama de Robustez UC-23 Gerenciar Linhas de Experimento

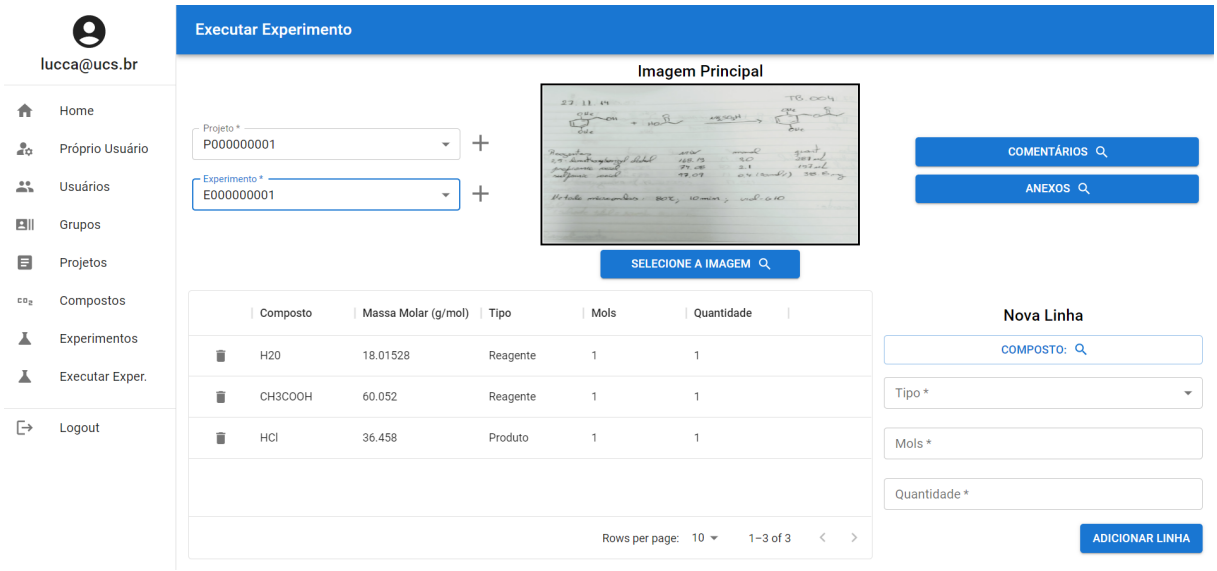


Fonte: O Autor (2022).

4.5.5.7 Caso de Uso UC-24 Executar um Experimento

O caso de uso 'UC-24 Executar um Experimento' atende ao requisito funcional RF-26. A Figura 62 apresenta a interface gráfica deste caso de uso.

Figura 62 – Interface Gráfica UC-24 Executar um Experimento



Fonte: O Autor (2022).

O Quadro 24 apresenta o detalhamento deste caso de uso.

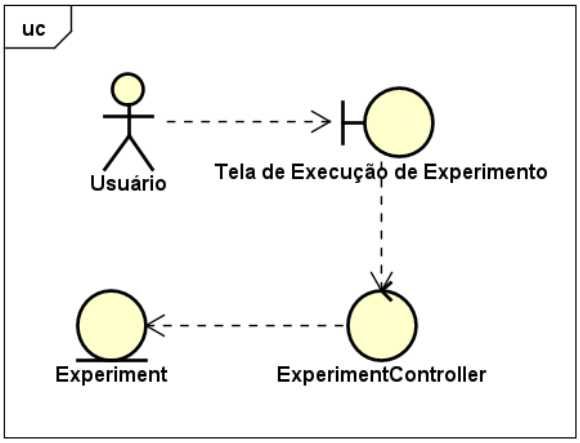
Quadro 24 – Detalhamento UC-24 Executar um Experimento

Código e Descrição	UC-24 Executar um Experimento
Ator	Usuário
Descrição	Permite que um usuário realize a execução de um experimento.
RF Associados	RF-26
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão ‘Executar um Experimento’.
Pós-Condições	Experimento executado.
Fluxo principal	1. Usuário seleciona a entrada ‘Executar Exper.’ no menu lateral. 2. Sistema exibe a tela ‘Executar Experimento’. 3. Sistema propriedades a serem definidadas : “Projeto”, “Experimento” e “Imagem Principal”. 4. Usuário define o projeto selecionando através de um combobox ou através do caso de uso UC-14 Gerenciar Projeto, clicando no ícone de adição. 5. Usuário define o experimento selecionando através de um combobox ou através do caso de uso UC-20 Gerenciar Experimento, clicando no ícone de adição. 6. Sistema exibe Imagem Principal, caso não seja nula. 7. Usuário define a imagem principal do experimento. 8. Sistema exibe botão “Anexos” acompanhado de um ícone de lupa. Permite a execução do caso de uso UC-22 Gerenciar Anexos de Experimento. 9. Sistema exibe botão “Comentários” acompanhado de um ícone de lupa. Permite a execução do caso de uso UC-22 Gerenciar Anexos de Experimento. 10. Sistema exibe tabela de linhas do experimento com as colunas "Composto", "Massa Molar (g/mol)", “Tipo”, “Mols“ e "Quantidade"e coluna com ícone de exclusão. 11. Para excluir uma linha: 11.1 Usuário clica no ícone de exclusão. 11.2 Sistema exclui linha de experimento. 12. Para editar propriedades de uma linha: 12.1. Usuário clica na célula ’Mols’ e/ou ’Quantidade’. 12.2. Sistema desbloqueia edição destas células. 12.3. Usuário define novos valores. 12.4. Usuário pressiona ’Enter’. 12.5. Sistema grava novos valores da linha. 13. Sistema exibe formulário denominado ’Nova Linha’ com o botão ’Composto’, ’Tipo’, ’Mols’ e ’Quantidade’. 13.1. Usuário define propriedades. 13.2. Usuário clica no botão ’Composto’. 13.3. Sistema exibe janela ’Compostos’ com tabela de compostos cadastrados. 13.4. Usuário seleciona o composto. 13.5. Usuário pressiona o botão ’Salvar’. 13.6. Sistema fecha janela ’Compostos’. 13.7. Usuário pressiona ’Adicionar Linha’. 13.7. Sistema valida dados. 13.8. Sistema grava nova linha de experimento. 13.9. Sistema atualiza tabela de linhas.
Fluxos alternativos	**Validação de Dados** 1.a) Dados obrigatórios. 1.1. Sistema informa que o ’Tipo’ é obrigatório. 1.2. Sistema informa que o ’Mols’ é obrigatório. 1.3. Sistema informa que o ’Quantidade’ é obrigatório. 1.4. Sistema informa que o ’Composto’ é obrigatório.

Fonte: O Autor (2022).

A Figura 63 apresenta o diagrama de robustez deste caso de uso.

Figura 63 – Diagrama de Robustez UC-24 Executar um Experimento

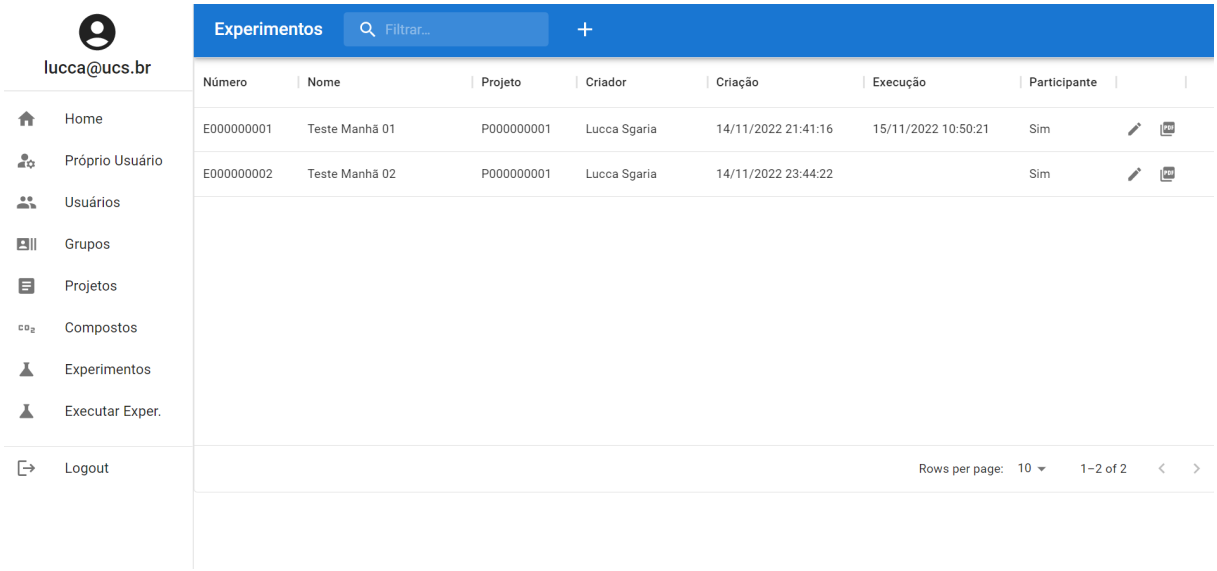


Fonte: O Autor (2022).

4.5.5.8 Caso de Uso UC-25 Exportar o conteúdo de um Experimento

O caso de uso 'UC-25 Exportar o conteúdo de um Experimento atende ao requisito funcional RF-27. A Figura 64 apresenta a interface gráfica deste caso de uso.

Figura 64 – Interface Gráfica UC-25 Exportar o conteúdo de um Experimento



Fonte: O Autor (2022).

O Quadro 25 apresenta o detalhamento deste caso de uso.

Quadro 25 – Detalhamento UC-25 Exportar o conteúdo de um Experimento

Código e Descrição	UC-25 Exportar o conteúdo de um Experimento
Ator	Usuário
Descrição	Permite que um usuário realize a exportação dos experimentos cadastrados no sistema que o mesmo possua permissão de acesso.
RF Associados	RF-27
Pré-Condições	1. O usuário deve estar autenticado no sistema. 2. O grupo do usuário deve possuir a permissão 'Listar Experimentos' e 'Exportar Experimento'. 3. Realização do caso de uso 'UC-18 Listar Experimentos'.
Pós-Condições	Conteúdo de um Experimento exportado
Fluxo principal	**A partir do UC-18 Listar Experimentos** 1. Usuário clica no ícone de Exportação. 2. Sistema cria arquivo PDF do conteúdo do experimento. 3. Sistema realiza download do arquivo PDF gerado.
Fluxos alternativos	

Fonte: O Autor (2022).

O caso de uso possui o mesmo diagrama de robustez apresentado na .Figura 51

5 CONSIDERAÇÕES FINAIS

Neste trabalho foi desenvolvido o protótipo de um *software* para utilização de pesquisadores do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos da UCS, a partir da necessidade de solucionar seus problemas descritos previamente.

A metodologia ICONIX serviu de base para a construção deste projeto. A utilização desta técnica caracterizou-se como adequada devido a sua natureza iterativa, fato esse que permitiu que adequações e refinamentos relacionados aos diagramas produzidos acontecessem durante o processo. Além disso, todos artefatos necessários para a etapa de implementação do *software* foram produzidos.

As escolhas das tecnologias que utilizadas na etapa de implementação deste *software* foram baseadas na detalhamento técnico das mesmas, sendo consideradas suas características, vantagens e em alguns casos a familiaridade do autor com a tecnologia em si.

Deste modo, o projeto proposto foi considerado como viável e serviu como base para a etapa de implementação do protótipo funcional. Este processo ocorreu de forma satisfatória no segundo semestre de 2022. Foi possível seguir a maior parte do mapeamento realizado com a metodologia ICONIX, sendo necessário apenas algumas atualizações e mudanças em relação ao diagrama de classes, protótipos de tela e casos de uso.

A implementação relacionada ao *back-end* do projeto, com a utilização do *framework* Quarkus, foi considerada positiva. Este fato ocorreu devido ao conhecimento prévio e experiência pessoal com a linguagem de programação Java, assim como relaciona-se com as características e funcionalidades do Quarkus, citadas neste trabalho.

Na parte do *front-end*, a utilização do React demandou um desafio relacionado a curva de aprendizagem, tanto da ferramenta em si quanto do conhecimento necessário para o desenvolvimento *front-end*. No entanto, a implementação desta etapa também ocorreu de forma satisfatória, sendo possível atingir todos os objetivos do projeto proposto.

Referente ao impacto do produto, conclui-se que a possível utilização do mesmo pelos pesquisadores proporciona a padronização dos processos realizados no laboratório, elimina a necessidade do armazenamento de cadernos físicos, facilita o compartilhamento de resultados e melhora o aspecto de reprodutibilidade dos experimentos químicos reproduzidos pelo sistema. Além disso, o *software* permite ao Prof. Dr. Thiago Barcellos da Silva, responsável pelo laboratório, uma visão geral e detalhada de todos os projetos sendo realizados no momento, assim como de todos já reproduzidos.

Por fim, considero que todo o processo do desenvolvimento deste Trabalho de Conclusão de Curso, desde a escrita da monografia, o mapeamento do projeto e a implementação

do protótipo, caracterizou-se como um importante desafio que representou um crescimento em minha vida pessoal, acadêmica e profissional. Mantenho a expectativa que o trabalho e o protótipo do *software* possam servir para uma futura implantação e utilização do mesmo pelos pesquisadores do Laboratório de Biotecnologia de Produtos Naturais e Sintéticos da UCS.

REFERÊNCIAS

BUENO, A. S.; PORTER, J. **Quarkus Cookbook Kubernetes-Optimized Java Solutions**. 1. ed. [S.l.]: O'Reilly Media, Inc., 2020. 374 p.

CLINGAN, J.; FINNIGAN, K. **Kubernetes Native Microservices with Quarkus and MicroProfile**. 1. ed. [S.l.]: Manning, 2022. 345 p.

CONNOLLY, R.; HOAR, R. **Fundamentals of Web Development, Global Edition**. 1. ed. [S.l.]: Pearson Education Limited, 2015. 1024 p.

FELKE-MORRIS, T. **Web development and design foundations with HTML5**. 9. ed. [S.l.]: Pearson, 2019. 720 p.

GONCALVES, A. **Understanding Quarkus**. [S.l.]: Independently Published, 2020. 408 p. (Agoncal Fascicles).

GRAALVM. **Introduction to GraalVM**. 2022. Disponível em: <https://www.graalvm.org/22.0/docs/introduction/>.

HIGGINS, S. G. *et al.* Considerations for implementing electronic laboratory notebooks in an academic research environment. **Nature Protocols**, Nature Publishing Group, v. 17, n. 2, p. 179–189, 2022.

KEMP, S. **Digital 2022 Global Overview Report**. 2022. Disponível em: <https://datareportal.com/reports/digital-2022-global-overview-report>.

KOLEOSO, T. **Beginning Quarkus Framework Build Cloud-Native Enterprise Java Applications and Microservices**. 1. ed. [S.l.]: Apress, 2020. 301 p.

MARTINOTTO, I. **Comparativo entre as ferramentas de desenvolvimento Web: Angular, Vuejs e React para construção de um aplicativo de E-commerce**. 111 p. Monografia (Trabalho de Conclusão de Curso) — Universidade de Caixas do Sul, Caxias do Sul, 2021.

QUARKUS. **What is Quarkus?** 2022. Disponível em: <https://quarkus.io/about/>.

ROSENBERG, D.; STEPHENS, M. **Use case driven object modeling with UML: theory and practice**. 1. ed. [S.l.]: Apress, 2007. 440 p.

ROSSATO, A.; SILVA, S. Ávila e. Development of prototype of electronic notes notebook for scientific chemistry laboratories. 2019.

WIEGERS, K. E.; BEATTY, J. **Software Requirements**. 3. ed. [S.l.]: Microsoft Press, 2013. 672 p.