



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
COLEGIADO DO CURSO DE ENGENHARIA DE COMPUTAÇÃO

Rafael Franco Ribeiro

**Evolução do C2D: Módulo de Credenciamento e
Recredenciamento de Docentes do Sistema
Marvin**

Vitória, ES

2022

Rafael Franco Ribeiro

Evolução do C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Monografia apresentada ao Curso de Engenharia de Computação do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Universidade Federal do Espírito Santo – UFES

Centro Tecnológico

Colegiado do Curso de Engenharia de Computação

Orientador: Prof. Vítor E. Silva Souza, Ph.D.

Vitória, ES

2022

Rafael Franco Ribeiro

Evolução do C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin/ Rafael Franco Ribeiro. – Vitória, ES, 2022-
88 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Vítor E. Silva Souza, Ph.D.

Monografia (PG) – Universidade Federal do Espírito Santo – UFES
Centro Tecnológico
Colegiado do Curso de Engenharia de Computação, 2022.

1. Palavra-chave1. 2. Palavra-chave2. I. Souza, Vítor Estêvão Silva. II. Universidade Federal do Espírito Santo. IV. Evolução do C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

CDU 02:141:005.7

Rafael Franco Ribeiro

Evolução do C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Monografia apresentada ao Curso de Engenharia de Computação do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Trabalho aprovado. Vitória, ES, (dia) de (mês) de (ano):

Prof. Vítor E. Silva Souza, Ph.D.
Orientador

Prof^a. Monalessa Perini Barcellos,
D.Sc.
Universidade Federal do Espírito Santo

Bruno Borlini Duarte, D.Sc.
Universidade Federal do Espírito Santo

Vitória, ES
2022

Agradecimentos

Em primeiro lugar gostaria de agradecer a Deus, por ter permitido que eu tivesse saúde e determinação para ultrapassar os obstáculos e desafios encontrados ao longo desse anos de formação acadêmica.

Aos meus pais e meu irmão Arlete, Clóvis, Rodolfo a minha eterna gratidão por todo o suporte que vocês me deram durante a graduação. Agradeço a eles por todo amor, carinho, dedicação e ensinamento que me tornaram a pessoa que sou hoje.

Muitas das dificuldades diárias do curso foram vencidas com a ajuda de amigas que fiz dentro da Universidade ao longo desse anos. Aos meus amigos, Bruno, Elson, Santana, Risso, companheiros de cursos, agradeço por toda troca de conhecimento que realizamos diversas vezes até tarde da noite e pelos momentos de descontração que tivemos para recuperarmos nossas forças.

Agradeço também a todos os professores que fizeram parte desta caminhada acadêmica, com seus ensinamentos e orientações consegui absorver diferentes conceitos técnicos necessários para que possa me tornar um engenheiro da computação. Em especial agradeço ao meu orientador Vítor, por me guiar na construção deste trabalho, tirando minhas dúvidas da melhor maneira possível.

Por último agradeço a minha namorada Myllena que me deu mais para concluir este curso. Mostrou para mim uma nova face da vida repleta de bondade, amor, carinho e compaixão. Espero do fundo do coração que possamos seguir construindo uma vida juntos.

Eu acredito que não conseguimos chegar a algum lugar sem ajuda das outras pessoas. Então como maior presente que essa jornada de formação acadêmica me deixou foi de ter encontrado pessoas que me ajudaram e ajudaram a avançar na vida.

Resumo

Neste trabalho visamos construir uma aplicação Web para auxiliar e agilizar o processo de avaliação de docentes de Pós-Graduação em Informática da Universidade Federal do Espírito Santo (PPGI/UFES) realizada pelos seus coordenadores e secretários. Essa avaliação ocorre anualmente para definir o credenciamento dos docentes, servindo para definir se ele pode lecionar, orientar estudantes, usar recursos financeiros para projetos, entre outros.

Para que o docente seja credenciado diversos critérios são analisados, porém neste trabalho trataremos apenas os critérios referentes aos trabalhos produzidos pelos docentes. Os veículos onde esses trabalhos são publicados recebem uma nota de classificação chamada de Qualis, assim os Programas de Pós-Graduação (PPGs) definem as pontuações que cada docente deve atingir através dos seus trabalhos produzidos dentro de um determinado período.

Propõe-se esta aplicação Web como um módulo do Marvin. O sistema Web Marvin foi criado para agregar ferramentas úteis para tarefas de ensino, pesquisa e processo administrativos com o intuito de auxiliar professores, alunos e servidores das universidades. Neste contexto, muitos estudantes da UFES desenvolvem módulos para serem incorporados ao Marvin como projeto de conclusão de curso.

Desta forma, a partir de um levantamento, análise e documentação de requisitos foi desenvolvido um novo módulo para o Marvin chamado de Classificação e Credenciamento de Docentes (C2D), aplicando o método FrameWeb na fase de projeto de sistema, e reutilizando algumas funcionalidades já desenvolvidas no Marvin na implementação. O objetivo principal deste módulo é informar se um docente pode ou não ser credenciado a um PPG, segundo as regras de classificação configuradas para o mesmo.

Palavras-chaves: PPG. Qualis. Marvin. C2D. FrameWeb.

Lista de ilustrações

Figura 1 – O processo da Engenharia de Requisitos (adaptado de (KOTONYA; SOMMERVILLE, 1998)).	17
Figura 2 – Representação da Arquitetura de Software MVC.	20
Figura 3 – Camadas do FrameWeb (SOUZA, 2020).	21
Figura 4 – Diagrama de casos de uso.	28
Figura 5 – Arquitetura de Software.	29
Figura 6 – Modelo de Entidades do C2D.	31
Figura 7 – Classe CrudServiceBean.	32
Figura 8 – Classes de Serviço do C2D.	33
Figura 9 – Classe BaseJPADAO.	34
Figura 10 – Objetos DAO do C2D.	34
Figura 11 – Modelo de navegação CRUD.	35
Figura 12 – Modelo de navegação do cálculo de pontuação dos docentes.	36
Figura 13 – Tela de Login.	37
Figura 14 – Tela de Home.	38
Figura 15 – Tela dos PPG.	39
Figura 16 – Tela de Administração de um PPG.	40
Figura 17 – Tela de Gerenciamento de Qualis.	41
Figura 18 – Tela de cadastro de Qualis.	41
Figura 19 – Tela de Gerenciamento de Venue.	42
Figura 20 – Tela de Importação de Venue.	43
Figura 21 – Tela de Gerenciamento de Regras.	44
Figura 22 – Tela de Cadastro de Regras.	44
Figura 23 – Tela de Importação de Currículo Lattes.	45
Figura 24 – Tela de Confirmação de Importação.	46
Figura 25 – Tela de Associação da Publicação com o Venue.	46
Figura 26 – Tela Indicação de Docente para Avaliação.	47
Figura 27 – Tela exibição da avaliação.	50

Lista de tabelas

Lista de abreviaturas e siglas

UML	Unified Modeling Language
PPG	Programa de Pós-Graduação
PPGI	Programa de Pós-Graduação em Informática
C2D	Classificação e Credenciamento de Docentes
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
UFES	Universidade Federal do Espírito Santo
DI	Departamento de Informática
MVC	Model View Controller
SI	Sistema de Informação
TAEs	Técnicos Administrativos
ABNT	Associação Brasileira de Normas e Técnicas
HTTP	Hypertext Transfer Protocol
JSP	Java Server Pages
JSF	Java Server Faces
JPA	Java Persistence API
EJB	Enterprise Java Beans
API	Application Programming Interface
HTML	Linguagem de Marcação de HiperTexto
XML	eXtensible Markup Language
XHTML	Linguagem de Marcação para Hipertexto Extensível
PDF	Portable Document Format
WIS	Web Information System
DAO	Data Access Object

CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
PET	Programa de Educação Tutorial
CSV	Comma Separated Values
ISSN	International Standard Serial Number

Sumário

1	INTRODUÇÃO	12
1.1	Motivação e Justificativa	12
1.2	Objetivos	13
1.3	Método de Desenvolvimento do Trabalho	13
1.4	Organização da Monografia	14
2	REFERENCIAL TEÓRICO E TECNOLOGIAS UTILIZADAS	15
2.1	Engenharia de Software	15
2.1.1	Engenharia de Requisitos	16
2.1.2	Projeto e Implementação	17
2.2	Desenvolvimento Web	18
2.3	O método FrameWeb	21
3	ESPECIFICAÇÃO DE REQUISITOS	23
3.1	Descrição do Minimundo	23
3.1.1	Cálculo da pontuação	24
3.1.2	Ingresso no programa	24
3.1.3	Regras para o Recredenciamento	25
3.2	Fluxo de trabalho do C2D	26
3.3	Diagrama de Caso de Uso	27
4	ARQUITETURA E MODELAGEM DO PROJETO	29
4.1	Arquitetura do sistema	29
4.2	Modelagem do sistema	30
4.2.1	Camada de Negócio	31
4.2.2	Camada de Acesso a Dados	33
4.2.3	Camada de Apresentação	35
5	APRESENTAÇÃO	37
5.1	Login	37
5.2	Manage PPGs	39
5.3	Manage Qualis	40
5.4	Manage Venue	42
5.5	Manage Rules	43
5.6	Upload lattes CV	45
5.7	Calculate Score	47

6	CONCLUSÃO	51
6.1	Considerações Finais	51
6.2	Trabalhos Futuros	52
	REFERÊNCIAS	53
	APÊNDICES	55

1 Introdução

Todo início de ano, os docentes do Programa de Pós-Graduação em Informática da Universidade Federal do Espírito Santo (PPGI/UFES) são avaliados pelas suas publicações de artigos científicos e tecnológicos.¹ Os trabalhos produzidos são qualificados de acordo com os veículos da publicação, por meio de um índice chamado Qualis,² definido pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).³ Os docentes que fazem parte do PPGI devem ser credenciados para possam lecionar disciplinas, orientar estudantes, usar recursos financeiros do PPGI, etc.

Este cenário e outros processos que podem ser otimizados dentro da universidade estão dentro do escopo de um projeto de desenvolvimento de um sistema Web chamado Marvin.⁴ Tal sistema engloba vários projetos de graduação desenvolvidos pelos estudantes do Departamento de Informática (DI), que criam ferramentas úteis para as atividades de ensino, pesquisa e extensão da UFES, funcionando como um mecanismo de integração entre essas diversas ferramentas.

Pontua-se que o processo de credenciamento e de classificação dos docentes dos programas de pós-graduação pode ser incluído como uma funcionalidade do Marvin, visando facilitar e otimizar a atividade de classificação e gerência de docentes do PPGI/UFES e de outros programas que utilizem critérios similares. Nesse sentido, propõe-se o desenvolvimento de um módulo Classificação e Credenciamento de Docentes (C2D), obedecendo os critérios estabelecidos pelo PPGI/UFES para que ele possa administrar a entrada e saída no programa para os docentes.

O desenvolvimento desse módulo já proposto por outra monografia de projeto de graduação (SILVA, 2017), porém os requisitos referentes à classificação foram alterados, de tal forma que o sistema construído não atende o necessário para realizar a classificação. Portanto, este trabalho é uma reconstrução do módulo C2D adequado às novas regras de classificação.

1.1 Motivação e Justificativa

A escolha da temática foi motivada pela oportunidade de contribuir para o Marvin, que é um projeto Web que tem relevância para a UFES e que segue as boas práticas do desenvolvimento de software. Na UFES existem muitos processos que ainda são feitos

¹ <<https://informatica.ufes.br/pt-br/pos-graduacao/PPGI/credenciamento-de-docentes>>

² <<https://www.gov.br/capes/pt-br/assuntos/noticias/capes-melhora-ferramentas-de-avaliacao-da-pos-graduacao>>

³ <<https://www.gov.br/capes/pt-br>>

⁴ <<https://github.com/dwws-ufes/marvin/>>, <<https://projetos.ufes.br/#/projetos/523/informacoes>>

manualmente por técnicos administrativos (TAEs) da universidade, fazendo com que ocorra uma série de consequências negativas, como por exemplo:

- Retrabalho, em função dos muitos erros cometidos durante as atividade para concluir o processo;
- Falta de agilidade nos processos não automatizados;
- Perda da mão de obra, para poder realizar o processo.

A criação do módulo C2D visa eliminar ou atenuar tais consequências negativas que o processo de credenciamento e gerência de docentes do PPGs gera para a Universidade. Deve ser ressaltado que os critérios de avaliação que a CAPES e/ou o PPGI/UFES utilizam podem ser alterados no futuro, por isso o módulo deve permitir a configuração de novos critérios para não perder a sua utilidade para a área acadêmica.

1.2 Objetivos

O objetivo geral desse trabalho é a construção de um sistema Web para classificar e credenciar docentes segundo os critérios estabelecidos pelo PPGI/UFES. Desse modo, para realizar o objetivo geral serão realizados os seguintes objetivos específicos:

- Levantamento, análise e documentação dos requisitos do software;
- Especificação e documentação do projeto arquitetural do software, incluindo experimentação do método de desenvolvimento de sistemas de informação Web FrameWeb (SOUZA, 2020);
- Implementação de um protótipo do sistema na forma de um módulo a ser integrado ao Marvin.

1.3 Método de Desenvolvimento do Trabalho

O sistema foi desenvolvido utilizando *frameworks* e ferramentas como a plataforma Java EE, Git, componentes de interface PrimeFaces e banco de dados MySQL, as mesmas tecnologias já utilizadas no Marvin, sistema ao qual este novo módulo se integra. Para o desenvolvimento desse trabalho foram aplicados os conceitos adquiridos nas disciplinas de Engenharia de Software, Programação, Linguagens de Programação e Banco de Dados. Destacam-se as seguintes atividades:

- Estudo: adotamos a pesquisa bibliográfica na qual analisamos os fundamentos teóricos

e conceituais das disciplinas de Programação Orientada a Objetos, Engenharia de Software e Banco de Dados, para embasar suas boas práticas no desenvolvimento do módulo C2D. Além das pesquisas bibliográficas também foram estudadas as documentações das tecnologias aplicadas para construção do módulo;

- Engenharia Requisitos e Projeto de Sistema de Software: produzimos inicialmente o Documento de Especificação de Requisitos, no qual consta a descrição do minimundo do sistema, os requisitos funcionais e não funcionais, bem como as regras de negócio, além da modelagem de casos de uso, modelo estrutural e glossário do projeto. Posteriormente, e seguindo as diretrizes propostas no método FrameWeb (SOUZA, 2020), construímos o Documento de Projeto, no qual é detalhada a arquitetura do software e cada camada é modelada em mais detalhes;
- Implementação do sistema: utilizamos tecnologias voltadas ao desenvolvimento Web, tais como: a linguagem de programação Java, a plataforma Java EE, a IDE Eclipse, Git, a biblioteca de componentes de interface com usuário PrimeFaces e banco de dados MySQL para implementação do sistema;
- Verificação e Validação do sistema: implementadas as funcionalidade do sistema, com a realização de testes manuais foi feita a validação e verificação do seu funcionamento;
- Redação da Monografia: feita com o auxílio do \LaTeX , utilizando o editor Texstudio e o *template* abnTeX, o qual foi construído para atender as normas da Associação Brasileira de Normas e Técnicas (ABNT), que padroniza a redação de documento técnicos e científicos brasileiros.

1.4 Organização da Monografia

Além desta introdução, esta monografia é composta por outros cinco capítulos:

- O Capítulo 2 apresenta os aspectos relativos ao conteúdo teórico relevante para o trabalho;
- O Capítulo 3 apresenta os requisito levantados para esse trabalho;
- O Capítulo 4 apresenta a arquitetura e a modelagem do sistema;
- O Capítulo 5 apresenta as principais tela do sistema.
- O Capítulo 6 apresenta as considerações finais do trabalho.

2 Referencial Teórico e Tecnologias Utilizadas

Este capítulo resume os conceitos teóricos e fundamentais para o desenvolvimento do módulo C2D. A Seção 2.1 aborda a Engenharia de Software, demonstrando os principais processos e conceitos utilizados. A Seção 2.2 apresenta os principais conceitos de desenvolvimento de sistemas na plataforma Web. A Seção 2.3 destaca o método FrameWeb.

2.1 Engenharia de Software

A sociedade contemporânea vive uma intensa substituição dos processos manuais pelos produtos que a tecnologia oferece hoje. Constata-se uma maior procura por software para a resolução das mais diversas demandas das mais diversas áreas de conhecimento humano, devido a isso observa-se um aumento nas atividades de desenvolvimento de software (BARCELLOS, 2018).

Inicialmente, a primeira habilidade da área de desenvolvimento de software que deve ser adquirida com objetivo de trabalhar a resolução de problemas é o raciocínio lógico, o qual é desenvolvido por meio de aplicações dos conceitos de programação e estrutura de dados em projetos práticos. No entanto, à medida que a complexidade desses problemas vão aumentando gradativamente, apenas o raciocínio lógico não será capaz de resolvê-los sem perder a qualidade do software. A qualidade de um software deve ser avaliada quando analisamos: por diferentes perspectivas (usuário, desenvolvedor e cliente); diferentes características (usabilidade, confiabilidade, eficiência, manutenibilidade, portabilidade, segurança e produtividade), que devem ser alcançadas dependendo do propósito do software (BARCELLOS, 2018).

Nesse sentido, a Engenharia de Software trabalha para estabelecer processos, métodos, técnicas, ferramentas e ambiente visando: organização, produtividade e qualidade no desenvolvimento do software. Com isso destacam-se algumas atividades que são realizadas com esses propósitos:

- Levantamento e análise de requisitos;
- Projeto de arquitetura;
- Implementação.

Além disso, ressalta-se que o conteúdo de outras disciplinas, como por exemplo a Gerência de Projeto, pode auxiliar no desenvolvimento do software nos processos de

organização de recursos humanos, na definição de cronograma do projeto, na análise de custos e riscos, etc.

Para apresentar de maneira mais clara e destacar a importância da Engenharia de Software para o desenvolvimento deste trabalho, as seções a seguir abordam com mais detalhes as atividades de Engenharia de Requisitos (Seção 2.1.1) e projeto e implementação (Seção 2.1.2).

2.1.1 Engenharia de Requisitos

A Engenharia de Requisitos têm um papel central no desenvolvimento de software, sendo ela considerada uma das bases para a definição das funcionalidades de um sistema. O levantamento e a análise dos requisitos tornam-se determinantes para o sucesso ou fracasso de um projeto de software. Sendo assim, as principais atividades da Engenharia de Requisitos são os processos de levantar, analisar, documentar, gerenciar e controlar a qualidade dos requisitos (BARCELLOS, 2018).

Existem diversas definições de requisitos que podem ser encontradas na literatura, como por exemplo:

- Requisitos são descrições dos serviços que devem ser providos pelo sistema e de suas restrições operacionais (KOTONYA; SOMMERVILLE, 1998);
- Um requisito é uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir seus objetivos (PFLEEGER, 2004);
- Um requisito é alguma coisa que o produto tem de fazer ou uma qualidade que ele precisa apresentar (ROBERTSON; ROBERTSON, 1998).

Portanto, com essas definições, conclui-se que um requisito de sistema deve ter as seguintes informações:

- Especificação das funcionalidades que um sistema deve ter;
- Restrições em que ele deve operar;
- Propriedades gerais do sistema;
- Restrições que devem ser obedecidas durante o processo de desenvolvimento.

Com base nas definições acima, os requisitos são comumente divididos em 3 tipos:

- **Requisitos Funcionais:** são as descrições de serviços que os sistemas fornecem;

- **Requisitos Não Funcionais:** especificações de um determinado comportamento do sistema, como usabilidade, segurança, disponibilidade, compatibilidade, dentre outros;
- **Regras do negócio:** são premissas e restrições que sofre uma operação comercial de uma empresa, que precisa ser atendida para que o negócio funcione de maneira adequada.

Nesta etapa não estamos preocupados com a solução técnica do problema e sim com o domínio do problema. Deste modo, a Engenharia de Requisitos envolve as atividades ilustradas na Figura 1. A primeira etapa do processo, o levantamento de requisitos, leva em consideração as necessidades dos usuários e clientes. Depois da identificação dos requisitos inicia-se o processo de análise de requisitos, que serão usados como base para a modelagem do sistema. Os processos devem ser documentados no Documento de Especificação de Requisitos, além de serem verificados e validados.

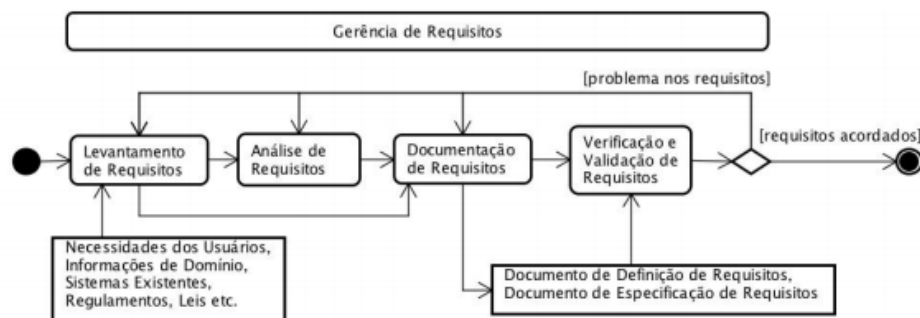


Figura 1 – O processo da Engenharia de Requisitos (adaptado de (KOTONYA; SOMMERVILLE, 1998)).

Assim, caso cliente, usuário e desenvolvedor estejam de acordo com os requisitos e a modelagem conceitual, utilizada para um melhor entendimento e visualização do domínio do problema, esteja terminada, podemos então pensar numa solução técnica para iniciar a implementação da aplicação, caso contrário deve-se retornar ao processo onde o problema foi identificado.

2.1.2 Projeto e Implementação

Após termos feito todos os processos da etapa anterior podemos iniciar a etapa de projeto, que tem como objetivo definir o software a ser implementado bem como a solução técnica que será utilizada. O projeto de software corresponde à primeira atividade que leva em conta os aspectos tecnológicos.

Produz-se uma solução para o problema identificado e modelado, incorporando as tecnologias aos requisitos essenciais do usuário e projetando o que será construído na

implementação. Com objetivo de representar o que será codificado na fase de implementação, o projeto deverá produzir os seguintes modelos de projetos gerados a partir da análise do projeto:

- **Projeto da Arquitetura do Software:** visa a definição dos componentes estruturais do software e seus relacionamentos;
- **Projeto de Dados:** tem como finalidade projetar a estrutura que será responsável por armazenar os dados necessários para implementar o projeto;
- **Projeto de Interfaces:** escreve como o software deverá se comunicar dentro dele mesmo (interfaces internas), com outros sistemas (interface externas) e com pessoas que o utilizam (interface com o usuário);
- **Projeto Detalhado:** tem como finalidade refinar e detalhar a descrição dos componentes estruturais do software.

Portanto, para que possamos começar a implementar o projeto C2D, espera-se que a arquitetura do sistema esteja bem definida. Para isso, temos que ter seus projetos de componentes, divididos de maneira que as funcionalidades diferentes pertençam a camadas diferentes. Existem diversas estruturas de projeto de software quando entramos no contexto de aplicações Web, uma das estruturas mais utilizadas é a que separa o sistema de informação (SI) em três camadas: a camada de apresentação, a camada de domínio e a camada de persistência (FOWLER, 2002).

A camada de **apresentação** é aquela que está em contato direto com o usuário e lidando com as interações dele, como os cliques no mouse e as informações que ele digita/insere dentro do sistema, além de lidar, na plataforma Web, com requisições HTTP e exibição de informações da aplicação. O trabalho de implementar as regras de negócio e requisitos do domínio do sistema é destinado à camada de **domínio**. Por último, a camada de **persistência** lida com o relacionamento do banco de dados com a aplicação, garantindo a persistência ou o armazenamento de dados e a recuperação deles quando for necessário.

2.2 Desenvolvimento Web

Nas últimas décadas o acesso à Internet se tornou cada vez mais fácil. Hoje estima-se que 4,66 bilhões de pessoas tenham acesso à Internet, com isso, o número de aplicações na Web vem aumentando a cada dia (ISTO, 2021). Anteriormente, essas aplicações só podiam ser utilizadas em computadores, agora estão presentes nos *tablets*, nos *smartphones* e em outros dispositivos. Devido a isso, é essencial que características como manutenibilidade, escalabilidade e compatibilidade sejam pensadas no processo de desenvolvimento de aplicações Web.

Os softwares desenvolvidos para a Web estão em constante evolução, tentando manter dados mais atualizados e concisos, com uma alta demanda de novos requisitos, quando comparados ao modelo mais popular de aplicações, antigamente aplicações de *desktop*. Com isso, a manutenibilidade e escalabilidade da aplicação devem ser planejadas de modo que a manutenção e novas funcionalidade possam ser incorporadas ao sistema, mas que não prejudique a experiência do usuário com o sistema, mantendo assim um software de qualidade.

Com a crescente demanda por aplicações Web, surgiu a Engenharia Web como a finalidade de promover ativamente abordagens sistemáticas, disciplinadas e quantificáveis, para o sucesso do desenvolvimento de aplicações e sistemas baseados na Web. Tornando mais fácil lidar com as complexidades e adversidades deste tipo de aplicações.

Para o desenvolvimento dessa aplicação utilizaremos o Java Platform Enterprise Edition (Java EE), que é uma plataforma ou ambiente projetada para o desenvolvimento de aplicações de grande porte e aplicações Web, que possui bibliotecas e funcionalidades que implementam software baseado na linguagem de programação Java. Essa plataforma oferece ao cliente e ao desenvolvedor recursos de criação de sistema, a possibilidade de aplicar os conceitos de segurança, escalabilidade, integridade, entre outros requisitos para o desenvolvimento Web (FARIA, 2015).

A plataforma Java EE possui uma série de tecnologias com objetivos distintos, sendo alguns dos mais conhecidos como:

- **Servlets:** são componentes Java executados no servidor que tem o objetivo de gerar conteúdo (HTML e XML) dinâmico para a Web;
- **Java Server Pages (JSP):** especialização de servlets, permite que as aplicações em Java sejam mais robustas e tenham facilidades no seu desenvolvimento;
- **Java Server Faces (JSF):** é um *framework* construído com o padrão MVC (*Model-View-Controller*) baseado em Java que ajuda a simplificar o desenvolvimento de interfaces (telas do sistema) através de um modelo de *User Interface* (UI);
- **Java Persistence API (JPA):** é uma API padrão do Java que usa o conceito de mapeamento objeto/relacional, sendo utilizada para a persistência dos dados;
- **Enterprise Java Beans (EJB):** são componentes que executam em um *container* de aplicação e que oferecem a facilidade e produtividade no desenvolvimento de componentes distribuídos, transacionados, seguros e portáteis (DEVMEDIA, 2021).

Além dessas tecnologias, também utilizaremos a biblioteca de componentes PrimeFaces, pois ela possui componentes Web responsivos (tabelas, campo de formulários, botões, dentre outros), tornando o módulo C2D mais agradável aos olhos dos usuários.

Com o aumento diário de usuários da Internet, o número de aplicações Web cresce em um ritmo acelerado, fazendo surgir a necessidade de criar uma forma que agilize o desenvolvimento sem comprometer o produto final. Assim, foram criados uma série de padrões de arquitetura de software com propriedades desejáveis como:

- Fraco acoplamento: tendo como princípio deixar suas camadas mais isoladas possível para que não interfiram umas com as outras;
- Alta coesão: tendo como princípio deixar as classe com responsabilidade única e realizá-la de maneira satisfatória;
- Descentralização do desenvolvimento: propõe-se dividir em vários componentes para que os desenvolvedores possam trabalhar em componentes diferentes, sem que atrapalhem o desenvolvimento de outra pessoa, tornando possível o trabalho em paralelo.

Um dos padrões mais utilizados é o *Model-View-Controller* (MVC) (DUARTE, 2011), que é adotado pelos *frameworks* utilizados na criação do módulo C2D. A característica principal desse padrão é a divisão em três camadas conforme apresentado na Figura 2. Cada camada tem sua funcionalidade definida:

- **View:** é a camada responsável pela interação do o usuário como o sistema, exibindo dados por meio de arquivos HTML, XML, XHTML, PDF, entre outros;
- **Model:** camada que contém a lógica da aplicação, responsável pelas regras de negócio e pela comunicação da aplicação com o banco de dados;
- **Controller:** camada intermediadora do sistema, comunicando-se com as outras duas camadas. As requisições provenientes do *browser* são processadas pelo *controller*, o qual acessa as informações dos *models* e as retorna para as *views*, exibindo de forma adequada ao usuário.

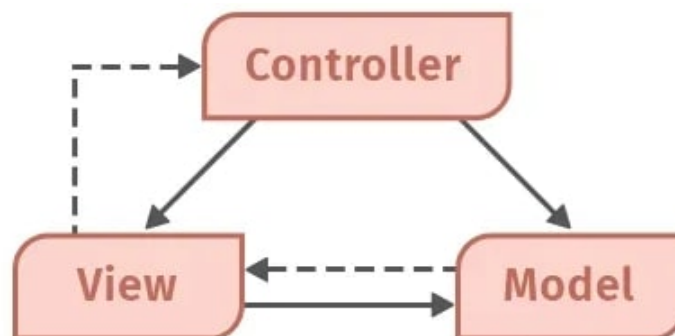


Figura 2 – Representação da Arquitetura de Software MVC.

2.3 O método FrameWeb

Como método de projeto para desenvolvimento de sistemas de informação Web, ou *Web Information Systems* (WIS's), utilizaremos o FrameWeb que é baseado em *frameworks*. Este método define uma arquitetura básica para o WIS e os *frameworks* que serão utilizados na construção do sistema, além de sugerir modelos de projeto com uma estrutura parecida com a dos sistemas que utilizam os *frameworks* (SOUZA, 2020). Tal arquitetura é representada na Figura 3.

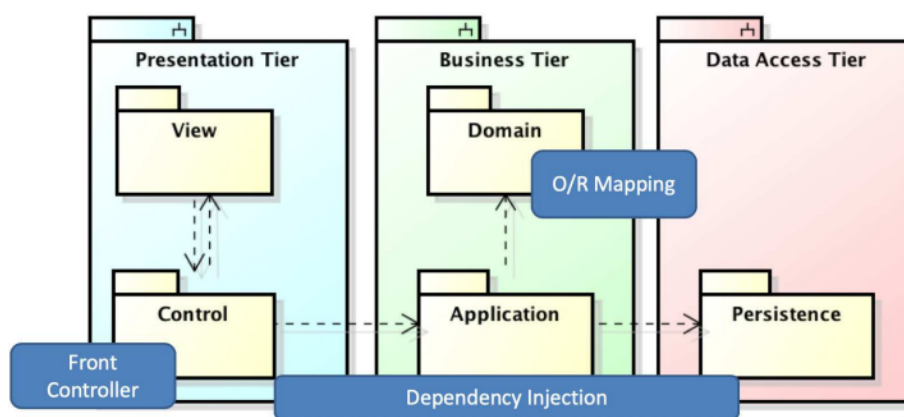


Figura 3 – Camadas do FrameWeb (SOUZA, 2020).

O FrameWeb estabelece uma arquitetura padrão para WIS's, que tem como base o padrão arquitetônico *Service Layer* (Camada de Serviço) (FOWLER, 2002), para que seja possível integrar com os *frameworks* escolhidos. O sistema é dividido nas seguintes camadas (SALVATORE, 2016):

- **Camada de Negócio (*Business Tier*):** atua diretamente com as regras de negócio, provendo as funcionalidades para que as mesmas funcionem dentro do sistema. Esta camada é dividida em duas: Lógica de Domínio (*Domain*) e Lógica de Aplicação (*Application*);
- **Camada de Apresentação (*Presentation Tier*):** lida com as funcionalidades necessárias para criação de interface para que o usuário consiga interagir com o sistema. Esta camada é dividida em duas: Visão (*View*) e Controle (*Control*);
- **Camada de Acesso a Dados (*Data Access Tier*):** trata das funcionalidades relacionadas à persistência de dados.

Para deixar os diagramas mais próximos da implementação, o FrameWeb propõe um conjunto de modelos de projeto que trazem conceitos utilizados pelos *frameworks* que

serão utilizados, por meio da definição de uma linguagem específica de domínio (MARTINS, 2016; SOUZA, 2020).

O metamodelo da UML é estendido pelo FrameWeb com o intuito de poder representar componentes físicos da plataforma Web e dos *frameworks* utilizados, especificando uma sintaxe própria. Assim é possível criar quatro tipos de diagramas:

- **Modelo de Entidades** (*Entity Model*): é um diagrama de classes da UML que representa os objetos do domínio do problema e seu mapeamento para a persistência de dados;
- **Modelo de Persistência** (*Persistence Model*): é um diagrama de classes da UML que representa as classes de acesso aos dados (DAO) existentes, responsáveis pela persistência das instâncias de classes de domínio;
- **Modelo de Navegação** (*Navigation Model*): é um diagrama de classes da UML contendo os diferentes componentes que formam a camada de Lógica de Apresentação, como páginas Web, Formulários HTML e *controllers*;
- **Modelo de Aplicação** (*Application Model*): é um diagrama de classes da UML que representa as classes de serviço, que lidam com a codificação de casos de uso, e suas dependências.

3 Especificação de Requisitos

Neste capítulo será mostrado o resultado da análise e levantamento de requisitos do C2D. Na Seção 3.1 o mimimundo do problema a ser resolvido é apresentado; na Seção 3.2 é descrito o fluxo de trabalho esperado para o sistema; na Seção 3.3 é apresentado o diagrama de casos de uso do projeto. No apêndice se encontra o Documento de Especificação de Requisitos, que possui mais informações sobre os requisitos desse projeto.

3.1 Descrição do Minimundo

O PPGI possui a função de classificar e credenciar seu docentes conforme suas produções de trabalhos científicos e tecnológicos. Tais produções são avaliadas a partir dos veículos nas quais foram publicadas, segundo os critérios definidos pela CAPES. Assim, os(as) docentes que estão credenciados podem lecionar disciplinas, orientar estudantes, usar recursos financeiros do PPGI, etc.

O PPGI definiu 6 normas referentes ao Credenciamento e Recredenciamento de Docentes,¹ a saber:

- (a) ingresso no programa;
- (b) recredenciamento no programa;
- (c) credenciamento como orientador no doutorado;
- (d) requisitos mínimos de publicação;
- (e) critérios de (re)credenciamento para membros em licença maternidade;
- (f) categorização de docentes;

Essas normas possuem vários critérios para suas análises como: orientação de alunos, vínculo do docentes com a instituição de ensino, publicação de trabalhos produzidos, atividade administrativas que desempenha, dentre outros.

Os requisitos do módulo C2D serão apenas aqueles com relação direta com a avaliação de publicações, por serem mais trabalhosos de se lidar manualmente e passíveis de automação por um sistema de informação. Portanto o sistema atenderá apenas os requisitos das normas (a), (b), (c) e (d) que façam avaliação das publicações dos(as) docentes.

¹ <<https://informatica.ufes.br/pt-br/pos-graduacao/PPGI/credenciamento-de-docentes>>

3.1.1 Cálculo da pontuação

A norma (d) define as regras de pontuações mínimas de publicação para que o(a) docente possa ingressar ou ser recredenciado no PPGI. Atualmente as regras definidas são:

1. As pontuações são concedidas de acordo com os estratos do Qualis, definido pela CAPES no documento de área da Computação, conforme esquema de pontos que estiver vigente em um dado momento. Por exemplo, atualmente temos: A1 = 1; A2 = 0,85; B1 = 0,7; B2 = 0,5; B3 = 0,2; B4 = 0,1; B5 = 0,05. Os níveis A1, A2 e B1 são conhecidos como “estrato restrito”;²
2. Em caso de co-autoria, todos os(as) docentes pontuam o valor cheio associado ao estrato Qualis do veículo da publicação, ou seja, não há divisão de pontos;
3. Seja o(a) docente um orientador de Mestrado, ele deve pontuar, no biênio, um total de 1,6 pontos, sendo 0,4 em periódico (qualquer estrato) e 0,7 no estrato A1-B1 (periódico ou conferência);
4. Seja o(a) docente um orientador de Doutorado, ele deve pontuar, no quadriênio, 3,2 pontos, sendo 0,7 em periódico do estrato A1-B1 e 1,4 no estrato A1-B1 (periódico ou conferência).

3.1.2 Ingresso no programa

Existem dois critérios para o ingresso do(a) docente no programa, são eles:

1. Ter plano de trabalho aprovado pelo colegiado, considerando alinhamento com as linhas de pesquisa e estrutura curricular do programa;
2. Atender aos requisitos de recredenciamento do programa (definidos na Seção 3.1.3), considerando-se os trabalhos publicados no período (biênio/quadriênio) imediatamente anterior à solicitação, porém utilizando os requisitos mínimos de publicação do ano seguinte à solicitação.

Para o ingresso de docentes no módulo C2D, será considerado que o primeiro critério já tenha sido atendido, ou seja o colegiado já aprovou o plano de trabalho do(a) docente.

² Recentemente a CAPES modificou o Qualis, que atualmente é composto de 8 níveis de classificação (A1–A4, B1–B4, sendo A1–A4 estrato restrito), além do nível C (não classificado). No entanto, para este documento nos pautamos nos níveis indicados nas normas do PPGI, conforme divulgado em seu

3.1.3 Regras para o Recredenciamento

No começo do ano os membros dos PPGI são avaliados para que possam ser recredenciados. Então o C2D fará essa validação seguindo os seguintes critérios estabelecidos pelo PPGI, para que possa determinar se o(a) docente deve ser recredenciado ou descredenciado. Os critérios são:

1. Produção: o(a) docente deve atender aos requisitos mínimos de publicação para permanência no programa considerando-se os trabalhos publicados no período anterior; OU ser bolsista de produtividade do CNPq; OU ser o atual coordenador do PPGI ou ter sido coordenador do PPGI por pelo menos 6 meses no ano anterior;
2. Orientação: o(a) docente deve ter tido ao menos 4 alunos orientados naquele ano, do contrário ter ofertado ao menos 2 vagas no ano;
3. Docência: professores permanentes devem ofertar ao menos 1 disciplina por ano no PPGI, colaboradores 1 a cada 2 anos, ficando isentos(as) docentes que tiveram qualquer tipo de licença ou afastamento de longo prazo ao longo do ano, como pós-doc e capacitação;
4. Administrativo: o(a) docente deve participar do Colegiado do PPGI ou de alguma de suas comissões, ficando isentos(as) docentes que são chefes de departamento, coordenadores de curso ou de PET, além dos 1/3 docentes mais produtivos do último quadriênio.

Se o(a) docente não conseguir atender a nenhum desses requisitos e não possuir orientações ativas ele será descredenciado do programa. Caso ainda possua alunos orientados, as seguintes regras serão aplicadas:

1. O professor não será descredenciado;
2. O professor será impedido de oferecer disciplinas até regularizar seu recredenciamento;
3. O professor será impedido de assumir novas orientações até regularizar seu recredenciamento;
4. Para cada aluno que possivelmente não concluirá o curso no ano corrente, o professor deve estabelecer co-orientação com outro professor do programa;
5. O professor não terá suas publicações registradas na Plataforma Sucupira a menos que consiga regularizar seu recredenciamento;

6. Para regularizar seu credenciamento, o professor deverá comprovar, em qualquer momento do ano, que atingiu os critérios de credenciamento do ano seguinte;
7. No caso de não atingir os critérios de credenciamento até o final do ano, o professor será descredenciado no ano seguinte e deverá passar as orientações dos alunos que ainda não concluíram ao co-orientador escolhido, conforme item 4, acima.

Novamente, é importante lembrar que o C2D atua apenas na automação do cálculo de pontuação da produção para atendimento aos requisitos de publicação, sendo as demais regras, também apresentadas acima, fora do escopo do sistema.

3.2 Fluxo de trabalho do C2D

O módulo C2D deverá funcionar de maneira mais automatizada possível, para isso ele deve registrar quem são os membros do PPG a partir dos cadastros básicos do Marvin, bem como inserir em lote as publicações dos(as) docentes e a avaliação Qualis de cada veículo a partir de arquivos extraídos de outros sistemas. Além disso, deve permitir a atualização de seus dados sempre que for necessário. Desse modo o C2D trabalhará da seguinte forma:

1. Em primeiro lugar, é preciso que exista o cadastro do programa de pós-graduação (PPG), com indicação do(a) coordenador(a), subcoordenador(a) e secretário(a), usuários pré-cadastrados no Marvin (módulo núcleo) que terão acesso às funcionalidades do C2D em relação ao PPG em questão (para simplificar, na sequência faremos referência apenas ao/à secretário(a));
2. O(A) secretário(a) deve indicar as pontuações, conforme normas do programa. Para isso ele(a) deve cadastrar os diferentes níveis de classificação definidos no Qualis (A1, A2, B1, B2, etc.) e então informar, para cada nível, um nome e sua pontuação para cada tipo de veículo (periódico e conferência). Deve-se indicar, também, quais níveis fazem parte do estrato restrito;
3. O C2D precisa saber como classificar cada publicação de acordo com o Qualis. Dessa forma, o secretário(a) envia arquivos em formatos padronizados ao sistema para o C2D extrair as informações de classificação Qualis dos veículos de publicação (conferências e periódicos);
4. Para que possa verificar os requisitos mínimos de publicação, o(a) secretário(a) deve primeiramente cadastrar os(as) docentes do programa a partir dos usuários já cadastrados no Marvin, indicando o tipo (atualmente orientadores de mestrado e doutorado);

5. Após isso ele(a) poderá inserir as regras de pontuação no sistema, informando nome, a que tipo de docentes ela se refere, uma data de início e fim da vigência, o período em que as publicações serão avaliadas (em número de anos passados) e a quantidade de pontos que deve ser alcançada em quatro categorias: pontuação total, pontuação em periódicos, pontuação no estrato restrito, pontuação em periódicos no estrato restrito;
6. O(A) secretário(a) deve então recuperar as publicações dos(as) docentes a partir de seus Currículos Lattes. O C2D tentará associar cada publicação extraída aos veículos cadastrados anteriormente;
7. Por fim, o(a) secretário(a) pode então solicitar o cálculo da pontuação dos(as) docentes para o C2D para que ele informe se o(a) docente atendeu as requisitos de publicação.

As pontuações dos níveis Qualis, a classificação dos veículos nos níveis e as regras de requisitos mínimos de publicação devem estar associadas a datas de início e fim de vigência, mantendo, assim, um histórico das configurações anteriores, bem como a configuração vigente do programa.

3.3 Diagrama de Caso de Uso

A partir dos requisitos descritos na Seção 3.2, foi feito um diagrama com todos os casos de uso necessários para o sistema funcionar. O diagrama se encontra na Figura 4. O Documento de Especificação de Requisitos, incluído nos apêndices desta monografia, foram descritos de forma detalhada os casos de uso e as user stories.

O diagrama traz cada um dos passos descritos na Seção 3.2 como um caso de uso, sendo o cadastro dos PPGs feito pelo administrador e os demais passos conduzidos pelo(a) secretário(a) do PPG (podendo ser substituído por seu/sua coordenador(a)).

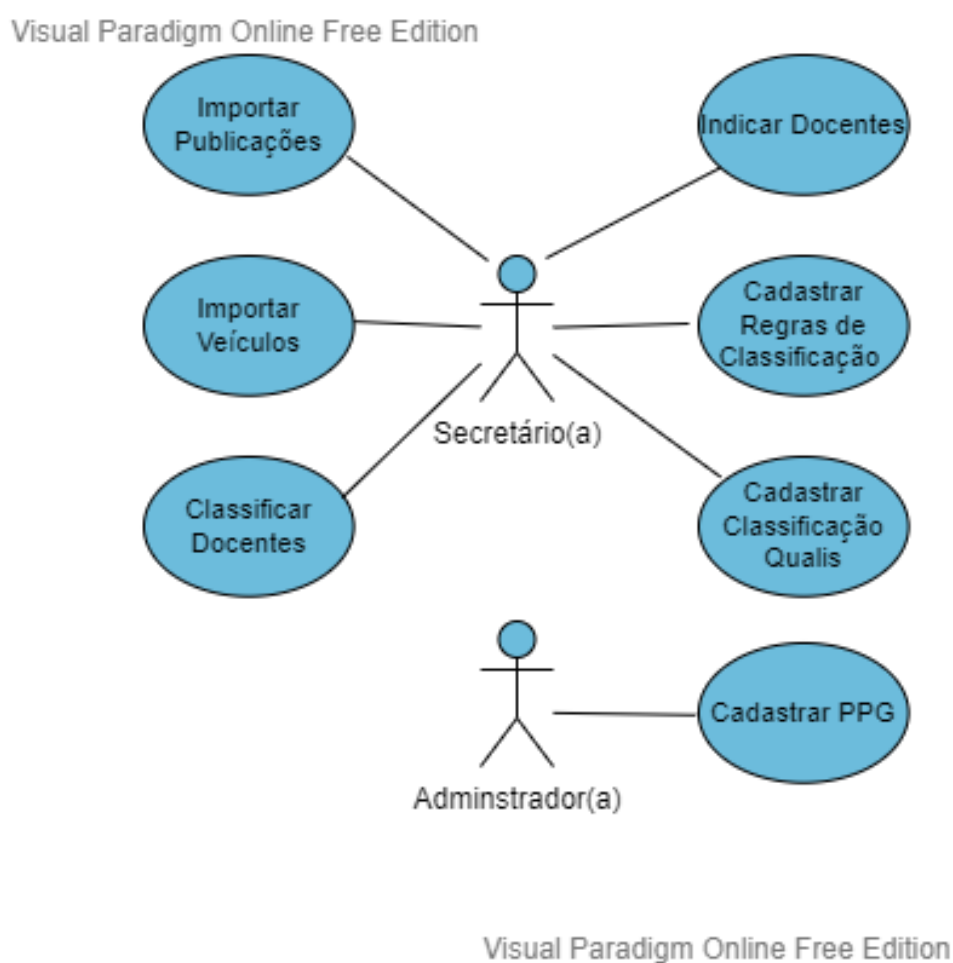


Figura 4 – Diagrama de casos de uso.

4 Arquitetura e Modelagem do Projeto

Neste capítulo serão mostradas a arquitetura e a modelagem FrameWeb do C2D. Na Seção 4.1 é apresentada a arquitetura seguida para o novo módulo; na Seção 4.2 é descrita a modelagem do sistema. No apêndice se encontra o Documento de Especificação do Projeto, que possui mais informações sobre a arquitetura desse projeto.

4.1 Arquitetura do sistema

O sistema C2D possui como requisito ser integrado ao Marvin. Desse modo, para facilitar a integração e possíveis melhorias das suas funcionalidades no futuro, foi utilizado o FrameWeb (SOUZA, 2020), que propõe a arquitetura de software na qual o Marvin foi desenvolvido. Tal arquitetura é baseada no padrão arquitetural Camada de Serviço (*Service Layer*) (FOWLER, 2002). Desse modo, qualquer aplicação que for desenvolvida sobre seu modelo deve ter as camadas e modelos descritos na Seção 2.3. A Figura 5 mostra a arquitetura do sistema C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin.

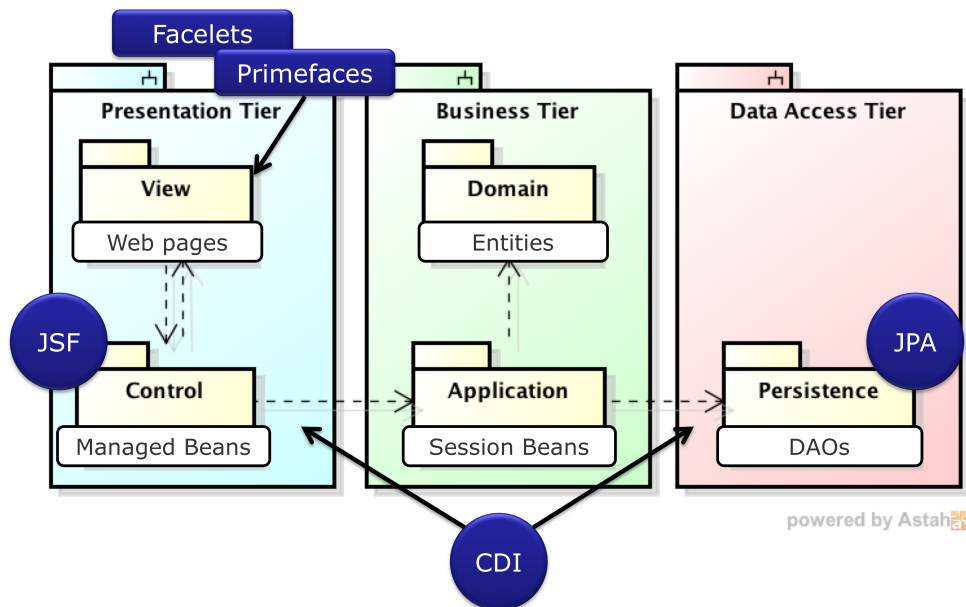


Figura 5 – Arquitetura de Software.

Conforme ilustrado na Figura 5 a arquitetura do software é dividida em três camadas:

- Camada de negócio (*Business Tier*);
- Camada de acesso de dados (*Data Access Tier*);

- Camada de apresentação (*Presentation Tier*).

Dentro da camada de negócio está presente o pacote da aplicação que possui as classes *Service Beans* com o papel de prover as funcionalidades do sistema como cadastro de PPG, listar regras, atualizar Qualis, desassociar docentes de um PPG, entre outros. Essas classes possuem conexão com a camada de acesso de dados, por possuírem dependência com o pacote de persistência para terem acesso aos dados do sistema, o controle dessa dependência é feita pela *framework* EJB do Java EE.

As classes *Service Beans* manipulam os objetos do pacote de domínio e os persistem por meio pacote de persistência. Na implementação dos objetos do pacote domínio são colocadas anotações que visam orientar o mapeamento objeto/relacional na camada persistência de seus dados, portanto esses objetos têm função de representar domínio do problema do software.

Como foi citado anteriormente as classes do pacote de persistência denominadas de DAO (*Data Access Object*) têm a função de fazer os acessos aos dados do sistema, em outras palavras armazenar/recuperar objetos do/no banco relacional. O banco de dados relacional utilizado por esse projeto é o MySQL, que utiliza a linguagem SQL como interface e com o auxílio do mapeamento objeto-relacional de estrutura realizada pelas classes do pacote *Domain* é possível fazer a comunicação entre o projeto e o banco de dados.

Na camada de apresentação estão presentes dois pacotes. O primeiro pacote é o *View* (Visão) que é o primeiro contato do usuário com o sistema, contendo páginas *Web*, folhas de estilo e scripts que juntos formam o leiaute das páginas do sistema. Para composição da visão foi utilizada a biblioteca de componentes PrimeFaces, que é compatível com o desenvolvimento de projetos Web com JSF.

O segundo pacote é Controle (*Control*) que possui as classes *Managed Beans* cujo objetivo é lidar com as solicitações proveniente do pacote *View*. As classes *Managed Beans* possuem dependência com as classes do pacote da aplicação para que consigam atender as requisições, assim como as classes *Service Beans* o controle de dependência é feito pelo EJB.

4.2 Modelagem do sistema

O C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin é um sistema Web cuja arquitetura utiliza *frameworks* comuns no desenvolvimento para esta plataforma. Desta forma, o sistema pode ser modelado utilizando a abordagem FrameWeb (SOUZA, 2020). Nas subseções a seguir, os modelos FrameWeb referentes a cada camada da arquitetura são apresentados.

4.2.1 Camada de Negócio

A camada de Negócio é dividida em duas: Lógica de Domínio (*Domain*) e Lógica de Aplicação (*Application*). A Figura 6 representa as entidades da lógica de domínio desse projeto.

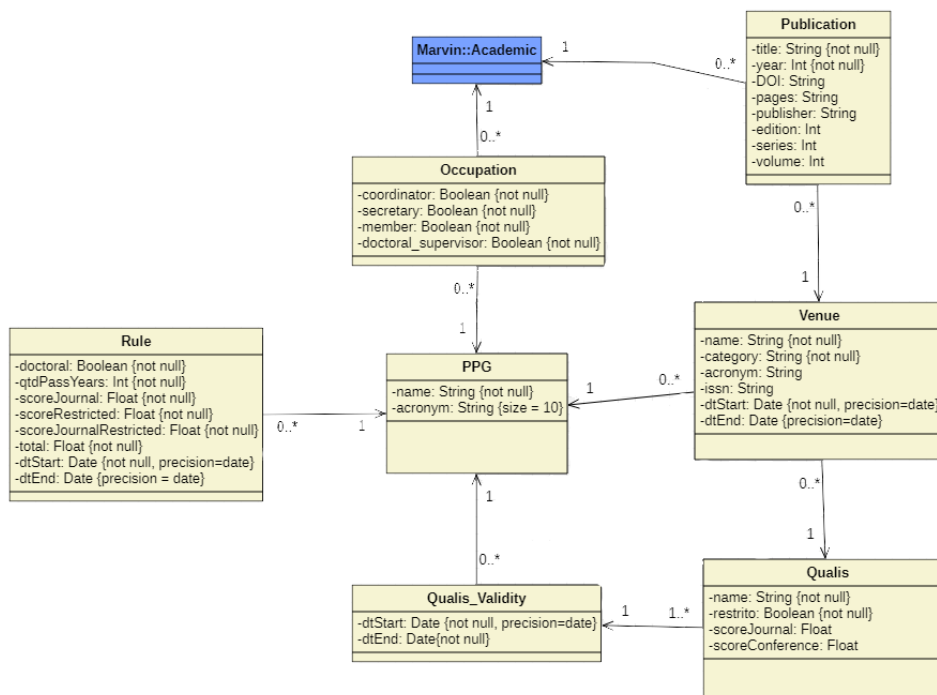


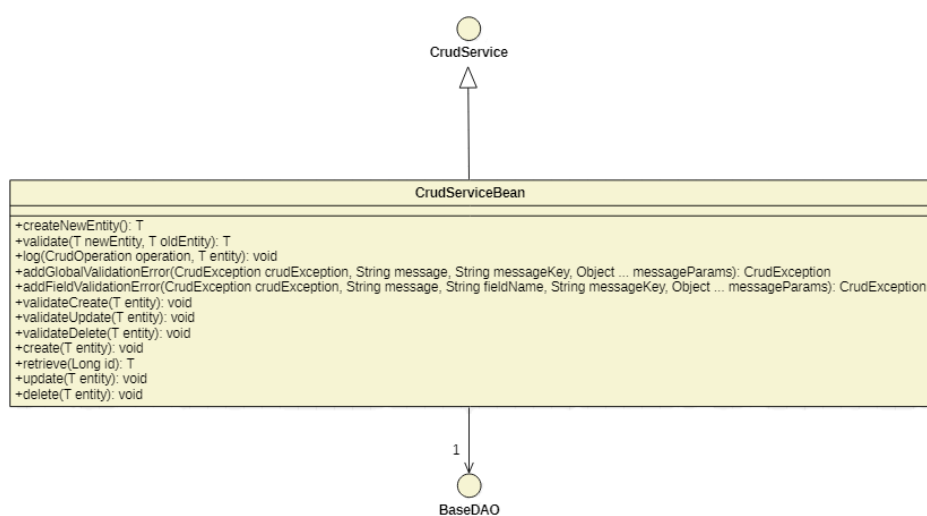
Figura 6 – Modelo de Entidades do C2D.

O modelo de entidades do C2D, representado na Figura 6, foi desenvolvido a partir dos requisitos documentados no Documento de Especificação de Requisitos. Com esse documento foi possível mapear todos os objetos contidos no domínio do problema definindo seus atributos e relacionamentos com outros objetos de acordo com a linguagem FrameWeb.

As regras de negócio são implementadas dentro das classes de serviço que utilizam as classes DAO (apresentadas na Seção 4.2.2) para utilizar os recursos do banco de dados. Algumas classes de serviço são destinadas a atender as funcionalidades de cadastro (CRUDs) do sistema, portanto herdam as características do modelo representado na Figura 7, que representa classes do utilitário JButler.

Na classe utilitária *CrudServiceBean* proveniente do pacote *JButler* estão implementadas funcionalidades de CRUD para uma entidade genérica, portanto utilizamos ela como herança no projeto para criação de novos serviços CRUD com o intuito de reutilização de código.

A Figura 8 representa as entidades da lógica de aplicação desse projeto. Para cada entidade do C2D foi implementado um *Service Bean* para fornecer funcionalidade CRUD

Figura 7 – Classe `CrudServiceBean`.

que a classe genérica `CrudServiceBean` não conseguia atender. Além dos serviços de CRUD existem outro serviço que atende outras necessidades como `UploadLattesCVServiceBean` e `CalculateScoresServiceBean`, sendo respectivamente utilizados para importar dados do currículo Lattes para o sistema e para calcular a pontuação do docente segundo a regra de Qualis vigente no sistema.

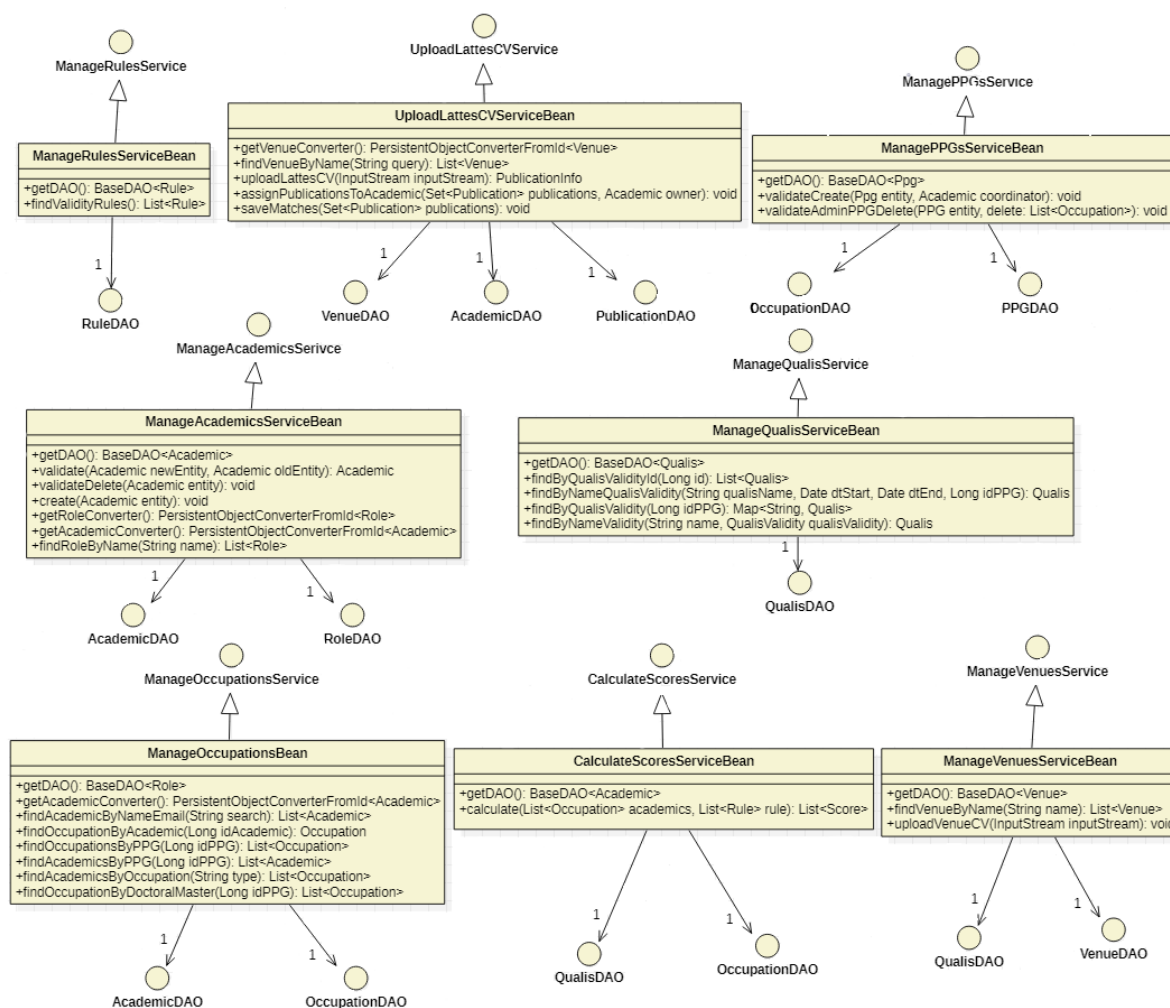


Figura 8 – Classes de Serviço do C2D.

4.2.2 Camada de Acesso a Dados

Na camada de acesso de dados deve se ter os modelos de persistência de uma aplicação FrameWeb. Os modelos de persistência desse projeto herdam as mesmas características da modelo de persistência da Figura 9, que representa as classes reutilizadas do JButler.

Com o objetivo de armazenar/recuperar objetos do/no banco de dados o JButler disponibiliza a Classe **BaseJPDAO** que contém um conjunto de métodos implementados para uma classe genérica possibilitando ser utilizada para qualquer entidade do projeto.

A Figura 10 representa os modelos de persistência principais para acesso a dados do módulo C2D.

A Figura 10 representa as classes DAO de todas as entidades do C2D, sendo que todas elas herdam as funcionalidades da Classe **BaseJPDAO** e adicionam novas funcionalidade de acordo com as necessidades da entidade que ela referencia.



Figura 9 – Classe BaseJPADAO.

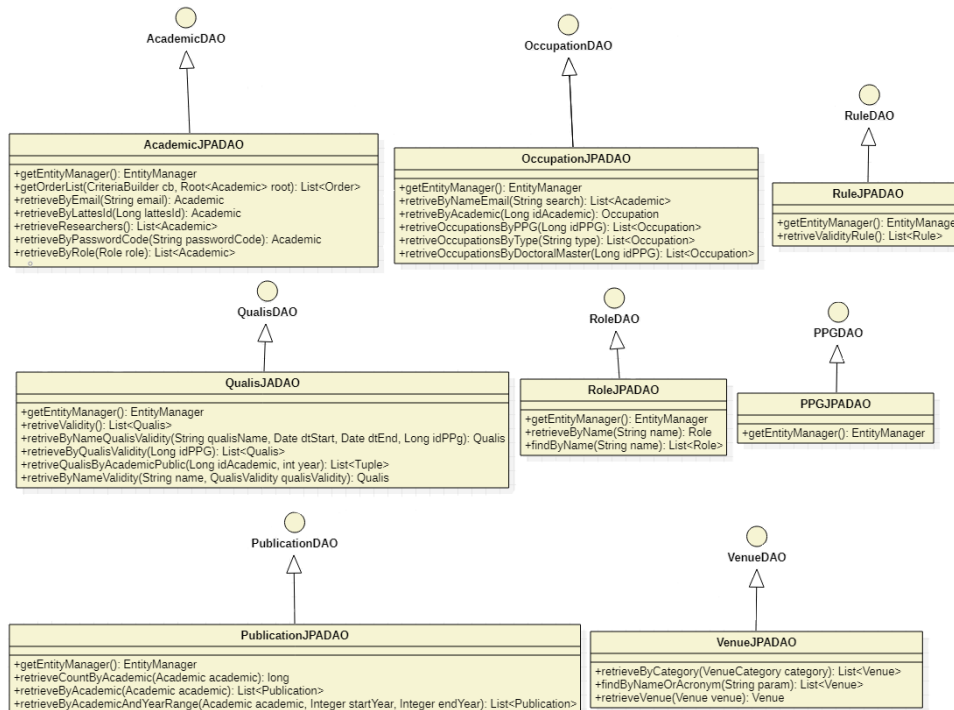


Figura 10 – Objetos DAO do C2D.

4.2.3 Camada de Apresentação

A camada de apresentação lida diretamente com o usuário, provendo o acesso às funcionalidades do sistema. A maioria das funcionalidade do módulo C2D são cadastros (CRUDs), por isso de maneira geral elas seguem o modelo de navegação da Figura 11.

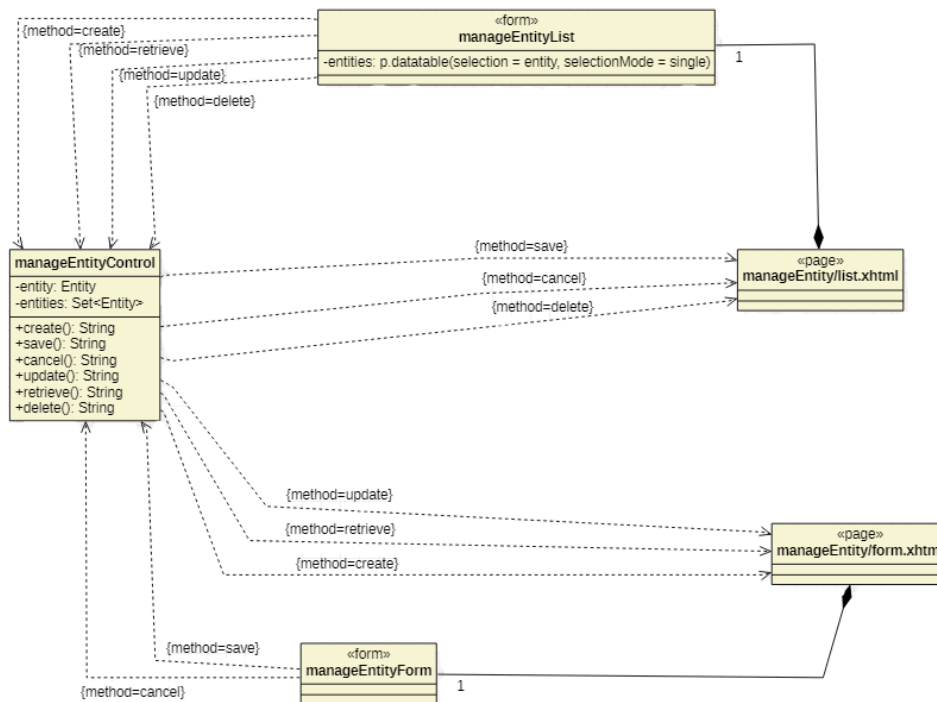


Figura 11 – Modelo de navegação CRUD.

Na Figura 11 está representada a navegação das telas CRUD do sistema. Inicialmente o usuário tem contato com a página `list.xhtml` que lista as entidades salvas no banco. Estas entidades são listadas pelo `manageEntityList` utilizando `p:datatable` do PrimeFaces. Se o usuário não selecionar nenhuma entidade a única funcionalidade disponível é a `create` que é o método do controlador utilizado para criar um novo objeto.

No caso de selecionar um objeto estarão disponíveis as funções de `update`, `delete` e `retrieve`, de modo que possuem respectivamente as seguintes funções para o objeto selecionado: atualizar, excluir e recuperar. As quatro funções citadas anteriormente fazem parte do controlador da página que para as funções de criar, atualizar e recuperar redireciona para página `form.xhtml` onde poderão ser feitas as alterações da entidade e salvas no banco de dados com o método `save`.

O cálculo de pontuação dos docentes não segue o modelo de navegação padrão dessa forma ele é representado pela Figura 12.

No modelo de navegação do cálculo de pontuação dos docentes o primeiro contato do usuário é com a página `index.xhtml` que possui uma lista de docentes que terão sua pontuação calculada. O usuário pode executar a função `addAcademic` para adicionar um

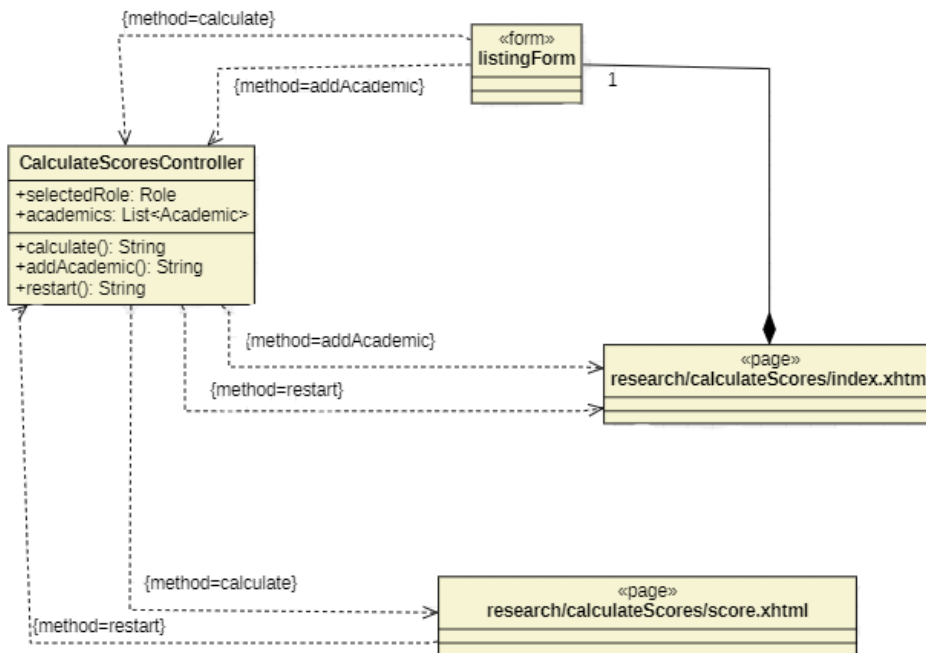


Figura 12 – Modelo de navegação do cálculo de pontuação dos docentes.

novo acadêmico a ser avaliado e caso ele selecione um elemento da lista ele pode executar a função `removeAcademic` para remover um acadêmico da avaliação.

O método do controlador responsável por exibir o cálculo do score é o `calculate`, que quando tiver realizado todos os cálculos necessários com a lista de docentes irá redirecionar para página `score.xhtml` para exibição do resultado e se caso ele deseje retornar à tela inicial e refazer os cálculos teve se utilizar o método `restart`.

5 Apresentação

Neste capítulo são apresentadas diversas capturas de tela do sistema, a fim de demonstrar o resultado obtido. Na Seção 5.1 é demonstrada a funcionalidade de login; na Seção 5.2 são apresentados o cadastro de PPGs e a indicação de secretário(a) ou coordenador(a); na Seção 5.3 é mostrado o cadastro dos Qualis; na Seção 5.4 o local de cadastro de Venues e sua importação para o sistema; na Seção 5.5 é demonstrado como cadastrar uma regra de publicação mínima no C2D; na Seção 5.6 é mostrada a importação de dados do currículo Lattes dos docentes; finalmente, na Seção 5.7 é apresentado o cálculo da pontuação dos docentes associados.

5.1 Login

A Figura 13 representa a tela de login padrão do sistema, onde qualquer tipo de usuário coloca sua credencial para ter acesso ao sistema.

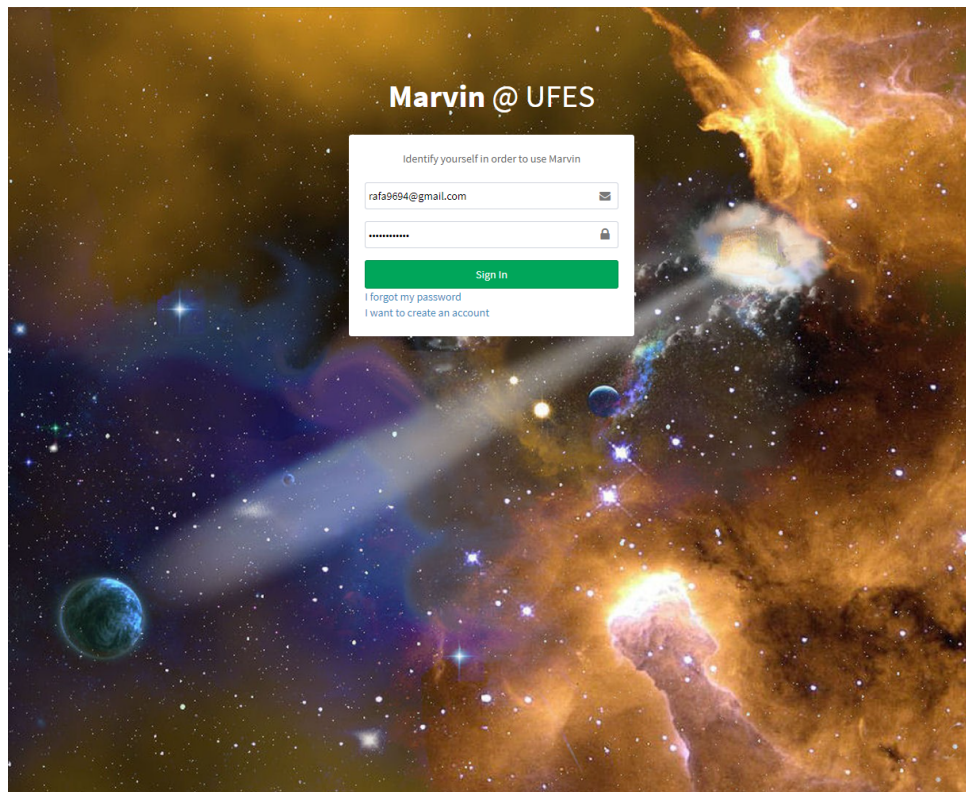


Figura 13 – Tela de Login.

Após a efetuação do login o usuário terá acesso a tela Home do sistema, representada pela Figura 14, no menu lateral poderá ter acesso aos módulos disponíveis no sistema caso ele possua permissão de acesso. Os módulos disponíveis são:

- Gerenciar Acadêmicos (*Manage Academics*);
- Gerenciar PPGs (*Manage PPGs*);
- Importar Currículos do Lattes (*Upload Lattes CV*);
- Gerar Bibliografia (*Generate Bibliography*);
- Gerenciar Local de Publicação (*Manage Venues*);
- Gerenciar Qualis (*Manage Qualis*);
- Gerenciar Regras (*Manage Rules*);
- Calcular Pontuação (*Calculate Scores*).

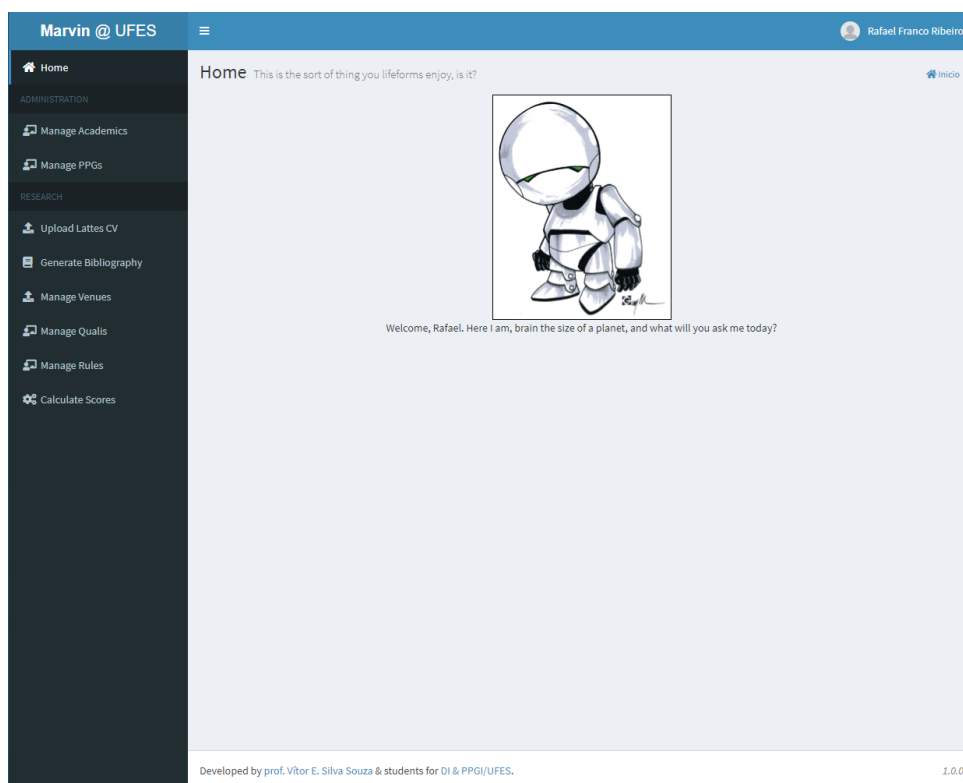


Figura 14 – Tela de Home.

Como o C2D foi integrado ao Marvin, foram utilizadas algumas funcionalidades já desenvolvidas pelo Marvin como o login ao sistema, Gerenciar Acadêmicos e Importar Currículos do Lattes. Desse modo as próximas seções serão apresentadas na ordem do fluxo padrão para se ter todos os dados necessários para garantir que o C2D possa calcular a pontuação de cada acadêmico indicado e informar pela regra que ele se encaixa.

5.2 Manage PPGs

A Figura 15 mostra a tela de Gerenciamento de PPGs. Essa tela permite cadastrar novos PPGs, atualizar e excluir. Além disso, se selecionar um PPG o botão **Administrators** fica disponível para uso e, ao apertá-lo, o sistema se redireciona à tela da Figura 16, responsável por indicar o acadêmico cuja função será de secretário ou coordenador para esse PPG.

The screenshot displays the 'Manage PPGs' interface. At the top, the user is logged in as 'Rafael Franco Ribeiro'. The main content area features a table with the following data:

Id	Name	Acronym
67	Programa de Pós-Graduação em História	PPGUH
19358	Programa de Pós-graduação em Informática	PPG

Below the table, there is a pagination control showing '10' items per page and '(1 of 1)' total pages. Action buttons for 'New', 'Administrators', 'Modify', and 'Delete' are visible. The left sidebar contains navigation options under 'ADMINISTRATION' and 'RESEARCH'. The footer indicates the application was developed by prof. Vítor E. Silva Souza & students for DI & PPGI/UFES, with a version number of 1.0.0.

Figura 15 – Tela dos PPG.

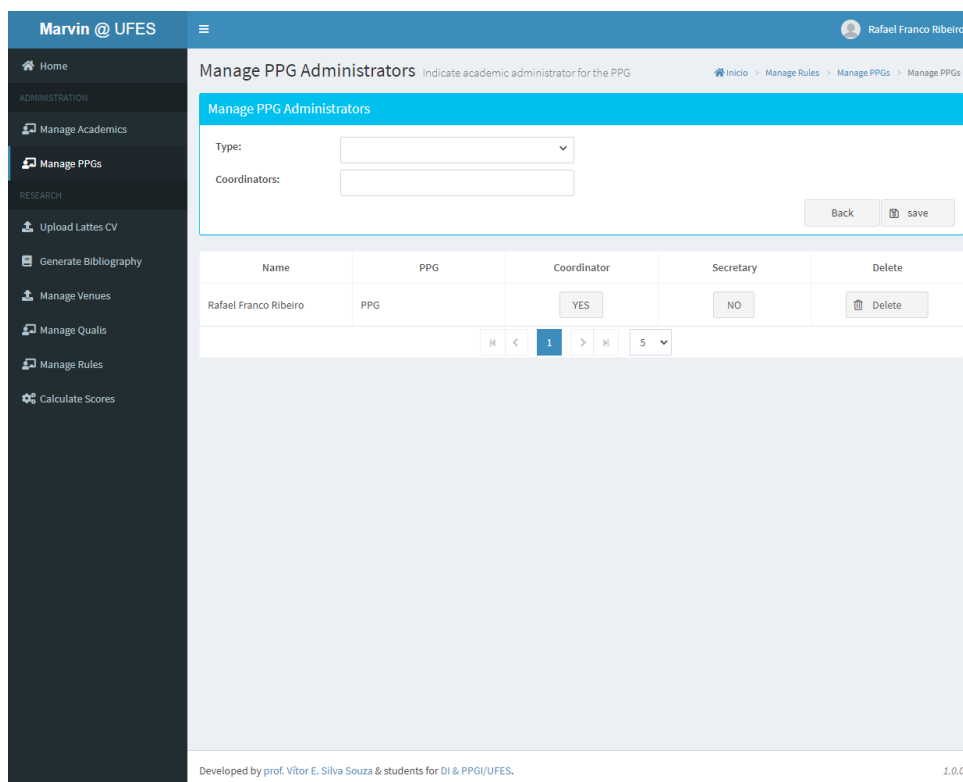


Figura 16 – Tela de Administração de um PPG.

5.3 Manage Qualis

Após o cadastro do PPG e da a indicação do secretário/coordenador, deve-se ir ao módulo de gerenciamento de Qualis, que possui as funcionalidades de cadastro, atualização, visualização e exclusão de um Qualis. Sendo o Qualis considerado vigente aquele que não possui uma data de fim de vigência, como é representado na Figura 17, na qual todos os Qualis presentes estão vigentes.

Para se ir à tela de cadastro de Qualis, representada pela Figura 18, deve-se clicar no botão **New** na tela de Gerenciamento de Qualis. Nesta, é possível cadastrar diversos Qualis sem retornar para a tela inicial do módulo.

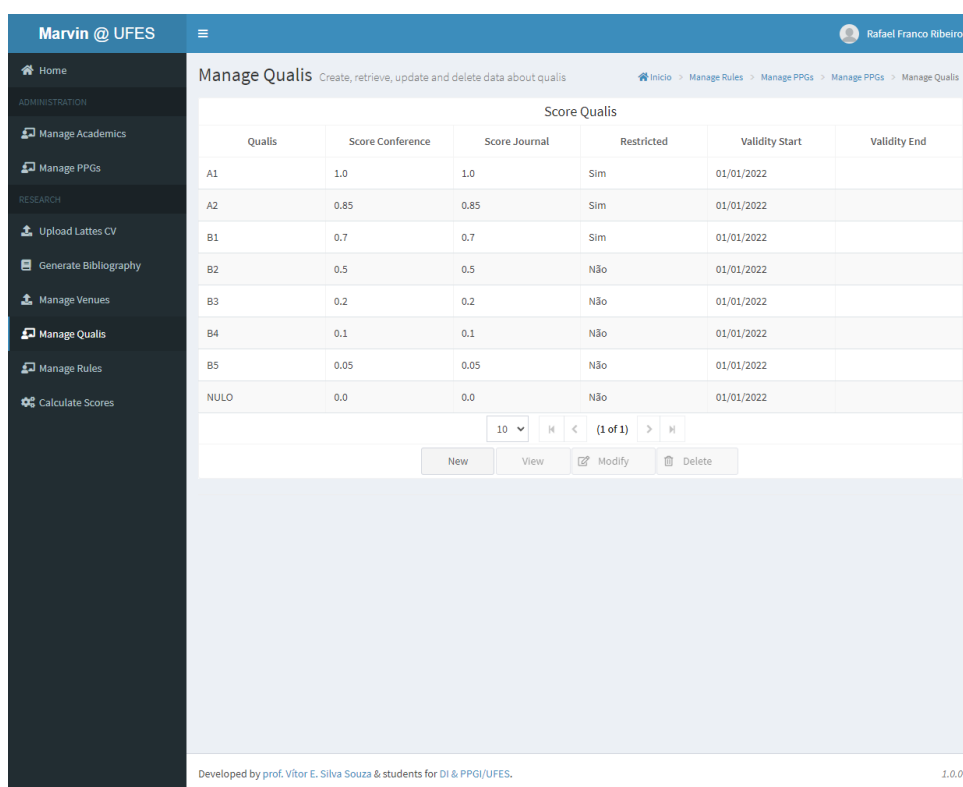


Figura 17 – Tela de Gerenciamento de Qualis.

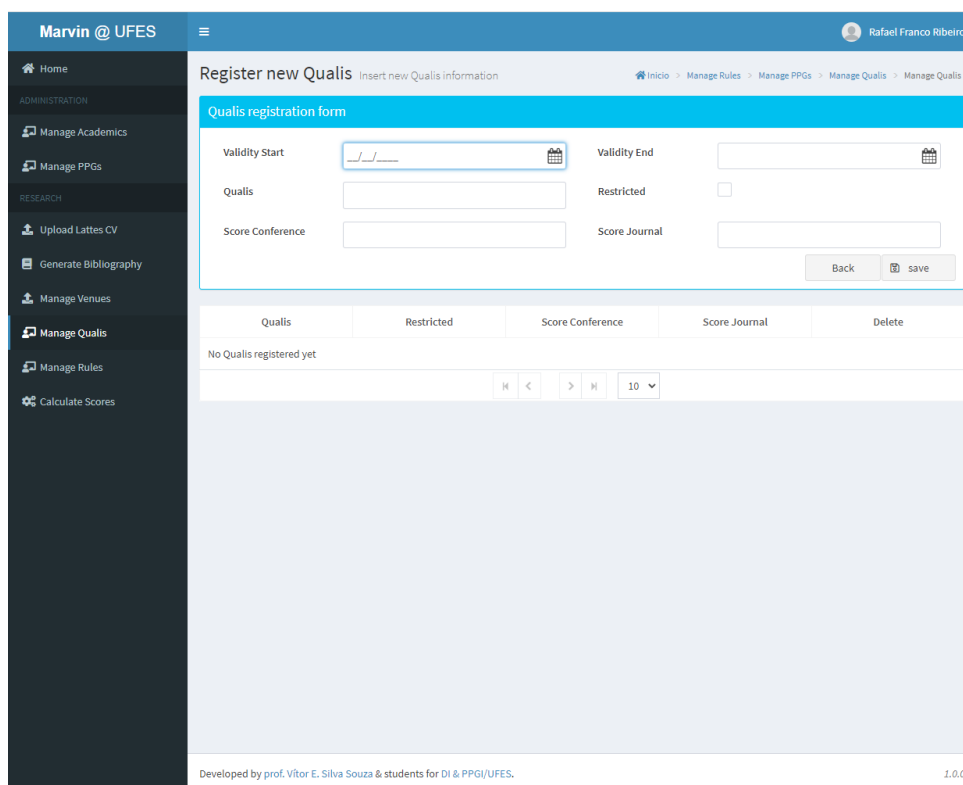


Figura 18 – Tela de cadastro de Qualis.

5.4 Manage Venue

Na Figura 19 está a tela inicial do módulo de gerenciamento de Venue, onde o secretário/coordenador pode cadastrar a relação entre o local de publicação dos trabalhos acadêmicos e a pontuação Qualis. Portanto é necessário que seja feito o cadastro de todos os Qualis antes da importação dos Venues visto que o Qualis é uma informação obrigatória para o Venue.

Type	Name	Acronym	ISSN
Journal	4OR (BERLIN)		1619-4500
Journal	A.A.N.A. JOURNAL		0094-6354
Journal	AACE INTERNATIONAL TRANSACTIONS		1528-7106
Journal	AASRI PROCEDIA		2212-6716
Journal	ABAKÓS		2316-9451
Journal	ABCD : ARQUIVOS BRASILEIROS DE CIRURGIA DIGESTIVA		0102-6720
Journal	ACADEMIC JOURNAL OF SCIENCE		2165-6282
Journal	AÇÃO ERGONÔMICA		1519-7859
Journal	ACCREDITATION AND QUALITY ASSURANCE		0949-1775
Journal	ACIS INTERNATIONAL JOURNAL OF COMPUTER AND INFORMATION SCIENCE		1525-9293

Figura 19 – Tela de Gerenciamento de Venue.

Devido à quantidade de diferentes Venues, para o cadastro de Venues foi criada a funcionalidades de extração de informação a partir de um arquivo CSV com as informações separadas por ";". Desse modo, as informações devem estar na seguinte ordem: tipo (Conference ou Journal), sigla, nome, ISSN e nome do Qualis.

Assim como o cadastro de Qualis, os Venues também possuem um período de vigência e, caso o período de fim de vigência não esteja selecionado no momento da importação, este então será considerado como os Venues vigentes. A tela de importação de Venue é mostrada na Figura 20.

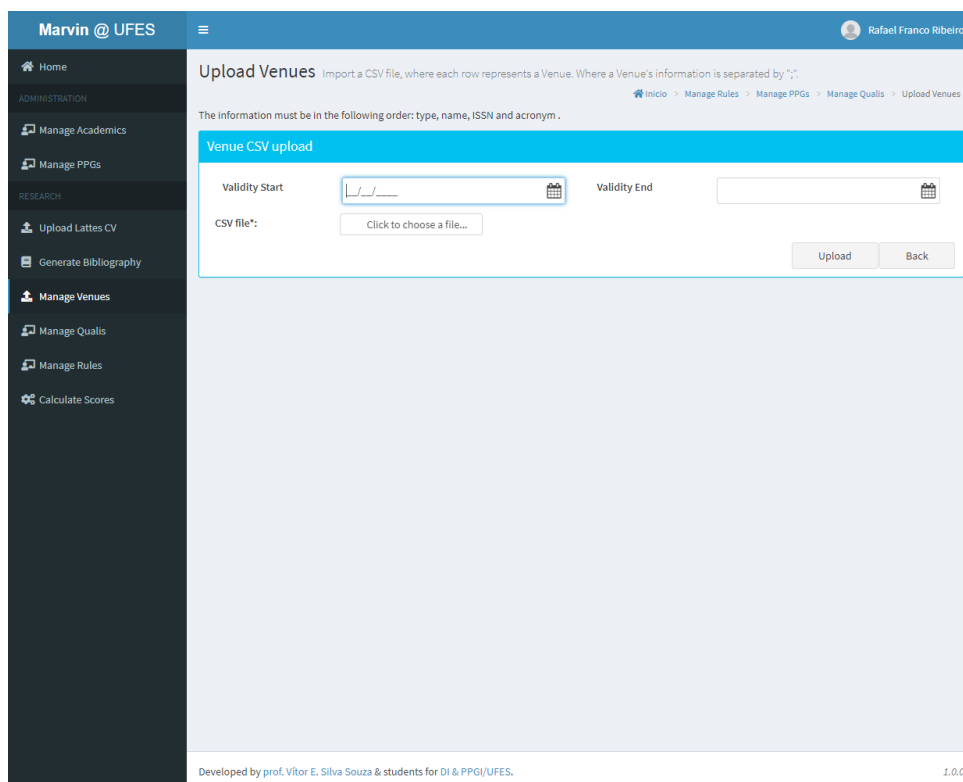


Figura 20 – Tela de Importação de Venue.

5.5 Manage Rules

Na Figura 21 é mostrada a tela inicial do gerenciamento das regras. Nas funcionalidades contidas a partir dessa tela, é possível cadastrar as regras que os acadêmicos selecionados serão avaliados.

A regra vigente que será utilizada para a avaliação dos docentes, sendo a definição de regras vigente uma regra que não possui final de vigência.

Para o cadastro de uma nova regra deve-se clicar no botão de **New** para ir à tela da Figura 22, onde é realizado o cadastro. Com exceção do campo **Final Date** todos os outros devem ser preenchidos.

Não é possível repetir o tipo de regra cadastrada para um mesmo período e caso não exista nenhum critério sobre algumas das pontuação deve se colocar o valor de 0.

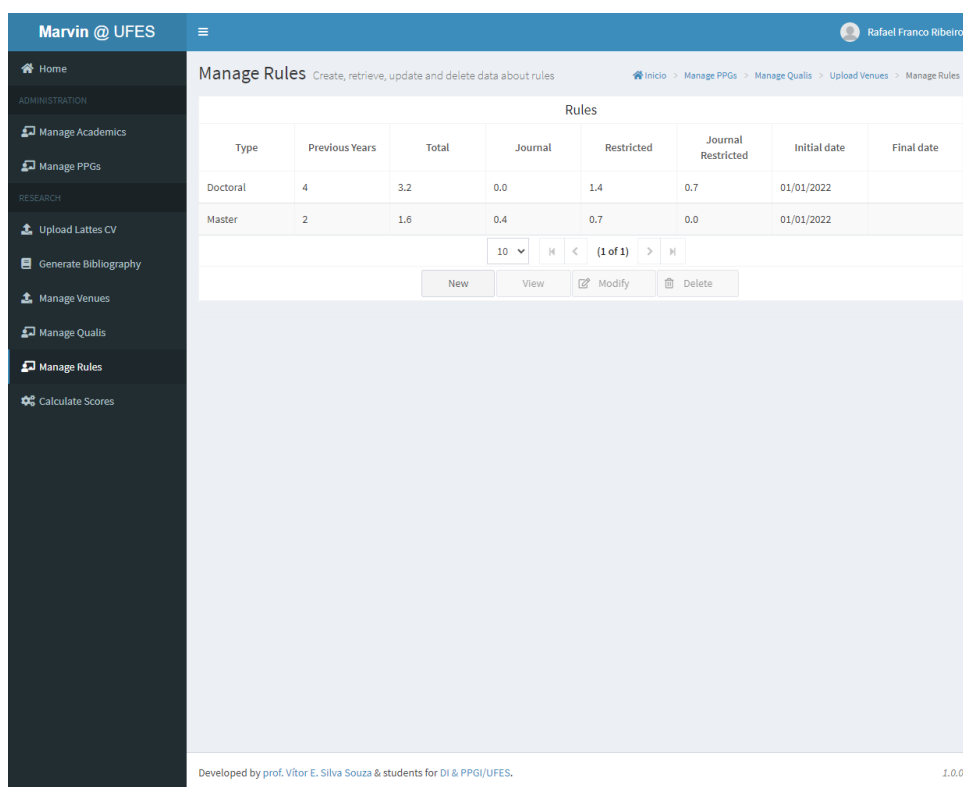


Figura 21 – Tela de Gerenciamento de Regras.

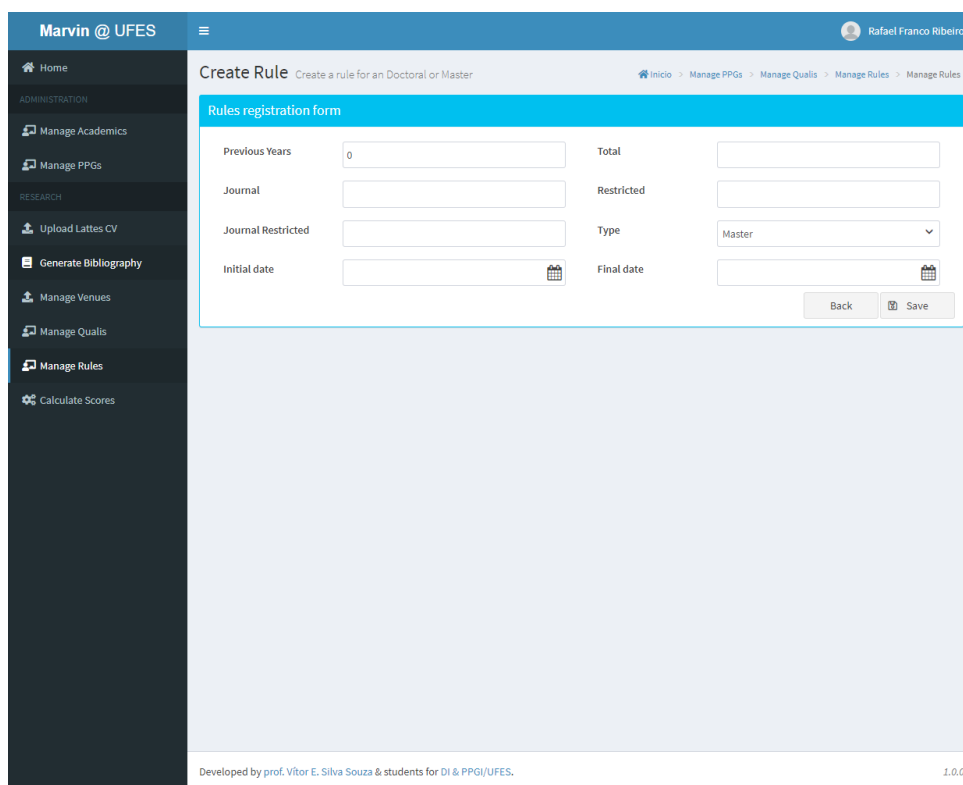


Figura 22 – Tela de Cadastro de Regras.

5.6 Upload lattes CV

O módulo Importação de currículo Lattes era uma funcionalidade previamente desenvolvida que, além de fazer importação dos dados de publicações dos docentes, também faz a associação das publicações com o Venue como é mostrado na Figura 25.

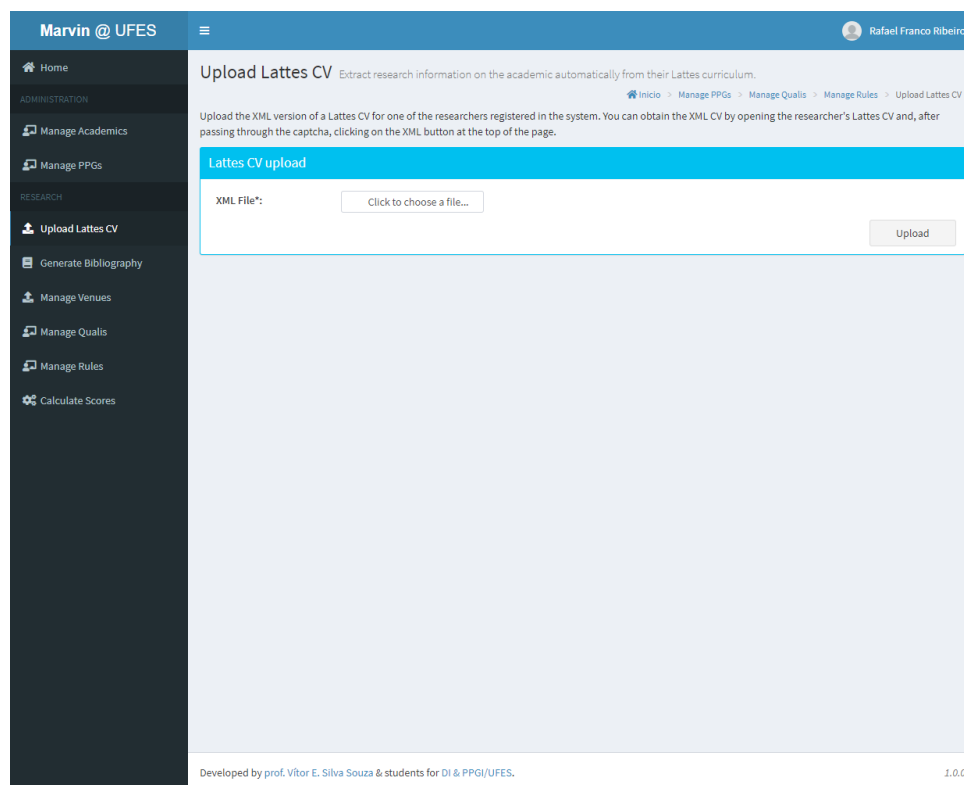


Figura 23 – Tela de Importação de Currículo Lattes.

Ao se carregar o XML no campo XML File* e clicar em Upload na tela inicial será redirecionado a tela de confirmar, vide Figura 24, que após confirmar salvará as publicações no banco de dados.

A tela exibida na Figura 25 é renderizada após a confirmação das publicações. Para que seja possível associar o Venue com a publicação, o secretário/coordenador clica na quarta coluna da tabela para pesquisar e selecionar a Venue correta. Ao final das associações, clicar no botão confirmar presente no final da tabela.

Marvin @ UFES Rafael Franco Ribeiro

Upload Lattes CV Extract research information on the academic automatically from their Lattes curriculum.

Please review the publications extracted from the uploaded CV and confirm you wish to assign them to Vítor Estêvão Silva Souza. Note that all publications currently assigned to this researcher will be deleted and replaced by the ones below. This researcher currently has 63 publications and will have 63 publications.

Type	Title	Year
Journal	An ontological analysis of software system anomalies and their associated risks	2021
Conference	Source Code Interoperability based on Ontology	2021
Conference	Towards a Characterization to Aid in Ontology Reuse	2021
Journal	Finding reusable structured resources for the integration of environmental research data	2020
Chapter	The FrameWeb Approach to Web Engineering: Past, Present and Future	2020
Conference	GO-FOR: A Goal-Oriented Framework for Ontology Reuse	2019
Conference	GORO 2.0: Evolving an Ontology for Goal-Oriented Requirements Engineering	2019
Journal	GRALD: an Approach for Goal and Risk Analysis in the Development of Information Systems for the Web of Data	2019
Conference	OOC-O: A Reference Ontology on Object-Oriented Code	2019
Journal	Reducing power companies billing costs via empirical bayes and seasonality remover	2019
Conference	Representação Ontológica de Frameworks de Mapeamento Objeto/Relacional	2019
Conference	Using GORO to Provide Ontological Interpretations of iStar Constructs	2019
Conference	iStar2.0-OWL: an Operational Ontology for iStar	2019
Conference	A Domain-Specific Language for Fault Diagnosis in Electrical Submersible Pumps	2018
Journal	Engineering Self-Adaptive Software Systems	2018
Conference	Goal and Risk Analysis in the Development of Information Systems for the Web of Data	2018
Conference	Integração de um Gerador de Código ao FrameWeb Editor	2018
Conference	Ontological Representation of Relational Databases	2018

Figura 24 – Tela de Confirmação de Importação.

Marvin @ UFES Rafael Franco Ribeiro

Upload Lattes CV Extract research information on the academic automatically from their Lattes curriculum.

Successfully assigned 63 publications to Vítor Estêvão Silva Souza. Any previously assigned publications were deleted. There was an attempt to match the papers with known venues. Please review, change and confirm to save the matches.

Year	Title	Declared Venue	Known Venue
2021	An ontological analysis of software system anomalies and their associated risks	DATA & KNOWLEDGE ENGINEERING	(No venue matched)
2021	Source Code Interoperability based on Ontology	Proc. of the 17th Brazilian Symposium on Information Systems	(No venue matched)
2021	Towards a Characterization to Aid in Ontology Reuse	Proc. of the 14th Seminar on Ontology Research in Brazil (ONTOBRAS 2021)	(No venue matched)
2020	Finding reusable structured resources for the integration of environmental research data	ENVIRONMENTAL MODELLING & SOFTWARE	(No venue matched)
2019	GO-FOR: A Goal-Oriented Framework for Ontology Reuse	2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI)	(No venue matched)
2019	GORO 2.0: Evolving an Ontology for Goal-Oriented Requirements Engineering	Advances in Conceptual Modeling, ER 2019 Workshops FAIR, MREBA, EmpER, MoBID, OntoCom, and ER Doctoral Symposium Papers	(No venue matched)
2019	GRALD: an Approach for Goal and Risk Analysis in the Development of Information Systems for the Web of Data	JOURNAL OF SOFTWARE ENGINEERING RESEARCH AND DEVELOPMENT	(No venue matched)
2019	OOC-O: A Reference Ontology on Object-Oriented Code	Conceptual Modeling - 38th International Conference, ER 2019, Salvador, Brazil, November 4-7, 2019, Proceedings	(No venue matched)
2019	Reducing power companies billing costs via empirical bayes and seasonality remover	ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE	(No venue matched)
2019	Representação Ontológica de Frameworks de Mapeamento Objeto/Relacional	Proc. of the 12th Seminar on Ontology Research in Brazil (ONTOBRAS 2019)	(No venue matched)
2019	Using GORO to Provide Ontological Interpretations of iStar Constructs	Proc. of the 12th International I* Workshop (iStar 2019)	(No venue matched)

Figura 25 – Tela de Associação da Publicação com o Venue.

5.7 Calculate Score

Com todos os dados reunidos nos fluxos anteriores o C2D está habilitado para calcular as pontuação de cada docente associado. Na Figura 26 está a tela com a funcionalidade de indicar um docente a ser avaliado.

Deve pesquisar o docente pelo campo **Academic** e selecionar o tipo pelo campo **Type** e clicar no botão **Save** para indicá-lo para avaliação. Para trocar o tipo do docente basta pesquisá-lo no campo de **Academic** e trocar seu tipo. A tela também possui a funcionalidade de removê-lo.

Name	Master	Doctoral
Vitor Estêvão Silva Souza	NO	YES

Figura 26 – Tela Indicação de Docente para Avaliação.

Com os docentes indicados, o secretário/coordenador pode iniciar o processo de cálculo de pontuação a partir do botão **Calculate**. O código-fonte relativo ao processo de cálculo da pontuação é exibido nas listagens 5.1 e 5.2.

O processo de cálculo de pontuação é feito principalmente pela função `calculate` dentro da classe de Serviço `CalculateScoresServiceBean`. As informações de entradas são a lista de acadêmicos indicados e as regras vigentes.

Primeiramente é feito mapeamento da lista de regras para facilitar o acesso a regras a ser aplicada. A regra a ser aplicada no docente é obtida através da utilização do tipo de ocupação do docente. Depois é feito um `loop` onde passaremos por todos os docentes indicados.

Listagem 5.1 – Função principal para o calculo da pontuação - Parte 1.

```

1 public List<Score> calculate(List<Occupation> academics, List<Rule> rule) {
2     List<Score> score = new ArrayList<Score>();
3     Map<String, Rule> mapRule = new HashMap<String, Rule>();
4
5     Date today = new Date();
6     SimpleDateFormat yearsFormat = new SimpleDateFormat("yyyy");
7
8     for (Rule elem : rule) {
9         String key = elem.isDoctoral() ? "doctoral" : "master";
10        mapRule.put(key, elem);
11    }
12
13    for (Occupation academic : academics) {
14        Rule activeRule = academic.isDoctoral_supervisor() ? mapRule.get("doctoral")
15            : mapRule.get("master");
16        int qtdPassYears = activeRule.getQtdPassYears();
17
18        Calendar c = Calendar.getInstance();
19        c.setTime(today);
20
21        c.add(Calendar.YEAR, -qtdPassYears);
22        Date currentDate = c.getTime();
23
24        int year = Integer.parseInt(yearsFormat.format(currentDate));
25
26        Long academicId = academic.getAcademic().getId();
27        String academicName = academic.getAcademic().getName();
28
29        String type = activeRule.isDoctoral() ? "Doctoral" : "Master";
30
31        try {
32            List<Tuple> tupleDotted = qualisDAO.retrieveQualisByAcademicPublic(
33                academicId, year);
34
35            BigDecimal totalDotted = new BigDecimal(0);
36            BigDecimal journalDotted = new BigDecimal(0);
37            BigDecimal restrictedDotted = new BigDecimal(0);
38            BigDecimal journalRestrictedDotted = new BigDecimal(0);
39            for (Tuple tuple : tupleDotted) {
40                Venue venueDotted = (Venue) tuple.get(0);
41                Qualis qualisDotted = (Qualis) tuple.get(1);
42
43                VenueCategory category = venueDotted.getCategory();
44
45                BigDecimal journalQualis = new BigDecimal(0);
46                BigDecimal conferenceQualis = new BigDecimal(0);
47                if (category.getName().equalsIgnoreCase("JOURNAL")) {
48                    journalQualis = BigDecimal.valueOf(qualisDotted.getScoreJournal());
49                } else if (category.getName().equalsIgnoreCase("CONFERENCE")) {
50                    conferenceQualis = BigDecimal.valueOf(qualisDotted.getScoreConference());
51                }
52                totalDotted = totalDotted.add(journalQualis);
53                totalDotted = totalDotted.add(conferenceQualis);
54                journalDotted = journalDotted.add(journalQualis);
55
56                if (qualisDotted.isRestrito()) {
57                    restrictedDotted = restrictedDotted.add(journalQualis);
58                    restrictedDotted = restrictedDotted.add(conferenceQualis);
59                    journalRestrictedDotted = journalRestrictedDotted.add(journalQualis);
60                }
61            }
62        }
63    }
64 }

```

Listagem 5.2 – Função principal para o calculo da pontuação - Parte 2.

```
1     boolean approved = false;
2     if (total >= activeRule.getTotal() && journal >= activeRule.getScoreJournal
3         ()
4         && restricted >= activeRule.getScoreRestricted()
5         && journalRestricted >= activeRule.getScoreJournalRestricted()) {
6         approved = true;
7     }
8     Score academicScore = new Score(academicName, type, total, journal,
9         restricted, journalRestricted,
10        approved);
11    score.add(academicScore);
12
13    } catch (PersistentObjectNotFoundException |
14        MultiplePersistentObjectsFoundException e) {
15        // TODO Auto-generated catch block
16        e.printStackTrace();
17    }
18 }
19
20 return score;
21 }
```

Em seguida pegamos da regra compatível com o docente o ano inicial que as publicações poderão pontuar. Desse modo teremos as informações necessárias para pegar as pontuações das publicações dos docentes no período requisitado.

A função responsável por trazer essas informações é `retrieveQualisByAcademicPublic` que recebe como entrada o Id do acadêmico e o ano inicial da avaliação. Dentro dela é feita associação entre publicação, Venue e Qualis caso a publicação não tenha um Venue associado ela não será pontuada.

Então a função retorna uma lista de tupla de Venue e Qualis com isso temos as informações necessárias para calcular as pontuações. Criando um elemento da classe `Score` que armazena as informações calculadas para exibição. Teremos assim a lista de *scores* para serem exibidas na Figura 27.

As regras que o docente não tinha aplicado apareceram como NOT APPLIED e com o botão `Restart` é possível voltar à página de indicação para reiniciar o processo de avaliação.

The screenshot displays the 'Score Calculation Result' page in the Marvin @ UFES system. The page features a dark sidebar on the left with navigation options under 'ADMINISTRATION' and 'RESEARCH'. The main content area shows a table with the following data:

Name	Journal	Restricted	Journal Restricted	Total	Rule Doctoral	Rule Master
Vitor Estêvão Silva Souza	4.9	8.9	4.4	11.6	YES	NOT APPLIED

Below the table, there is a pagination control showing '10' items per page and '(1 of 1)' records. A 'Restart' button is located below the pagination controls. The footer of the page indicates it was developed by prof. Vitor E. Silva Souza & students for DI & PPGI/UFES, with a version number of 1.0.0.

Figura 27 – Tela exibição da avaliação.

6 Conclusão

A Seção 6.1 apresenta as considerações finais deste trabalho, suas principais contribuições, limitações, dificuldades enfrentadas ao longo do desenvolvimento do trabalho e perspectivas de trabalhos futuros. A Seção 6.2 apresenta melhorias para o C2D a partir de uma análise de suas limitações.

6.1 Considerações Finais

Neste trabalho foi proposta a criação de um módulo de credenciamento e classificação dos docentes integrado com o sistema Marvin, com o intuito de otimizar o processo de avaliação dos docentes do PPGI pelas suas publicações. Para apoiar o desenvolvimento do projeto foram definidos objetivos no Capítulo 1 que foram alcançados ao longo do seu desenvolvimento. Cada objetivo estabelecido contribuiu para o trabalho. Por exemplo, a produção do documento de análise e especificação de requisitos contribuiu para se ter uma visão ampla do projeto para que pudéssemos saber quais funcionalidades o projeto deveria ter. Já a construção do documento de projeto de software contribuiu diretamente para a construção do protótipo do sistema para que o padrão de arquitetura do Marvin e das boas práticas de programação estivessem presentes no módulo C2D.

Apesar do roteiro apresentado no Capítulo 1, que foi seguido ao longo do trabalho e ajudou muito na realização da proposta, foram surgindo dificuldades ao longo do projeto. Uma das principais dificuldades foi a modelagem do processo de credenciamento e classificação dos docentes devido ao alto número de regras e sub-regras, assim o módulo processa apenas as regras referentes às publicações do docente.

Outra dificuldade foi o grande volume de dados que o sistema deve tratar por exemplo, as publicações dos docentes que são importados para o sistema todos os trabalhos produzidos relevantes para sua vida acadêmica e as importações veículos de publicações que possuem uma quantidade elevada de dados.

O trabalho teve inúmeros desafios, solucionados através dos conhecimentos adquiridos nas disciplinas do curso de Engenharia da Computação, como Engenharia de Software, Programação Orientada a Objetos e Banco de Dados. Parte do conhecimento empregado nessas soluções veio também das experiências profissionais na área de software adquiridas por meio de estágios, bolsas e empregos.

6.2 Trabalhos Futuros

O ciclo da vida de um software não termina após a sua implementação. Ele ainda será utilizado durante muito tempo. Desta forma, os problemas que ele foi criado para resolver podem ter mudado suas regras, a tecnologia que foi utilizada em sua construção pode ser descontinuada ou precisar ser atualizada devido a falhas de segurança ou de funcionamento. Portanto, ele deve sofrer manutenções constantes, melhorias e correções nas suas funcionalidade para que o sistema continue a ser relevante.

Tendo em vista esse cenário de melhoria e fazendo uma análise dos resultados obtidos do trabalho, podemos observar uma série de limitações que podem gerar novos trabalho e projetos. Abaixo está uma lista com algumas limitações:

- Aprimorar a avaliação dos docentes para utilizar regras de períodos anteriores ao da vigência atual, para de permitir uma comparação entre as regras antigas com a atual;
- Permitir ao usuário gerar um arquivo com pontuação dos docentes em diferentes formatos como, por exemplo, Excel, CSV e PDF;
- Aprimorar a avaliação dos docentes para também avaliar com base na suas orientações, docências e carga administrativa;
- Aprimorar importações de dados dos veículos e publicações para que seja feita de forma assíncrona, permitindo aos usuários utilizar o sistema enquanto aguardam o fim das importações.

Referências

- BARCELLOS, M. P. Engenharia de software: notas de aula. Vitória: UFES, 2018. Disponível em: <<https://nemo.inf.ufes.br/wp-content/uploads/Monalessa/EngSoftware/NotasDeAula-EngSw-EngComp-v2018.pdf>>. Citado 2 vezes nas páginas 15 e 16.
- DEVMEDIA, P. Como funcionam as aplicações web. S.l., 2012. Disponível em: <<https://www.devmedia.com.br/como-funcionam-as-aplicacoes-web/25888>>. Nenhuma citação no texto.
- DEVMEDIA, P. Java ee: entendendo a plataforma. S.l., 2021. Disponível em: <<https://www.devmedia.com.br/java-ee-entendendo-a-plataforma/30195>>. Citado na página 19.
- DUARTE, A. R. *Metodologia Rails: Análise da arquitetura Model View Controller Aplicada*. [S.l.], 2011. Citado na página 20.
- FARIA, T. Java ee 7 com jsf, primefaces e cdi. [S.l.: s.n.], 2ª edição, 2015. Citado na página 19.
- FOWLER, M. Patterns of enterprise application architecture. [S.l.]: Addison-Wesley Longman Publishing Co., Inc., 2002. Citado 3 vezes nas páginas 18, 21 e 29.
- ISTO. Número de usuários de internet no mundo chega a 4,66 bilhões. *Isto é Dinheiro*, São Paulo, n. 1222, 2021. Disponível em: <<https://www.istoedinheiro.com.br/numero-de-usuarios-de-internet-no-mundo-chega-aos-466-bilhoes/>>. Citado na página 18.
- KOTONYA, G.; SOMMERVILLE, I. Requirements engineering: processes and techniques. *Chichester: England*, John Wiley, 1998. Citado 3 vezes nas páginas 6, 16 e 17.
- MARTINS, B. F. S. Evolução do método frameweb para o projeto de sistemas de informação web utilizando uma abordagem dirigida a modelos. dissertação (mestrado). Universidade Federal do Espírito Santo, 2016. Citado na página 22.
- PFLEEGER, S. I. Engenharia de software: Teoria e prática. *São Paulo: Prentice Hall*, 2ª edição, 2004. Citado na página 16.
- ROBERTSON, S.; ROBERTSON, J. Mastering the requirements process. *2nd Edition*, Addison Wesley, 1998. Citado na página 16.
- SALVATORE, T. R. Alocaweb e biblattes - módulos do sistema marvin. monografia (projeto de graduação). Universidade Federal do Espírito Santo, 2016. Citado na página 21.
- SILVA, A. A. *C2D - Módulo de Credenciamento e Classificação de Docentes do Sistema Marvin*. Vitória, ES, Brasil, 2017. Citado na página 12.
- SOUZA, V. E. S. The FrameWeb Approach to Web Engineering: Past, Present and Future. In: ALMEIDA, J. P. A.; GUIZZARDI, G. (Ed.). *Engineering Ontologies and Ontologies for Engineering*. 1. ed. Vitória, ES, Brazil: NEMO, 2020. cap. 8, p. 100–124.

ISBN 9781393963035. Disponível em: <<http://purl.org/nemo/celebratingfalbo>>. Citado 6 vezes nas páginas 13, 14, 21, 22, 29 e 30.

Apêndices



Documento de Requisitos de Sistema

C2D – Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Registro de Alterações:

Versão	Responsável	Data	Alterações
1.0	Rafael Franco Ribeiro	13/08/2021	Versão Inicial
1.1	Rafael Franco Ribeiro	25/08/2021	Levantamento de requisitos
1.2	Rafael Franco Ribeiro	30/09/2021	Descrição dos requisitos
1.3	Rafael Franco Ribeiro	12/09/2021	Descrição dos caso de uso

Vitória, ES

2022

1 Introdução

Este documento apresenta a especificação dos requisitos do sistema *C2D – Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin*.

Esta especificação foi construída aplicando-se técnicas de levantamento de requisitos, bem como modelagem de casos de uso e de classes utilizando a linguagem UML.

A Seção 2 descreve os requisitos levantados junto aos *stakeholders*. A Seção 3 explica a divisão em subsistemas, descrevendo brevemente cada um deles. A Seção 4 apresenta o modelo de casos de uso, incluindo descrições de atores, os diagramas de casos de uso e suas respectivas descrições. A Seção 5 traz os modelos conceituais estruturais do sistema na forma de diagramas de classes. Por fim, a Seção 6 detalha o dicionário do projeto, contendo as definições das classes identificadas.

2 Definição de Requisitos

Esta seção descreve o resultado da atividade de levantamento de requisitos. A Subseção 2.1 resume de forma geral o propósito do sistema e a motivação para seu desenvolvimento. A Subseção 2.2 descreve o minimundo do sistema, apresentando superficialmente suas principais características. A Subseção 2.3 lista os requisitos de usuário do sistema, na forma de histórias de usuário e requisitos não-funcionais.

2.1 Descrição do Propósito do Sistema

Esse sistema tem como propósito facilitar e otimizar a atividade de classificação e credenciamento de docentes do Programa de Pós-Graduação em Informática (PPGI) da Universidade Federal do Espírito Santo (UFES) pelo seus trabalhos produzidos.

Os trabalhos produzidos são avaliados com base no veículo de sua publicação gerando uma pontuação para ele de acordo com os critérios estabelecidos pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), portanto o sistema deve receber essas informações para que ele possa gerar as pontuações dos(as) docentes.

2.2 Descrição do Minimundo

O PPGI possui a função de classificar e credenciar seu docentes conforme suas produções de trabalhos científicos e tecnológicos. Tais produções são avaliadas a partir dos veículos nas quais foram publicadas, segundo os critérios definidos pela CAPES. Assim, os(as) docentes que estão credenciados podem lecionar disciplinas, orientar estudantes, usar recursos financeiros do PPGI, etc.

O PPGI definiu 6 normas referentes ao Credenciamento e Recredenciamento de Docentes,¹ a saber: (a) ingresso no programa; (b) recredenciamento no programa; (c) credenciamento como orientador no doutorado; (d) requisitos mínimos de publicação; (e) critérios de (re)credenciamento para membros em licença maternidade; (f) categorização de docentes. Essas normas possuem vários critérios para suas análises como: orientação de alunos, vínculo do docentes com a instituição de ensino, publicação de trabalhos produzidos, atividade administrativas que desempenha, dentre outros.

Os requisitos do módulo C2D serão apenas aqueles que tem relação direta com a avaliação de publicações, por serem mais trabalhosos de se lidar manualmente e passíveis de automação por um sistema de informação. Portanto o sistema atenderá apenas os

¹ <<https://informatica.ufes.br/pt-br/pos-graduacao/PPGI/credenciamento-de-docentes>>

requisitos das normas (a), (b) e (c) que façam avaliação das publicações dos(as) docentes.

2.2.1 Cálculo da pontuação

A norma (d) define as regras de pontuações mínimas de publicação para que o(a) docente possa ingressar ou serem reconhecidos na PPGI. Atualmente as regras definidas são:

1. As pontuações são concedidas de acordo com os estratos do Qualis, definido pela CAPES no documento de área da Computação, conforme esquema de pontos que estiver vigente em um dado momento. Por exemplo, atualmente temos: A1 = 1; A2 = 0,85; B1 = 0,7; B2 = 0,5; B3 = 0,2; B4 = 0,1; B5 = 0,05. Os níveis A1, A2 e B1 são conhecidos como “estrato restrito”;²
2. Em caso de co-autoria, todos os(as) docentes pontuam o valor cheio associado ao estrato Qualis do veículo da publicação, ou seja, não há divisão de pontos;
3. Seja o(a) docente um orientador de Mestrado, ele deve pontuar, no biênio, um total de 1,6 pontos, sendo 0,4 em periódico (qualquer estrato) e 0,7 no estrato A1-B1 (periódico ou conferência);
4. Seja o(a) docente um orientador de Doutorado, ele deve pontuar, no quadriênio, 3,2 pontos, sendo 0,7 em periódico do estrato A1-B1 e 1,4 no estrato A1-B1 (periódico ou conferência).

2.2.2 Ingresso no programa

Existem dois critérios para o ingresso do(a) docente no programa, são eles:

1. Ter plano de trabalho aprovado pelo colegiado, considerando alinhamento com as linhas de pesquisa e estrutura curricular do programa;
2. Atender aos requisitos de reconhecimentos do programa (definidos na Seção 2.2.3), considerando-se os trabalhos publicados no período (biênio/quadriênio) imediatamente anterior à solicitação, porém utilizando os requisitos mínimos de publicação do ano seguinte à solicitação.

Para o ingresso de docentes no módulo C2D, será considerado que o primeiro

² Recentemente a CAPES modificou o Qualis, que atualmente é composto de 8 níveis de classificação (A1–A4, B1–B4, sendo A1–A4 estrato restrito), além do nível C (não classificado). No entanto, para este documento nos pautamos nos níveis indicados nas normas do PPGI, conforme divulgado em seu site, já citado anteriormente.

critério já tenha sido atendido, ou seja o colegiado já aprovou o plano de trabalho do(a) docente.

2.2.3 Regras para o Recredenciamento

No começo do ano os membros dos PPGI são avaliados para que possam ser recredenciados. Então o C2D fará essa validação seguindo os seguintes critérios estabelecidos pelo PPGI, para que possa determinar se o(a) docente deve ser recredenciado ou descredenciado. Os critérios são:

1. Produção: o(a) docente deve atender aos requisitos mínimos de publicação para permanência no programa considerando-se os trabalhos publicados no período anterior; OU ser bolsista de produtividade do CNPq; OU ser o atual coordenador do PPGI ou ter sido coordenador do PPGI por pelo menos 6 meses no ano anterior;
2. Orientação: o(a) docente deve ter tido ao menos 4 alunos orientados naquele ano, do contrário ter ofertado ao menos 2 vagas no ano;
3. Docência: professores permanentes devem ofertar ao menos 1 disciplina por ano no PPGI, colaboradores 1 a cada 2 anos, ficando isentos(as) docentes que tiveram qualquer tipo de licença ou afastamento de longo prazo ao longo do ano, como pós-doc e capacitação;
4. Administrativo: o(a) docente deve participar do Colegiado do PPGI ou de alguma de suas comissões, ficando isentos(as) docentes que são chefes de departamento, coordenadores de curso ou de PET, além dos 1/3 docentes mais produtivos do último quadriênio.

Se o(a) docente não conseguir atender a nenhum desses requisitos e não possuir orientações ativas ele será descredenciado do programa. Caso ainda possua alunos orientados, as seguintes regras serão aplicadas:

1. O professor não será descredenciado;
2. O professor será impedido de oferecer disciplinas até regularizar seu recredenciamento;
3. O professor será impedido de assumir novas orientações até regularizar seu recredenciamento;
4. Para cada aluno que possivelmente não concluirá o curso no ano corrente, o professor deve estabelecer co-orientação com outro professor do programa;
5. O professor não terá suas publicações registradas na Plataforma Sucupira a menos que consiga regularizar seu recredenciamento;

6. Para regularizar seu credenciamento, o professor deverá comprovar, em qualquer momento do ano, que atingiu os critérios de credenciamento do ano seguinte;
7. No caso de não atingir os critérios de credenciamento até o final do ano, o professor será descredenciado no ano seguinte e deverá passar as orientações dos alunos que ainda não concluíram ao co-orientador escolhido, conforme item 4, acima.

2.2.4 Fluxo de trabalho do C2D

O módulo C2D deverá funcionar de maneira mais automatizada possível, para isso ele deve registrar quem são os membros do PPG a partir dos cadastros básicos do Marvin, bem como inserir em lote as publicações dos(as) docentes e a avaliação Qualis de cada veículo a partir de arquivos extraídos de outros sistemas. Além disso, deve permitir a atualização de seus dados sempre que for necessário. Desse modo o C2D trabalhará da seguinte forma:

1. Em primeiro lugar, é preciso que exista o cadastro do programa de pós-graduação (PPG), com indicação do(a) coordenador(a), subcoordenador(a) e secretário(a), usuários pré-cadastrados no Marvin (módulo núcleo) que terão acesso às funcionalidades do C2D em relação ao PPG em questão (para simplificar, na sequência faremos referência apenas ao/à secretário(a));
2. O(A) secretário(a) deve indicar as pontuações, conforme normas do programa. Para isso ele(a) deve cadastrar os diferentes níveis de classificação definidos no Qualis (A1, A2, B1, B2, C1, etc.) e então informar, para cada nível, um nome e sua pontuação para cada tipo de veículo (periódico e conferência). Deve-se indicar, também, quais níveis fazem parte do estrato restrito;
3. O C2D precisa saber como classificar cada publicação de acordo com o Qualis. Dessa forma, o secretário(a) envia arquivos em formatos padronizados ao sistema para o C2D extrair as informações de classificação Qualis dos veículos de publicação (conferências e periódicos);
4. Para que possa verificar os requisitos mínimos de publicação, o(a) secretário(a) deve primeiramente cadastrar os(as) docentes do programa a partir dos usuários já cadastrados no Marvin, indicando o tipo (atualmente orientadores de mestrado e doutorado);
5. Após isso ele(a) poderá inserir as regras de pontuação no sistema, informando nome, a que tipo de docentes ela se refere, uma data de início e fim da vigência, a que tipos de docentes ela se refere, um nome, o período em que as publicações serão avaliadas (em número de anos passados) e a quantidade de pontos que deve ser alcançada em

quatro categorias: pontuação total, pontuação em periódicos, pontuação no estrato restrito, pontuação em periódicos no estrato restrito

6. O(A) secretário(a) deve então recuperar as publicações dos(as) docentes a partir de seus Currículos Lattes. O C2D tentará associar cada publicação extraída aos veículos cadastrados anteriormente;
7. Com todos esse dados, o(a) secretário(a) pode então solicitar o cálculo da pontuação dos(as) docentes para o C2D para que ele informe se o(a) docente atendeu as requisitos de publicação.

As pontuações dos níveis Qualis, a classificação dos veículos nos níveis e as regras de requisitos mínimos de publicação devem estar associadas a datas de início e fim de vigência, mantendo, assim, um histórico das configurações anteriores, bem como a configuração vigente do programa.

2.3 Requisitos de Usuário

Tomando por base o contexto do sistema descrito na Seção 2.2 e considerando como principais *stakeholders* os(as) coordenadores(as), o(a) secretário(a) e os demais membros do programa, foram identificadas estórias de usuário e requisitos não-funcionais.

As estórias de usuário são apresentadas na Tabela 1 e os requisitos não-funcionais globais (ou seja, aqueles que não são caracterizados como critérios de aceitação de estórias de usuário específicas) na Tabela 2.

Tabela 1 – Estórias de Usuário.

ID:	US-1	Depende:		Prioridade:	Alta
Descrição:	Como administrador(a), quero cadastrar o PPG e indicar seus/suas coordenadores(as) e secretários(as), para que possam usar o C2D.				

Critérios de Aceitação:	<ul style="list-style-type: none"> – CA1: Para cadastrar o PPG, é necessário indicar nome, sigla, coordenadores(as) e secretários(as); – CA2: O(A) administrador(a) pode buscar os(as) acadêmicos(as) para indicar como coordenadores(as) e secretários(as) por meio do nome ou e-mail; – CA3: Um PPG deve ter ao menos um(a) coordenador(a); – CA4: O(A) administrador(a) pode consultar e editar PPGs; – CA5: Um PPG não pode ser excluído se estiver relacionado a outros dados do C2D. <p>Obs.: o cadastro de acadêmicos e a autenticação de usuários é responsabilidade do módulo <i>núcleo</i> do Marvin. O C2D apenas verifica se o(a) acadêmico(a) é coordenador(a) ou secretário(a) de um PPG para oferecer as demais funcionalidades. Nas demais estórias, onde lê-se “como secretário(a)”, leia-se “como secretário(a) ou coordenador(a)” e considere que o acesso é sempre aos dados do próprio PPG.</p>
--------------------------------	---

ID:	US-2	Depende:	US-1	Prioridade:	Alta
Descrição:	Como secretário(a), quero cadastrar os níveis de classificação das publicações, para fazer os cálculos de acordo com a classificação mais recente do Qualis.				
Critérios de Aceitação:	<ul style="list-style-type: none"> – CA1: Para cadastrar os níveis de classificação, é necessário indicar uma data de início e fim da vigência e um conjunto de níveis de classificação, cada qual com um nome, sua pontuação para conferência e para periódicos, e se aquele nível faz parte do estrato restrito (sim/não); – CA2: Se a data de fim da vigência não for informada, essa classificação será considerada vigente; – CA3: Pode existir somente uma classificação vigente; – CA4: A consulta da classificação pode ser feita pela data de vigência; – CA5: A classificação não pode ser excluída se tiver uma regra associada a ela. 				

ID:	US-3	Depende:	US-2	Prioridade:	Alta
------------	------	-----------------	------	--------------------	------

Descrição:	Como secretário(a), quero inserir em lote as classificações do Qualis dos veículos de publicação, para que possa ter a relação entre o trabalho produzido e o nível de classificação que ele se encontra dentro do sistema.
Critérios de Aceitação:	<ul style="list-style-type: none"> – CA1: Todo envio em lote deve informar uma data de início e fim da vigência; – CA2: Para que essa relação seja criada para o caso dos periódicos deve ser informado o ISSN, o nome e a classificação do Qualis de cada periódico; – CA3: Para o que essa relação seja criada para o caso das conferências devem ser informados a sigla, o nome e a classificação do Qualis de cada conferência; – CA4: No caso dos periódicos o ISSN deve ser único; – CA5: No caso das conferências o nome deve ser único.

ID:	US-4	Depende:	US-1	Prioridade:	Alta
Descrição:	Como secretário(a), quero indicar os(as) docentes do programa, para que possam ser avaliados dentro do sistema.				
Critérios de Aceitação:	<ul style="list-style-type: none"> – CA1: O(A) secretário(a) pode buscar os(as) acadêmicos(as) para indicar como docentes do programa por meio do nome ou e-mail; – CA2: Para cada docente indicado(a), deve-se especificar se orienta no Mestrado e se orienta no Doutorado; – CA3: O sistema deve exibir a lista de docentes do programa; – CA4: Docentes podem ser desvinculados(as) do programa livremente. 				

ID:	US-5	Depende:	US-1	Prioridade:	Média
Descrição:	Como secretário(a), quero inserir as regras de classificação mínimas de publicação, para que possa ter as regras mais recentes de publicação para avaliação dos(as) docentes.				

Critérios de Aceitação:	<ul style="list-style-type: none"> – CA1: Para cadastrar uma nova regra, é necessário informar uma data de início e fim da vigência, a que tipos de docentes ela se refere, o período em que as publicações serão avaliadas (em número de anos passados) e a quantidade de pontos que deve ser alcançada em quatro categorias: pontuação total, pontuação em periódicos, pontuação no estrato restrito, pontuação em periódicos no estrato restrito; – CA2: Se a data de fim da vigência não for informada, a regra será considerada em vigência; – CA3: Dentro de um mesmo período de vigência, só deve haver uma regra por tipo de docente; – CA4: As quantidades de pontos indicadas devem ser positivas; – CA5: O sistema deve mostrar as regras vigentes; – CA6: O sistema deve permitir busca por regras antigas por data de vigência; – CA7: Pode-se editar regras vigentes apenas; – CA8: Pode-se excluir quaisquer regras.
--------------------------------	---

ID:	US-6	Depende:	US-4	Prioridade:	Alta
Descrição:	Como secretário(a), quero inserir em lote os dados referentes às publicações dos(as) docentes que serão avaliados, para que seja possível avaliar os(as) docentes no requisito de publicação.				
Critérios de Aceitação:	<ul style="list-style-type: none"> – CA1: Para inserção em lote, deve-se enviar um arquivo referente a um Currículo Lattes e indicar a qual docente do programa ele pertence; – CA2: Para cadastrar uma publicação deve ser possível extrair do currículo obrigatoriamente o título, ISSN (apenas para periódicos), nome do evento (apenas para conferência) e o ano da publicação; – CA3: Como informação adicional é possível armazenar o DOI, intervalo de páginas, nome da editora, número da edição, série, volume. 				

ID:	US-7	Depende:	US-6, US-5	Prioridade:	Alta
------------	------	-----------------	------------	--------------------	------

Descrição:	Como secretário(a), quero que o sistema classifique os(as) docentes quanto às suas publicações, para que eu possa saber quais atendem aos requisitos mínimos de publicação.
Critérios de Aceitação:	– CA1: O resultados deve ser apresentado com as seguintes informações: nome do docente, pontuação nas quatro categorias (total, em periódicos, no estrato restrito, em periódicos no estrato restrito) e, para cada regra de classificação, se ele atende ou não os requisitos mínimos.

Tabela 2 – Requisitos Não Funcionais.

ID	Descrição	Categoria	Prioridade
RNF-1	O sistema deve ter uma interface intuitiva, para que o usuário tenha facilidade de usar o sistema.	Usabilidade	Alta
RNF-2	O sistema deve ser compatível com os principais navegadores disponíveis no mercado.	Portabilidade	Média
RNF-3	O desenvolvimento do sistema deve utilizar o conceito de reutilização de componentes.	Reusabilidade	Alta
RNF-4	O sistema deve ter um controle de acesso.	Segurança	Alta
RNF-5	O sistema deve ser construído de maneira que seja de fácil manutenção, para que possa ter novas funcionalidades no futuro.	Manutenção	Alta

3 Identificação de Subsistemas

Conforme descrito nos requisitos elencados na Seção 2, o módulo C2D será integrado ao Marvin a fim de utilizar algumas das funcionalidade já desenvolvidas nele. Os requisitos de cadastro de docentes e autenticação são providos pelo módulo núcleo do Marvin (desenvolvido e documentado separadamente) e, portanto, há uma dependência do C2D com este módulo, conforme ilustra a Figura 1.

Visual Paradigm Online Free Edition



Visual Paradigm Online Free Edition

Figura 1 – Diagrama de Pacotes e os Subsistemas Identificados.

4 Modelo de Casos de Uso

O modelo de casos de uso corresponde a uma tentativa de descrever a relação das funcionalidades do sistema com cada um de seus atores. Os atores identificados no contexto deste projeto estão descritos na Tabela 3.

Tabela 3 – Descrição dos atores envolvidos nos casos de uso.

Ator	Descrição
Secretário(a)	Responsável por manusear o sistema. Inclui o(a) coordenador(a) e o(a) subcoordenador(a), que podem substituir o(a) secretário(a) na ausência deste(a).
Administrador(a)	Responsável por cadastrar os PPGs e indicar o(a) coordenador(a) e o(a) secretário(a).

A seguir, são apresentados os diagramas de casos de uso e descrições associadas. A Figura 2 apresenta o diagrama de casos de uso do módulo C2D.

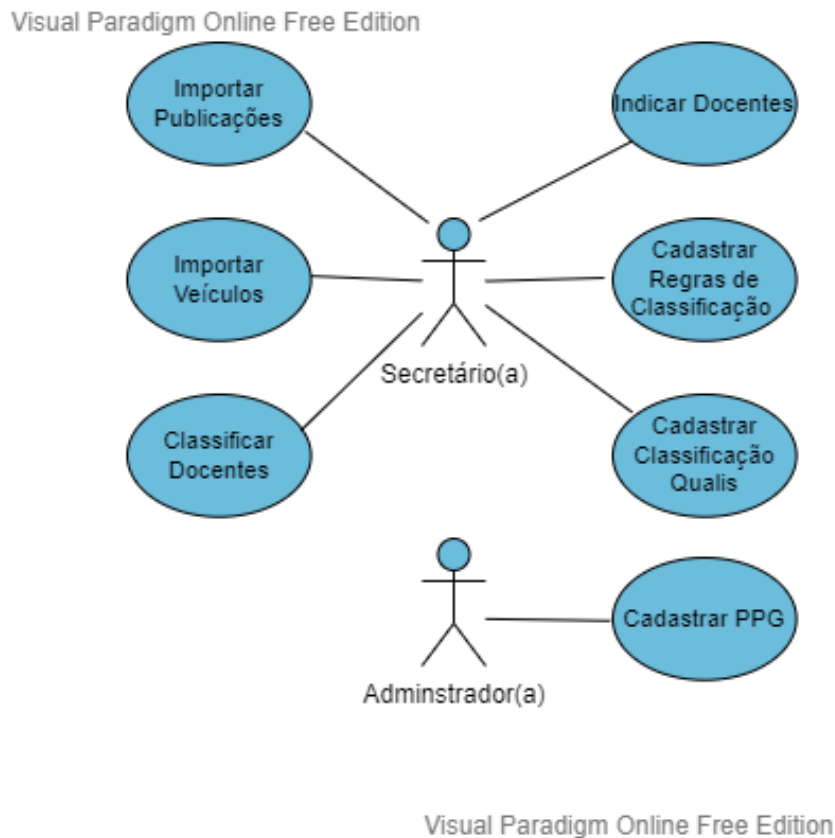


Figura 2 – Diagrama de Casos de Uso do sistema.

A seguir, são apresentadas as descrições de cada um dos casos de uso identificados

dentro do C2D. Os casos de uso cadastrais de baixa complexidade, envolvendo inclusão, consulta, alteração e exclusão (CRUDs), são descritos na Tabela 4.

Tabela 4 – Caso de uso cadastrais do sistema.

Id	Nome	Ações	Requisitos	Classes
UC-1	Cadastrar PPG	(I) Não pode haver duplicidade no atributo nome, um acadêmico só pode ser indicado para um cargo (coordenador(a) ou secretário(a)); (C) Sem restrições; (A) Não pode haver duplicidade no atributo nome, um acadêmico só pode ser indicado para um cargo (coordenador(a) ou secretário(a)); (E) Não pode ser excluído se estiver relacionado com outros dados do C2D.	US-1	PPG e Academic
UC-2	Cadastrar classificação da Qualis	(I) Não pode haver duplicidade no atributo nome; (C) Sem restrições; (A) Não disponível; (E) Sem restrições.	US-2	Qualis, Qualis Vigência e PPG
UC-3	Cadastrar regras de classificação	(I) As pontuações informadas devem ser positivas; (C) Sem restrições; (A) Sem restrições; (E) Sem restrições.	US-5	Regra e PPG

Os casos de uso deste subsistema que não se encaixam na categoria acima são descritos nas páginas subsequentes.

Descrição de Caso de Uso

Projeto: C2D – Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Identificador: UC-4

Nome: Importar Veículos

Descrição: Criar relação entre a classificação da Qualis com os veículos de publicação.

Tabela 5 – Fluxos de eventos normais para o caso de uso 05.

Cenário	Precondição	Descrição
Cenário 01	CSV tem que estar no padrão de importação de Veículos	<ol style="list-style-type: none"> 1. O secretário(a) importar o CSV com os dados dos veículos para o sistema; 2. O sistema mostra uma tabela com dados extraídos do CSV; 3. O secretário(a) aceita a inserção dessa tabela para o sistema; 4. O sistema salvo os dados no banco.
Cenário 02	CSV tem que estar no padrão de importação de Veículos	<ol style="list-style-type: none"> 1. O secretário(a) importar o CSV com os dados dos veículos para o sistema; 2. O sistema mostra uma tabela com dados extraídos do CSV; 3. O secretário(a) recusa a inserção dessa tabela para o sistema; 4. O sistema volta para tela de importação de veículos.

Fluxos variantes:

- No **cenário 01** ou **cenário 02**, no caso de importação conferência com nome já inserido, o sistema deve atualizar os dados desse registro se o secretário(a) confirmar a inserção;
- No **cenário 01** ou **cenário 02**, no caso de importação periódicos com ISSN já inserido, o sistema deve atualizar os dados desse registro se o secretário(a) confirmar a inserção.

Requisitos relacionados: [US-3](#)

Classes relacionadas: PPG, Veículo, Qualis e Qualis Vigência.

Projeto: C2D – Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Identificador: UC-5

Nome: Indicar Docentes

Descrição: Indicar docentes que serão avaliados dentro do sistema.

Tabela 6 – Fluxos de eventos normais para o caso de uso 06.

Cenário	Precondição	Descrição
Cenário 01	Docentes estar cadastrado no sistema	<ol style="list-style-type: none"> 1. O secretário(a) escolhe o docente para ser avaliado; 2. O sistema mostra uma modal para escolher se o docente será orientador de Mestrado ou Doutorado; 3. O secretário(a) escolhe o tipo de orientador; 4. O sistema salva a alteração no banco de dados.
Cenário 02	Docentes estar cadastrado no sistema	<ol style="list-style-type: none"> 1. O secretário(a) escolhe o docente para ser desvinculado ao sistema de avaliação; 2. O sistema exibe uma modal de confirmação da alteração; 3. O secretário(a) confirma a alteração; 4. O sistema salva a alteração no banco de dados.

Requisitos relacionados: [US-4](#)

Classes relacionadas: PPG e Academic.

Projeto: C2D – Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Identificador: UC-6

Nome: Importar Publicações

Descrição: Importar publicações dos docentes.

Tabela 7 – Fluxos de eventos normais para o caso de uso 05.

Cenário	Precondição	Descrição
Cenário 01	O docente deve estar cadastrado no sistema com o seu id do lattes	<ol style="list-style-type: none"> 1. O secretário(a) carrega o XML do currículo do lattes do docente para o sistema; 2. O sistema mostra uma tabela com dados extraídos do XML; 3. O secretário(a) aceita a inserção dessa tabela para o sistema; 4. O sistema salva as publicações.
Cenário 02	O docente deve estar cadastrado no sistema com o seu id do lattes	<ol style="list-style-type: none"> 1. O secretário(a) carrega o XML do currículo do lattes do docente para o sistema; 2. O sistema mostra uma tabela com dados extraídos do XML; 3. O secretário(a) recusa a inserção dessa tabela para o sistema; 4. O sistema volta para tela de importação de publicações.

Requisitos relacionados: [US-6](#)

Classes relacionadas: Academic, Publicação.

Projeto: C2D – Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Identificador: UC-7

Nome: Classificar Docentes

Descrição: Avaliar os docentes do PPG de acordo as regra de classificação vigentes.

Tabela 8 – Fluxos de eventos normais para o caso de uso 05.

Cenário	Precondição	Descrição
Cenário 01	As regras de classificação do PPG devem estar salvas no sistema	<ol style="list-style-type: none"> 1. O secretário(a) solicita avaliação do docentes; 2. O sistema filtra a lista de docentes que serão avaliados; 3. O sistema adquire as publicações que podem ser avaliadas de acordo as regras de vigência atuais; 4. O sistema calcula as pontuações do docente de acordo com suas publicações; 5. O sistema mostra pontuação do docentes junto com as regras que ele foi aceito.

Fluxos variantes:

- No **cenário 01**, no caso do docente não se enquadrar em uma regra, o sistema deve indicar que o docente não foi avaliado por essa regra quando for exibir o resultado da avaliação;

Requisitos relacionados: [US-7](#)

Classes relacionadas: PPG, Veículo, Qualis, Qualis Vigência, Academic, Regra e Publicação.

5 Modelo Estrutural

O modelo conceitual estrutural visa capturar e descrever as informações (classes, associações e atributos) que o sistema deve representar para prover as funcionalidades descritas nos casos de uso especificados na Seção 4.

A Figura 3 apresenta o diagrama de classes do módulo C2D. Na Seção 6 — Dicionário de Projeto — são apresentadas as descrições das classes, atributos e operações presentes no diagrama.

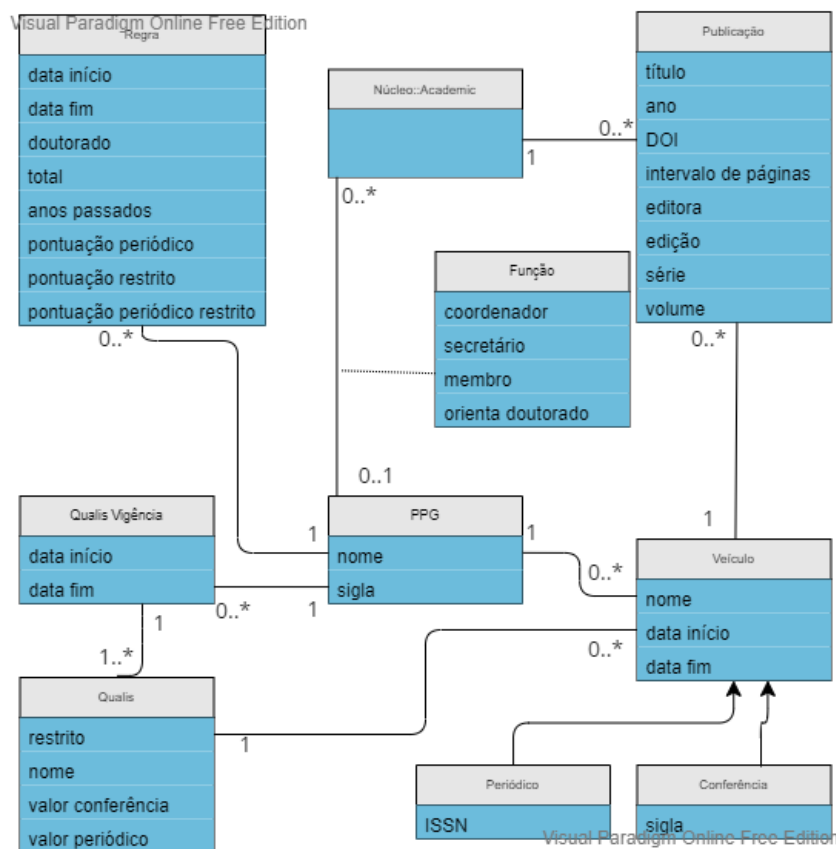


Figura 3 – Diagrama de classes do módulo C2D.

6 Dicionário de Projeto

Esta seção apresenta as definições detalhadas das classes, descrevendo seus atributos e associações e servindo como um glossário do projeto. As definições são organizadas por subsistema, cada classe sendo apresentada em uma tabela separada. A coluna “Obr.?” indica com um “x” se o atributo é obrigatório (deve possuir um valor para se criar um objeto da classe).

Tabela 9 – Detalhamento da classe *PPG*.

Propriedade	Tipo	Obr.?	Descrição
nome	String	x	Nome único do PPG.
sigla	String		Sigla do PPG.

Tabela 10 – Detalhamento da classe *Função*.

Propriedade	Tipo	Obr.?	Descrição
coordenador	Booleano	x	Indica se é um coordenador de PPG.
secretário	Booleano	x	Indica se é um secretário de PPG.
membro	Booleano	x	Indica se é um membro de PPG.
orienta doutorado	Booleano	x	Indica se é um orientador de doutorado.

Tabela 11 – Detalhamento da classe *Regra*.

Propriedade	Tipo	Obr.?	Descrição
data início	Data	x	Data de começo da vigência.
data fim	Data		Data de fim da vigência.
doutorado	Booleano	x	Indica se a regra é para orientadores de doutorado.
total	Real	x	Valor total que deve ser atingido.
anos passados	Inteiro	x	Quantidade de anos anteriores que serão avaliados.
pontuação periódico	Real		Valor que deve ser atingido pelo periódico.
pontuação periódico restrito	Real		Valor que deve ser atingido pelo periódico restrito.
pontuação restrito	Real		Valor total que deve ser atingido pelo restrito.

Tabela 12 – Detalhamento da classe *Publicação*.

Propriedade	Tipo	Obr.?	Descrição
título	String	x	Título da publicação.
ano	String	x	Ano da publicação.
DOI	String		Identificador de Objeto Digital.
intervalo de páginas	String		Intervalo páginas.
editora	String		Nome da editora.
edição	String		Número da edição.
série	String		Séria da publicação.
volume	String		Volume da publicação.

Tabela 13 – Detalhamento da classe *Veículo*.

Propriedade	Tipo	Obr.?	Descrição
nome	String	x	Nome da Veículo.
data início	Data	x	Data de começo da vigência.
data fim	Data		Data de fim da vigência.

Tabela 14 – Detalhamento da classe *Periódico*.

Propriedade	Tipo	Obr.?	Descrição
ISSN	String	x	Número Internacional Normalizado.

Tabela 15 – Detalhamento da classe *Conferência*.

Propriedade	Tipo	Obr.?	Descrição
sigla	String	x	Sigla da conferência.

Tabela 16 – Detalhamento da classe *Qualis*.

Propriedade	Tipo	Obr.?	Descrição
nome	String	x	Nome da Qualis.
valor periódico	Real	x	Valor da classificação Qualis para periódicos.
valor conferência	Real	x	Valor da classificação Qualis para conferência.
restrito	Booleano	x	Indica se é um estrato restrito.

Tabela 17 – Detalhamento da classe *Qualis Vigência*.

Propriedade	Tipo	Obr.?	Descrição
data início	Data	x	Data de começo da vigência.
data fim	Data		Data de fim da vigência.



Documento de Projeto de Sistema

C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin

Registro de Alterações:

Versão	Responsável	Data	Alterações
1.0	Rafael Franco Ribeiro	13/03/2022	Versão Inicial
1.1	Rafael Franco Ribeiro	16/03/2022	Finalizando Capítulo 5

Vitória, ES

2022

1 Introdução

Este documento apresenta o projeto (*design*) do sistema *C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin*. O módulo C2D é um sistema Web construído para auxiliar os secretários da PPG que avaliam anualmente os docentes que fazem parte do seu PPG para credenciá-los e classificá-los. Existem várias regras de avaliação, nesse módulo foi abordado apenas o critério de produção de publicações de artigos científicos e tecnológicos.

Além desta introdução, este documento está organizado da seguinte forma: a Seção 2 apresenta a plataforma de software utilizada na implementação do sistema; a Seção 3 apresenta a especificação dos requisitos não funcionais (atributos de qualidade), definindo as táticas e o tratamento a serem dados aos atributos de qualidade considerados condutores da arquitetura; a Seção 4 apresenta a arquitetura de software; por fim, a Seção 5 apresenta os modelos FrameWeb que descrevem os componentes da arquitetura.

2 Plataforma de Desenvolvimento

Na Tabela 1 são listadas as tecnologias utilizadas no desenvolvimento da ferramenta, bem como o propósito de sua utilização.

Tabela 1 – Plataforma de Desenvolvimento e Tecnologias Utilizadas.

Tecnologia	Versão	Descrição	Propósito
Java EE	7	Conjunto de especificação de APIs e tecnologias, que são implementadas por programas servidores de aplicação.	Redução da complexidade do desenvolvimento, implantação e gerenciamento de aplicações Web a partir de seus componentes de infra-estrutura prontos para o uso.
Java	13	Linguagem de programação orientada a objetos e independente de plataforma.	Escrita do código-fonte das classes que compõem o sistema.
JSF	2.3	API para a construção de interfaces de usuários baseada em componentes para aplicações Web	Criação das páginas Web e sua comunicação com as classes Java.
EJB	4.0.9	API para construção de componentes transacionais gerenciados por <i>container</i> .	Implementação das regras de negócio em componentes distribuídos, transacionais, seguros e portáveis.
JPA	2.2	API para persistência de dados por meio de mapeamento objeto/-relacional.	Persistência dos objetos de domínio sem necessidade de escrita dos comandos SQL.

Tecnologia	Versão	Descrição	Propósito
CDI	2.0	API para injeção de dependências.	Integração das diferentes camadas da arquitetura.
Facelets	2.0	API para definição de decoradores (<i>templates</i>) integrada ao JSF.	Reutilização da estrutura visual comum às páginas, facilitando a manutenção do padrão visual do sistema.
PrimeFaces	8.0	Conjunto de componentes visuais JSF <i>open source</i> .	Reutilização de componentes visuais Web de alto nível.
MySQL Server	8.0	Sistema Gerenciador de Banco de Dados Relacional gratuito.	Armazenamento dos dados manipulados pela ferramenta.
WildFly	19	Servidor de Aplicações para Java EE.	Fornecimento de implementação das APIs citadas acima e hospedagem da aplicação Web, dando acesso aos usuários via HTTP.

Na Tabela 2 vemos os softwares que apoiaram o desenvolvimento de documentos e também do código fonte.

Tabela 2 – Softwares de Apoio ao Desenvolvimento do Projeto

Tecnologia	Versão	Descrição	Propósito
TeX Live	2018	Implementação do $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$	Documentação do projeto arquitetural do sistema.
TeXstudio	3.1.1	Editor de LaTeX.	Escrita da documentação do sistema, sendo usado o <i>template abnTeX</i> . ¹
Eclipse Java EE IDE for Web Developers	4.8	Ambiente de desenvolvimento (IDE) com suporte ao desenvolvimento Java EE.	Implementação, implantação e testes da aplicação Web Java EE.
Apache Maven	3.5	Ferramenta de gerência/construção de projetos de software.	Obtenção e integração das dependências do projeto.

3 Requisitos Não Funcionais

A Tabela 3 apresenta a especificação dos requisitos não funcionais identificados no Documento de Especificação de Requisitos.

Tabela 3 – Especificação de Requisitos Não Funcionais.

RNF-1 – O sistema deve ter uma interface intuitiva.	
Categoria:	Usabilidade
Tática / Tratamento:	Seleção de dados para edição utilizar o mínimo possível do teclado e utilizar símbolos que ajudem na identificação de funcionalidades.

¹ <<http://www.abntex.net.br>>.

Medida:	Realizar pesquisa sobre a usabilidade do sistema com possíveis usuários que utilizaram o novo módulo. Deverá ser utilizado cinco classificações (muito ruim, ruim, normal, bom , muito bom).
Critério de Aceitação:	A média de classificação deve ficar entre bom e muito bom.

RNF-2 – O sistema deve com os principais navegadores do mercado.	
Categoria:	Portabilidade
Tática / Tratamento:	Utilização da biblioteca PrimeFaces para construção das páginas Web.
Medida:	Visualizar a execução do sistema e suas funcionalidades nos seguintes navegadores: Google Chrome, Firefox e Microsoft Edge.
Critério de Aceitação:	Todas as funcionalidades devem conseguir ser executadas no três navegadores.

RNF-3 – O sistema deve ter controle de acesso.	
Categoria:	Segurança
Tática / Tratamento:	Utilizar um framework para gerenciar autenticação e autorizações que siga as boas práticas de segurança. Dedicar um tempo para fazer as adaptação de configurações para ser integrado ao sistema.
Medida:	Fazer teste acesso a módulos com usuários de diferentes papéis durante o processo de desenvolvimento software e antes entrega final do módulo.
Critério de Aceitação:	Usuários que não tem permissão de acesso serão redirecionados.

RNF-4 – O sistema deve ser de fácil manutenção.	
Categoria:	Manutenibilidade
Tática / Tratamento:	Utilização da arquitetura MVC para construção do projeto.
Medida:	Quantidade de bugs e monitoramento do modelo MVC.
Critério de Aceitação:	Inexistência de bugs na entrega do projeto.

4 Arquitetura de Software

A Figura 1 mostra a arquitetura do sistema *C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin*.

O sistema C2D possui como objetivo ser integrado ao Marvin. Desse modo, para facilitar a integração e possíveis melhorias das suas funcionalidades no futuro, foi utilizado o FrameWeb (SOUZA, 2020), que propõe uma arquitetura de software na qual o Marvin foi desenvolvido. Tal arquitetura é baseada no padrão arquitetural Camada de Serviço (*Service Layer*) (FOWLER, 2002). Desse modo, qualquer aplicação que for desenvolvida

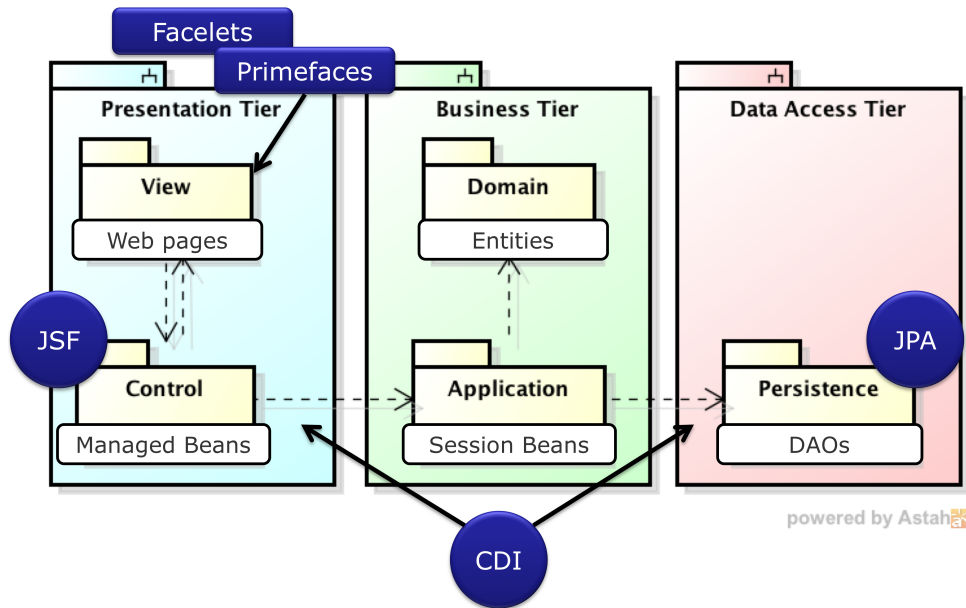


Figura 1 – Arquitetura de Software.

sobre seu modelo deve ter as seguintes camadas:

- **Camada de Negócio (*Business Tier*):** atua diretamente com as regras de negócio, provendo as funcionalidades para que as mesmas funcionem dentro do sistema. Esta camada é dividida em duas: Lógica de Domínio (*Domain*) e Lógica de Aplicação (*Application*);
- **Camada de Apresentação (*Presentation Tier*):** lida com as funcionalidades necessárias para criação de interface para que o usuário consiga interagir com o sistema. Esta camada é dividida em duas: Visão (*View*) e Controle (*Control*);
- **Camada de Acesso a Dados (*Data Access Tier*):** trata das funcionalidades relacionadas à persistência de dados.

5 Modelagem FrameWeb

C2D: Módulo de Credenciamento e Recredenciamento de Docentes do Sistema Marvin é um sistema Web cuja arquitetura utiliza *frameworks* comuns no desenvolvimento para esta plataforma. Desta forma, o sistema pode ser modelado utilizando a abordagem FrameWeb (SOUZA, 2020).

A Tabela 4 indica os *frameworks* presentes na arquitetura do sistema que se encaixam em cada uma das categorias de *frameworks* que FrameWeb dá suporte. Em

seguida, os modelos FrameWeb são apresentados para cada camada da arquitetura.

Tabela 4 – *Frameworks* da arquitetura do sistema separados por categoria.

Categoria de <i>Framework</i>	<i>Framework</i> Utilizado
Controlador Frontal	JSF
Injeção de Dependências	CDI
Mapeamento Objeto/Relacional	JPA
Segurança	JAAS

5.1 Camada de Negócio

A camada de Negócio é dividida em duas: Lógica de Domínio (Domain) e Lógica de Aplicação (Application). A Figura 2 representa as entidades da lógica de domínio desse projeto.

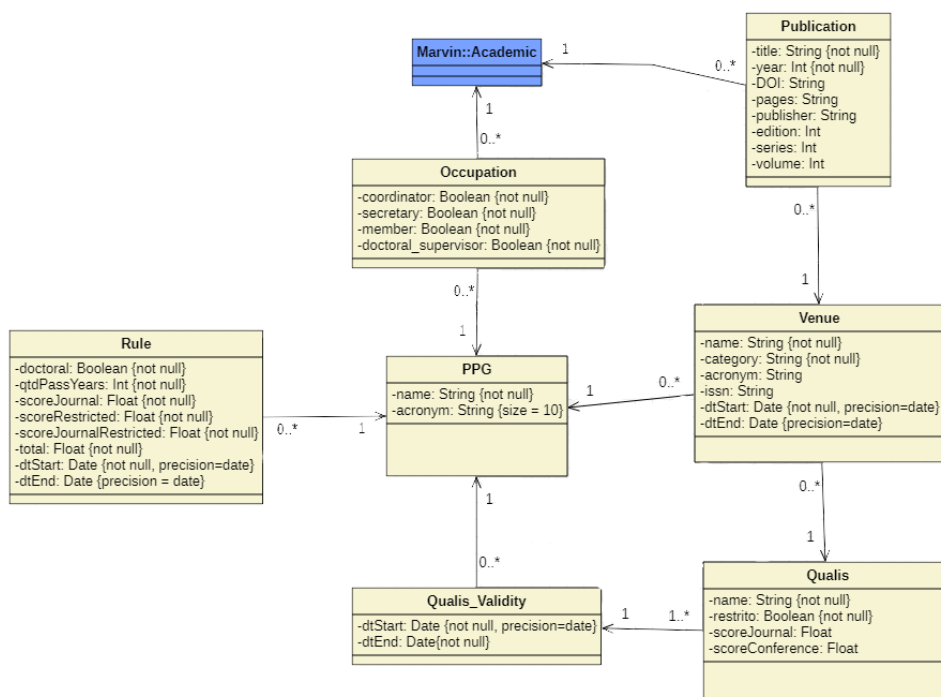


Figura 2 – Modelo de Entidades do C2D.

As regras de negócio são implementadas dentro das classes de serviço que utilizam as classes DAO (apresentadas na Seção 5.2) para utilizar os recursos do banco de dados. Algumas classes de serviço são destinadas atender os CRUDs do sistema, portanto herdam as características do modelo representado na Figura 3, que representa classes do utilitário JButler.

A Figura 4 representa as entidades da lógica de aplicação desse projeto.

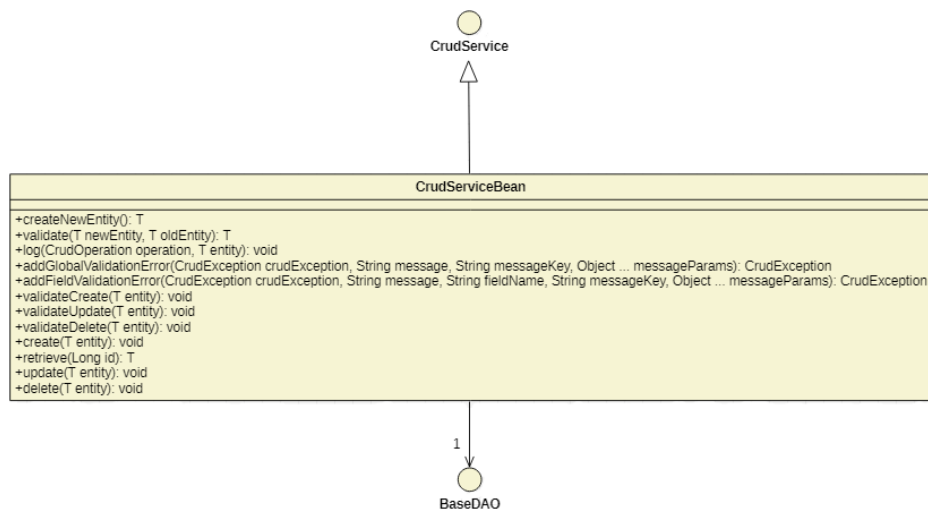


Figura 3 – Classes base do utilitário JButler para serviços de cadastro.

5.2 Camada de Acesso a Dados

Na camada de acesso a dados deve se ter os modelos de persistência de uma aplicação FrameWeb. Os modelos de persistências desse projeto herdam as mesmas características da modelo de persistência da Figura 5, que representa classes do utilitário JButler.

A Figura 6 representa os modelos de persistência principais para acesso a dados do módulo C2D.

5.3 Camada de Apresentação

A camada de apresentação lida diretamente com o usuário, provendo o acesso às funcionalidades do sistema. Algumas das funcionalidades do módulo C2D são de cadastro (CRUD — *create*, *retrieve*, *update*, *delete*), por isso de maneira geral elas seguem o modelo de navegação da Figura 7.

O cálculo de pontuação dos docentes não é uma funcionalidade de cadastro simples, dessa forma ele é representado pela Figura 8.

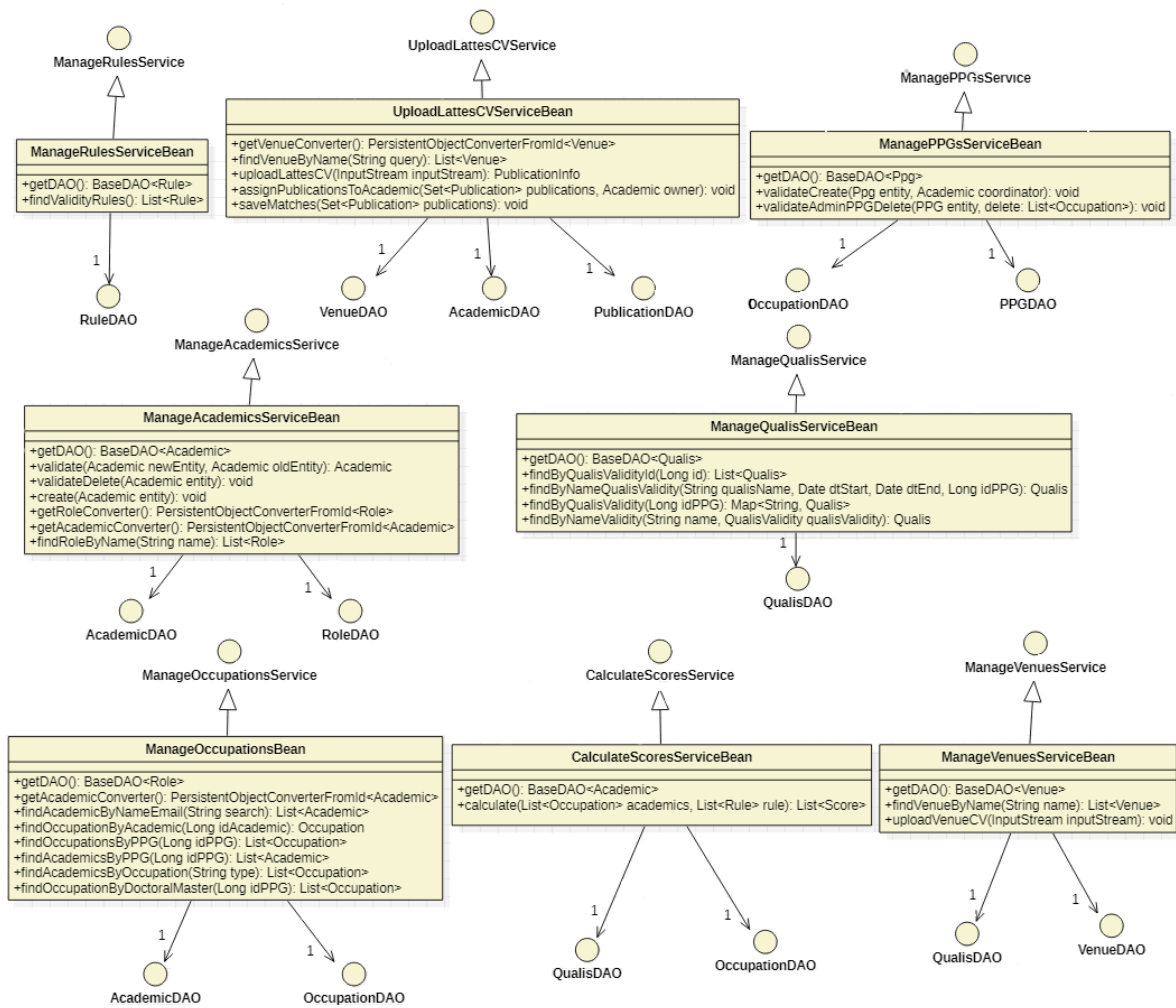


Figura 4 – Modelo de Aplicação do C2D.



Figura 5 – Classes base do utilitário JButler para a camada de acesso a dados.

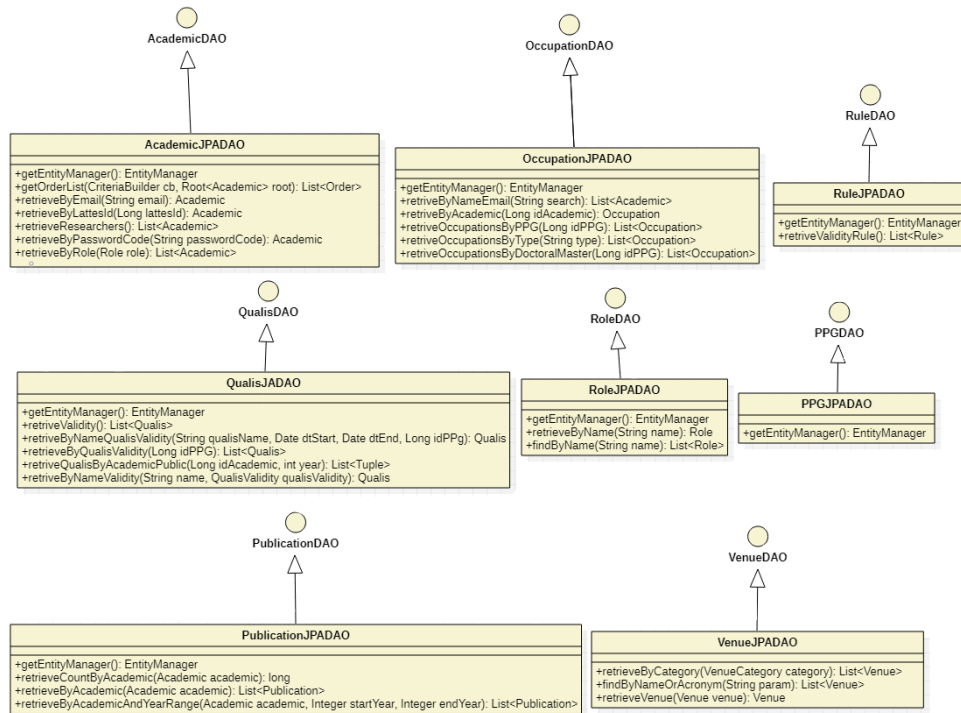


Figura 6 – Modelo de Persistência do C2D.

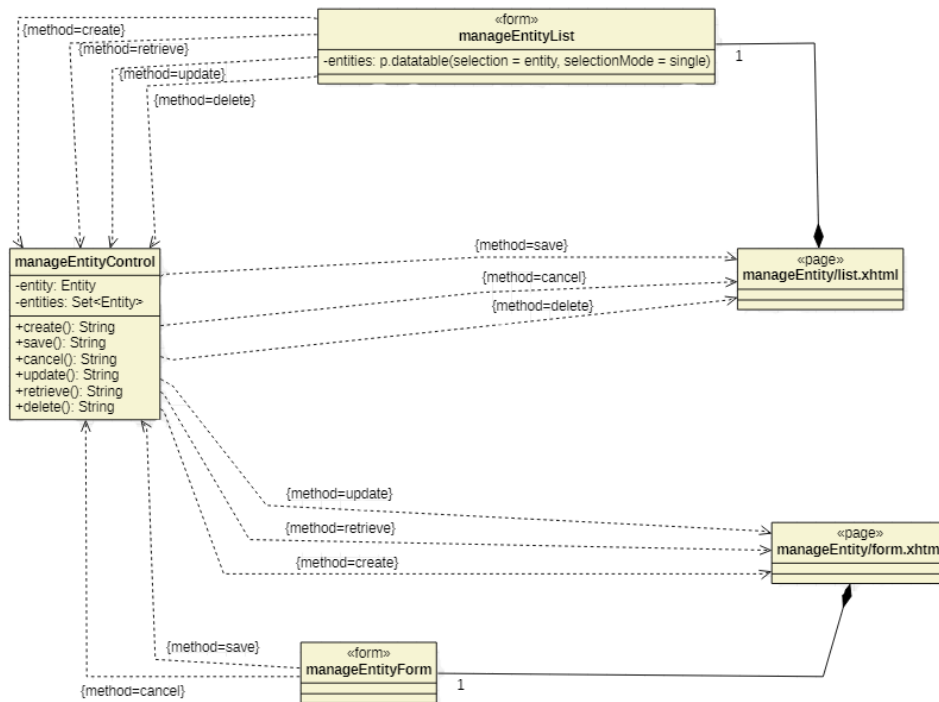


Figura 7 – Modelo de Navegação para funcionalidades de cadastro.

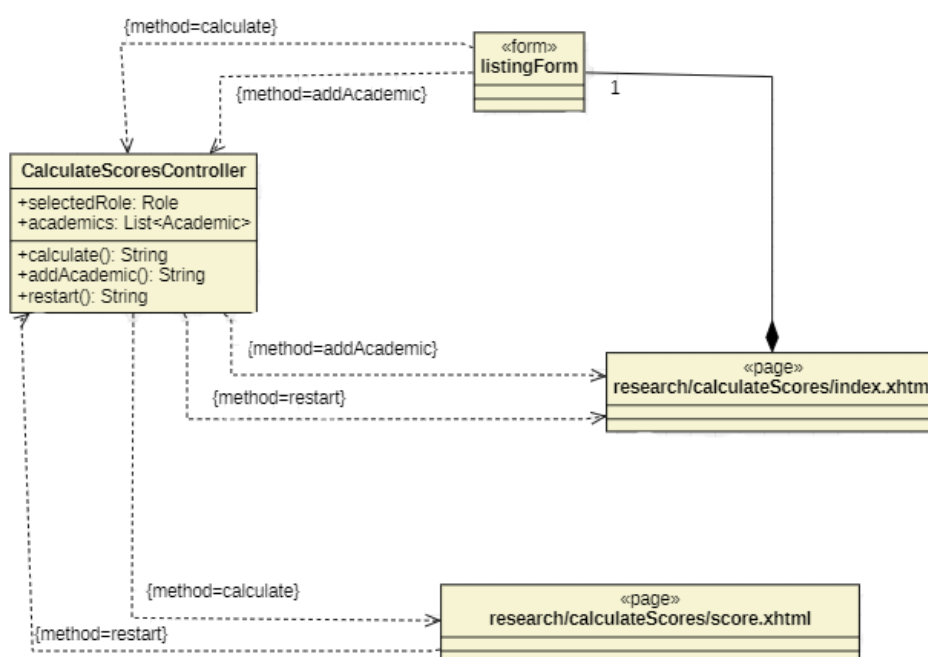


Figura 8 – Modelo de Navegação para o cálculo de pontuação dos docentes.

Referências

FOWLER, M. *Patterns of Enterprise Application Architecture*. 1. ed. [S.l.]: Addison-Wesley, 2002. ISBN 9780321127426. Citado na página 3.

SOUZA, V. E. S. The FrameWeb Approach to Web Engineering: Past, Present and Future. In: ALMEIDA, J. P. A.; GUIZZARDI, G. (Ed.). *Engineering Ontologies and Ontologies for Engineering*. 1. ed. Vitória, ES, Brazil: NEMO, 2020. cap. 8, p. 100–124. ISBN 9781393963035. Disponível em: <<http://purl.org/nemo/celebratingfalbo>>. Citado 2 vezes nas páginas 3 e 4.