

¹**PWAC-IFRS**
Plataforma Web para o Gerenciamento de Atividades Acadêmicas Complementares de Cursos de Graduação
Trabalho de Conclusão do Curso de
Tecnologia em Sistemas Para Internet

Airton da Rocha Bernardoni
Orientador: Fabio Yoshimitsu Okuyama

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS)
Campus Porto Alegre
Av. Cel Vicente, 281, Porto Alegre – RS – Brasil
conaron@gmail.com, fabio.okuyama@poa.ifrs.edu.br

***Resumo.** As atividades complementares são exigidas pelo MEC há quase trinta anos com a finalidade de ampliar a formação do estudante tornando-o um profissional mais apto a lidar com o mercado de trabalho. Atualmente, o IFRS Campus Porto Alegre não possui acesso a uma ferramenta que lide com esta demanda de forma eficaz e que engloba todas as necessidades que o processo carece. Este artigo propõe a criação de uma plataforma para gerir, intermediar e validar os processos de aproveitamento de atividades complementares para o curso de Sistemas para Internet. O desenvolvimento foi realizado utilizando a modelagem do sistema com UML, a plataforma Java, o SGBD PostgreSQL, e a interface foi idealizada através de JSF. Desta forma, esta plataforma possibilita ao coordenador do curso o controle metódico dos processos de aproveitamento através da análise das solicitações dos alunos.*

1. Introdução

As atividades complementares (AC) nos cursos superiores do Brasil foram introduzidas pelo Ministério da Educação em 1996, por meio da Resolução CNE/CES nº 4/96. Essa resolução estabeleceu a necessidade de incluir, nos currículos dos cursos de graduação, atividades acadêmicas que complementem a formação dos alunos, mas que não estejam diretamente relacionadas às disciplinas regulares. O objetivo dessas atividades é ampliar a formação do estudante, permitindo que ele desenvolva habilidades e competências importantes para sua formação profissional e pessoal, como a prática de esportes, participação em projetos de pesquisa, estágios e atividades culturais.

Desde então, as instituições de ensino superior passaram a incluir as atividades complementares em seus currículos e a exigir que os estudantes as cumpram para obterem o diploma de graduação. Essa inclusão proporciona uma formação mais completa e diversificada. Com o crescente número de atividades oferecidas pelas instituições, fica

¹ <https://github.com/conaron/PWAC-IFRS>

cada vez mais complexo gerenciar, registrar e controlar todas as informações envolvidas no processo.

Os sistemas informatizados de gestão de AC (Atividades Complementares) surgem como uma solução para essa demanda, permitindo a automação de tarefas como o registro e acompanhamento das atividades, validação e certificação das horas cumpridas pelos alunos, além de facilitar o acesso e a consulta das informações por parte de alunos e professores. Entre os sistemas utilizados pelas instituições de ensino brasileiras estão o Sistema de Gerenciamento de Atividades Acadêmicas (SGAA), o Sistema de Registros de Atividades Complementares (SISTRAC) pelo IFRS Campus Osório e o Atividades Complementares de Graduação (ACG) pela UFSM, que oferecem recursos específicos para cada tipo de atividade.

Segundo aponta OLIVEIRA (2022, p. 81-98) "O controle informatizado das atividades complementares é fundamental para que as instituições de ensino possam acompanhar de forma precisa e eficiente as atividades realizadas pelos estudantes". Este controle possibilita uma gestão mais eficaz na validação e administração dos requisitos definidos pelo curso, além de transparência e segurança.

Apesar de existirem ferramentas desenvolvidas que implementam a solução necessária para a gestão de AC, as mesmas são elaboradas especificamente para atender as necessidades das entidades que as desenvolveram. Mesmo com propostas encontradas em trabalhos de conclusão de curso que supostamente seriam aplicáveis a qualquer instituição, nada efetivo foi realizado. Visto que o curso de Sistemas para Internet do IFRS Campus POA não possui plataforma que atenda a gestão de AC, este projeto se compromete a elaborar e desenvolver a solução para o curso.

2. Fundamentação Teórica

Para fundamentar o desenvolvimento deste trabalho foi realizada uma pesquisa em periódicos, seminários e matérias publicadas através da internet sobre a participação da tecnologia da informação nos processos educacionais, em específico no controle de Atividades Complementares juntamente com as tecnologias selecionadas para o desenvolvimento deste projeto: Java e JSF.

2.1 Atividades Complementares

As atividades complementares são uma parte importante dos cursos superiores. De acordo com o Ministério da Educação (MEC), as atividades complementares são obrigatórias em todos os cursos de graduação do país e devem representar um máximo de 20% da carga horária total do curso².

Segundo Antunes (2019), as atividades complementares são uma oportunidade para os estudantes ampliarem sua formação acadêmica, desenvolvendo competências e habilidades que vão além do conteúdo das disciplinas regulares. Dentre as atividades complementares mais comuns, estão a participação em projetos de pesquisa, estágios, monitorias, atividades de extensão e a realização de cursos extracurriculares.

² [MEC - Processo nº 23001.000133/2008-37](#)

A realização de atividades complementares pode trazer diversos benefícios aos estudantes de cursos superiores. Em primeiro lugar, ela pode auxiliar no desenvolvimento de habilidades e competências importantes para a atuação profissional, como a capacidade de trabalhar em equipe, a comunicação efetiva e a resolução de problemas.

As atividades complementares podem ser realizadas de diversas formas, incluindo atividades dentro da instituição de ensino, como monitorias e projetos de pesquisa, e atividades fora da instituição, como estágios e atividades de extensão. Existem diversas oportunidades disponíveis para os estudantes de cursos superiores, e é importante que os alunos se informem sobre as possibilidades existentes e escolham aquelas que mais se adequam aos seus objetivos de formação.

A realização de atividades complementares também pode ser vantajosa para a instituição de ensino. Além disso, as atividades complementares podem contribuir para a formação de uma cultura de inovação e empreendedorismo na instituição, o que pode trazer benefícios para a sociedade como um todo.

A realização de atividades complementares pode trazer diversos benefícios aos estudantes, como o desenvolvimento de habilidades e competências importantes para a atuação profissional, contribuindo para a formação de um perfil mais completo e competitivo no mercado de trabalho.

2.2 Tecnologia da Informação

A Tecnologia da Informação (TI) é um campo interdisciplinar que envolve o uso de hardware, software e telecomunicações para processar, armazenar e transmitir informações em diversos contextos. Segundo Bardinelli e Marques (2019), a TI tem se mostrado cada vez mais importante para as organizações, uma vez que possibilita melhorias na eficiência operacional, aumento da produtividade, redução de custos e melhor tomada de decisão. Ainda de acordo com os autores, a TI pode ser vista como um recurso estratégico, capaz de agregar valor ao negócio.

No contexto acadêmico, a TI também tem ganhado destaque. De acordo com Bortoluzzi e Vanti (2017), as tecnologias educacionais vêm sendo amplamente utilizadas para aprimorar os processos de ensino e aprendizagem, possibilitando maior interatividade, colaboração e flexibilidade. Além disso, a TI também tem sido utilizada para gerenciar e monitorar as atividades acadêmicas, como é o caso dos sistemas de gestão de atividades complementares e portfólio.

2.3 Java

Java é uma linguagem de programação de alto nível, orientada a objetos, desenvolvida pela Sun Microsystems em 1995 e atualmente mantida pela Oracle Corporation. É uma das linguagens mais populares do mundo, utilizada tanto no desenvolvimento de aplicativos desktop como em sistemas empresariais de grande escala.

A linguagem Java foi projetada com o objetivo de ser portátil e independente de plataforma, permitindo que os programas escritos em Java sejam executados em diferentes sistemas operacionais sem a necessidade de adaptações adicionais. Isso é possível devido ao uso da máquina virtual Java (JVM), que é responsável por interpretar

o código escrito em Java e convertê-lo em instruções que podem ser executadas pelo sistema operacional.

2.4 JavaServer Faces (JSF)

O JavaServer Faces (JSF) é uma framework de interface de usuário baseada em componentes para aplicativos Java web. O JSF foi criado pela Sun Microsystems (agora Oracle Corporation) em 2004 e se tornou um padrão da Java Community Process (JCP) em 2005. A framework foi desenvolvida com base no padrão de design Model-View-Controller (MVC) e possui recursos que facilitam a criação de interfaces de usuário complexas e personalizadas para aplicativos da web.

A próxima seção apresenta uma lista de trabalhos relacionados com propostas similares às apresentadas neste trabalho.

3. Trabalhos Relacionados

Realizou-se uma pesquisa na internet para determinar se já existiam sistemas de gestão de atividades complementares disponíveis para uso com funcionalidades semelhantes às necessárias ao curso de Sistemas para Internet. Foram identificadas algumas características fundamentais, que serão descritas a seguir e indicadas em um quadro comparativo.

3.1 Trabalho 1 - SGAA (SIGAA) - Sistema de Gerenciamento de Atividades Acadêmicas

O SGAA é um sistema informatizado utilizado por instituições de ensino para gerenciar todas as atividades acadêmicas dos estudantes, incluindo matrícula, grade de horários, notas, faltas, histórico escolar e outras informações relevantes. O sistema permite que a instituição de ensino automatize e integre vários processos acadêmicos, o que torna o gerenciamento de dados mais fácil, seguro e preciso.

O SGAA pode ser integrado a outros sistemas de gerenciamento acadêmico, como o Sistema de Gerenciamento de Biblioteca (SGB), para fornecer uma visão abrangente das atividades dos alunos, mas infelizmente somente possui a gestão da quantidade de horas devidas totais relativas às atividades complementares.

3.2 Trabalho 2 – UFSM – Ciência da Computação - ACG

As Atividades Complementares de Graduação (ACG) são administradas diretamente pelo portal do aluno através de um processo chamado de Requerimento de Dispensa. Os processos de requisição são similares aos propostos na plataforma, mas não há muita informação sobre as etapas de avaliação e de controle das horas disponíveis por categoria.

A pesquisa restringiu-se ao que foi possível retirar de informação por manuais de orientação, visto que é necessário ser um aluno em graduação para ter acesso ao portal.

3.3 Trabalho 3 – GAC - Sistema para Gerenciamento das Atividades Complementares

Este trabalho de conclusão para o Curso de Tecnologia em Sistemas para Internet na Universidade Tecnológica Federal do Paraná (UTFPR) propõe-se a apresentar uma solução paralela à gestão de atividades complementares do curso citado.

Segundo Marlon [2018] a ausência de um sistema que administre as atividades complementares é um problema presente também na UTFPR e a solução apresentada não parece ter sido implementada, visto que o portal da instituição não possui nenhum link ou acesso ao sistema proposto, mas apenas às regulamentações e formulários para envio de solicitação.

3.4 Trabalho 4 - IFRS – SWGA - Sistema Web para Gerenciamento de Atividades Complementares

Este projeto foi apresentado ao IFRS Porto Alegre por Machado [2022]. A proposta dele consistia na modelagem e implementação de um sistema de controle de AC para o IFRS. As soluções sugeridas incluíam controle de categorias, análise de processo, restrição de horas aproveitadas, e a possibilidade de análise do processo pelo coordenador do curso. Infelizmente o projeto não saiu da fase de planejamento e acabou nunca sendo implementado.

O Quadro 1 executa uma comparação entre os sistemas relacionados e a plataforma proposta. O quadro irá apresentar somente as funcionalidades interpretadas como comuns a todos os sistemas para estabelecer visualmente o diferencial do sistema proposto. Uma visão mais profunda das funcionalidades do sistema está presente no levantamento de requisitos funcionais (seção 5.1).

Quadro 1 – Comparação entre os sistemas analisados e o sistema proposto

Sistema	SGAA	ACG	GAC	SWGA	PWAC-IFRS
Emissão de relatórios	Não	Não Informado	Sim	Sim	Sim
Controle de permissões de acesso por usuário	Sim	Sim	Sim	Sim	Sim
Acesso a partir da Internet	Sim	Sim	Sim	Sim	Sim
Upload da documentação comprobatória das AC	Não	Sim	Sim	Sim	Sim
Calcula as horas de AC	Não	Sim	Sim	Sim	Sim
Consultar estado do processo de AC	Não	Sim	Sim	Sim	Sim
Código aberto	Não Informado	Não	Não Informado	Sim	Sim
Disponível ao IFRS	Não	Não	Não	Não	Sim

Com base na fundamentação teórica realizada e no levantamento de trabalhos relacionados foi proposta a plataforma descrita na próxima seção.

4. Proposta Metodológica

Para desenvolver este projeto, foi preciso escolher um processo de desenvolvimento que determinasse quais etapas e elementos de software seriam requeridos para criar a solução. Foi, então, decidido seguir um conjunto de etapas amplamente utilizado para desenvolver diferentes tipos de software: coleta de requisitos, análise de requisitos, design e implementação com testes.

4.1 Levantamento e Análise dos Requisitos

Essa etapa é determinada pelo entendimento do problema, contextualizando o cenário e escalando por importância as necessidades e recursos do sistema. Para levantar os requisitos foram necessárias entrevistas com o coordenador do curso de Sistemas para Internet do IFRS com o propósito de ajustar o que seria ideal para o funcionamento do sistema. Assim foram elaborados estudos de caso sendo que dois deles estão dispostos na Seção 5.

Concluiu-se que a plataforma deve disponibilizar ao coordenador do curso a administração das categorias determinadas através dos currículos (Projetos Pedagógicos) do curso, a análise das solicitações feitas por alunos pertencentes a este currículo e o deferimento ou não destas solicitações. Para a modelagem dos requisitos optou-se pela utilização do diagrama de casos de uso (Figura 1), a fim de evidenciar as funcionalidades fornecidas pela plataforma. O cadastro dos currículos é realizado através de uma interface gráfica com o usuário coordenador. Uma vez cadastrado, o currículo permite a inclusão de categorias vinculadas a este. Esta inclusão pode ser feita de forma manual, com um registro por vez ou em larga escala, por importação de arquivos CSV.

Com a disponibilidade das categorias, o usuário aluno pode realizar solicitações de aproveitamento de horas por categoria específica restrito a quantidade de horas determinadas na categoria descontadas as horas já solicitadas em análise ou deferidas. Esta solicitação possibilita o envio do comprovante como certificado, diploma ou documento pertinente em formato PDF.

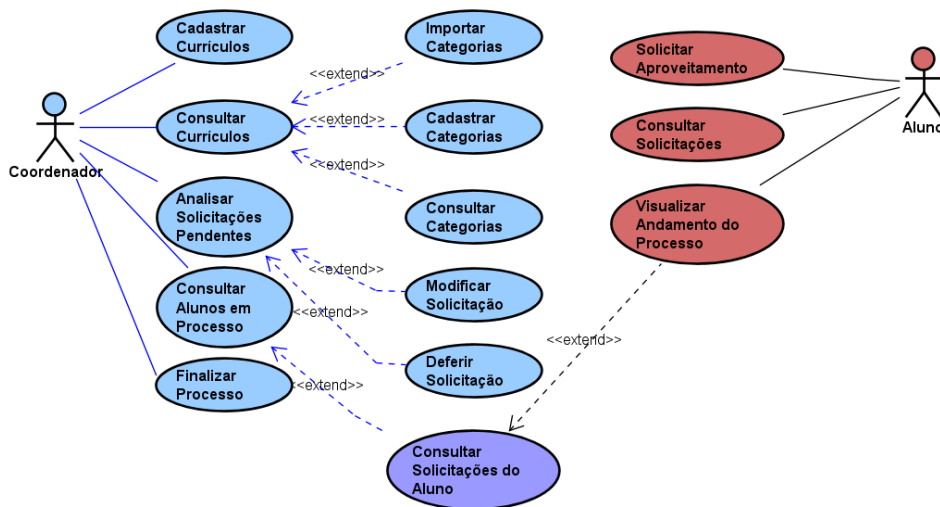


Figura 1. Diagrama de Casos de Uso da Plataforma Proposta

4.2 Análise e Projeto

Nessa fase foi levado em consideração o projeto detalhado com diagramas, a linguagem de programação e o Sistema Gerenciador de Banco de Dados (SGBD) que serão utilizados, os padrões de front-end ou de interface gráfica com o usuário.

A partir da análise dos requisitos observou-se que o login de ambos os usuários seriam através de LDAP ³(*Lightweight Directory Access Protocol*), trazendo dados de uma base de terceiros, gerando a necessidade de criar um cadastro espontâneo a partir do login do usuário.

Ao realizar login, o sistema obtém do retorno do LDAP, quando autenticado, informações de base como categoria do usuário, curso, ingresso, situação de matrícula e nome. Estes dados são utilizados para realizar o cadastro automático do usuário no sistema, identificando-o posteriormente a uma chave pré-definida fornecida no retorno do LDAP, dispensando a necessidade de manipulação de dados pessoais tanto do coordenador quanto do aluno.

A partir da disponibilidade do currículo cadastrado pelo coordenador do curso, o aluno pode solicitar o aproveitamento de horas em AC por seleção de categoria respeitando compulsoriamente a quantidade de horas disponíveis para o aproveitamento. Ao realizar a solicitação, o aluno deve determinar a categoria, a quantidade de horas e enviar o documento comprobatório que justifique a requisição. Este recurso é disponibilizado através de um único formulário ao aluno para simplificar a operação.

A solicitação é definida como pendente para ambos, passando ao coordenador a possibilidade de analisar, corrigir a categoria ou quantidade de horas quando necessário e deferir a solicitação, quando for o caso. As solicitações pendentes ficam em evidência na plataforma para ambos como forma de auxiliar o andamento fluido do processo.

Uma vez que a quantidade total de horas necessárias seja atingida, o aluno poderá realizar a solicitação do processo ao protocolo diretamente via sistema, com a geração do formulário e o envio do e-mail com cópia direcionada ao coordenador.

O diagrama de classes foi utilizado para mapear os recursos estruturais de código vinculados a esta fase do processo de desenvolvimento. Esse diagrama leva em consideração o projeto do Webservice e alguns aspectos da reflexão computacional. A partir do diagrama de classes foi construído o código, bem como foram realizados alguns testes do sistema. Alguns detalhes relacionados com a implementação e testes encontram-se descritos na próxima seção.

4.3 Implementação e Testes

Como já foi mencionado, a fase de implementação envolve a criação das soluções definidas nas etapas anteriores, utilizando as tecnologias selecionadas. Para desenvolver a solução proposta neste estudo, foram escolhidas as seguintes tecnologias: a linguagem

³ <https://ldap.com/>

de programação Java, os frameworks Hibernate e PrimeFaces com JSF. Em relação ao banco de dados, a opção foi pelo PostgreSQL. Apesar de o MySQL ser reconhecido pela sua agilidade e oferecer grande parte dos recursos disponíveis no PostgreSQL, este último se destaca pela sua robustez no tratamento de dados, com uma ampla variedade de tipos.

Inicialmente, foi necessário configurar as ferramentas para o desenvolvimento do sistema. Iniciou-se a configuração pela IDE NetBeans, após foram configurados a ferramenta Maven e os frameworks JPA, Hibernate, JSF e Primefaces. Ao concluir essa configuração inicial, procedeu-se com a implementação e alguns testes, que foram realizados em paralelo com a etapa de codificação.

Para a realização de todos os testes foi definido dentro do projeto um pacote cujo nome é “teste”, o qual é composto por classes que realizam a execução de métodos verificando a passagem de parâmetros e seu retorno. Procurou-se cobrir todos os casos hipotéticos concebidos pelo desenvolvimento.

Devido às restrições de acesso ao servidor LDAP, visto que o mesmo permite o login somente em sua rede interna, os testes foram realizados com retornos idealizados, com base nas informações fornecidas atualmente pelas plataformas SIGAA e MOODLE utilizadas pelo IFRS. O versionamento do projeto é gerenciado pela plataforma GitHub, considerando sua versatilidade e disponibilidades de ferramentas para integração com a IDE escolhida.

A próxima seção apresenta dois estudos de caso que foram desenvolvidos para melhor contextualizar a plataforma desenvolvida.

5. Sistema desenvolvido

A seguir são descritos os requisitos funcionais, a modelagem e estudos de caso do sistema desenvolvido.

5.1 Requisitos Funcionais

Com base na análise da proposta do SWGA e em entrevista com o coordenador do curso SSI foram definidos os requisitos funcionais apresentados no quadro 2.

Quadro 2 – Requisitos funcionais do sistema proposto

Identificação	Descrição	Origem da Ação
RF01	O sistema permitirá o acesso do usuário através do usuário (número de matrícula) e senha.	Aluno ou coordenador
RF02	O sistema deve permitir gerenciar os currículos do curso (Cadastrar, Excluir e Editar).	Coordenador
RF03	O sistema deve permitir gerenciar as categorias de atividades complementares (Cadastrar, Excluir e Editar).	Coordenador
RF04	O sistema deve permitir pesquisar/emitir relatório das solicitações que estão pendentes dos alunos.	Coordenador
RF05	O sistema deve permitir pesquisar/emitir relatório das solicitações que não estão pendentes dos alunos.	Coordenador

RF06	O sistema deve permitir ao coordenador retornar a solicitação para o aluno se precisar de correções.	Coordenador
RF07	O sistema deve exigir observações nas solicitações cadastradas pelo aluno que não passaram na avaliação, antes que o coordenador retorne a solicitação para o aluno.	Coordenador
RF08	O sistema deve permitir ao coordenador visualizar a solicitação após retornar para o aluno, mas só deve permitir alterações nele se o aluno reenviar a solicitação para o coordenador.	Coordenador
RF09	O sistema deve permitir definir o número de horas aproveitadas em cada atividade do aluno.	Coordenador
RF10	O sistema deve permitir alterar o nome formal do curso.	Coordenador
RF11	O sistema deve permitir alterar o e-mail da secretaria acadêmica.	Coordenador
RF12	O sistema deve permitir gerar a documentação referente aos processos para enviar para a secretaria acadêmica.	Coordenador
RF13	O sistema deve permitir gerenciar as atividades cadastradas pelo aluno (Cadastrar, Excluir e Editar).	Aluno
RF14	O sistema deve permitir fazer o upload dos documentos comprobatórios das atividades cadastradas pelo aluno.	Aluno
RF15	O sistema deve permitir pesquisar/emitir relatório sobre as horas aceitas/recusadas.	Aluno
RF16	O sistema deve permitir ao aluno cadastrar solicitação para avaliação do coordenador.	Aluno
RF17	O sistema deve permitir ao aluno visualizar a solicitação após o cadastro para avaliação do coordenador, mas só deve permitir alterações nele se o coordenador não definir a solicitação como aceita.	Aluno
RF18	O sistema não deve permitir ultrapassar o número de horas permitidas por tipo/categoria de atividade.	Sistema
RF19	O sistema deve calcular a soma das horas aproveitadas para atividades complementares por cada aluno. No caso da exclusão de alguma solicitação, antes da conclusão do processo, as horas aproveitadas serão recalculadas.	Sistema
RF20	O sistema não deve permitir que o aluno cadastre solicitação para categorias de atividades nas quais o total de horas permitidas já foram aproveitadas/aceitas.	Sistema
RF21	O sistema deve, quando não cadastrado, pré-cadastrar o curso com base em campo provido por login LDAP.	Sistema
RF22	O sistema deve, quando não cadastrado, exigir o cadastro do telefone do aluno para preenchimento de formulário à secretaria acadêmica.	Sistema
RF23	O sistema deve impedir a exclusão de uma categoria de atividade complementar enquanto a mesma estiver sendo usada em processo de solicitação.	Sistema
RF24	O sistema deve impedir a exclusão de um currículo enquanto alguma categoria de atividades complementar estiver sendo usada.	Sistema

5.2 Caso de uso

O primeiro caso de uso tem relação com a administração de currículos (PPC) relacionados ao curso que será administrado pelo coordenador em vigência. Desde a ação do login, o cadastro de informações correntes será espontâneo, com base no retorno dos dados de perfil fornecidos pelo LDAP utilizado para fazer o login. Por exemplo, a Figura 2 ilustra a interação do coordenador do curso com um novo currículo disposto para cadastro. O sistema fornece a funcionalidade de cadastro manual, assim como ajustes caso necessários, e importação através de arquivo CSV (Comma-separated values).

Este currículo determinará uma lista específica de categorias de atividades complementares a dispor de gestão, tornando a análise da utilização delas de forma mais precisa independente do período em que for auditado. A utilização de um currículo é disposta pelo início de sua vigência, fazendo com que o sistema sempre disponha, de forma automática, ao aluno a lista de categorias que compete à sua data de matrícula.

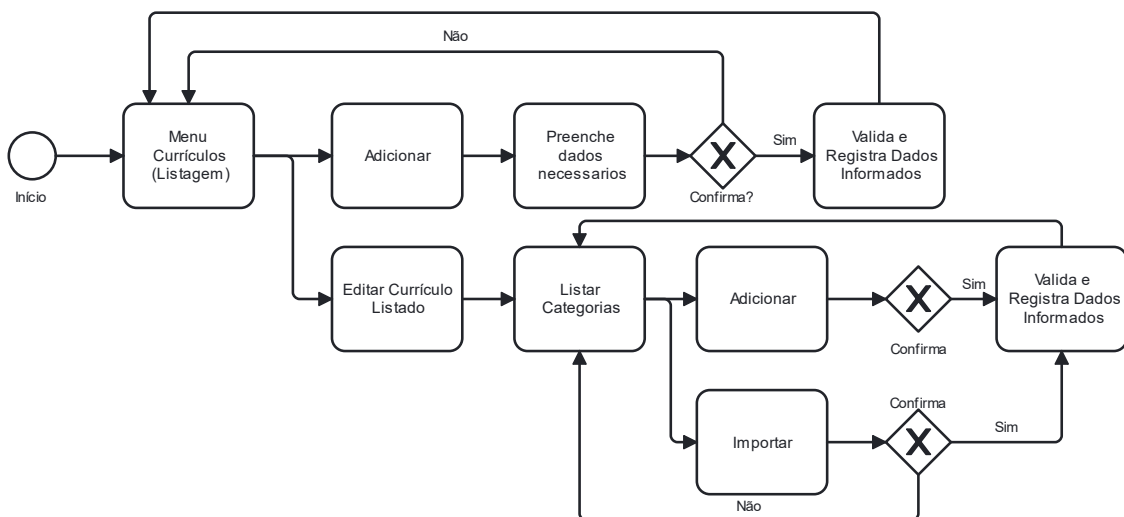


Figura 2. A inclusão de currículo para disponibilidade de solicitações.

A solicitação do aproveitamento da atividade complementar segue a direção contrária dos eventos de cadastro. O aluno realiza a solicitação via plataforma e anexa o documento comprobatório no formato de PDF nela. O coordenador recebe, por sua vez, a notificação da solicitação ao entrar no portal, estando ela na lista de pendências a resolver.

Após a análise da solicitação, o coordenador pode determinar que o documento não é válido (como visto na Figura 3) e solicitar a alteração do documento. Neste estágio, o sistema envia um e-mail para o aluno realizar a devida alteração. Após a solicitação sofrer a alteração devida, ela retorna para a lista de pendências do coordenador que irá avaliá-la novamente e deferir, se for este o caso.

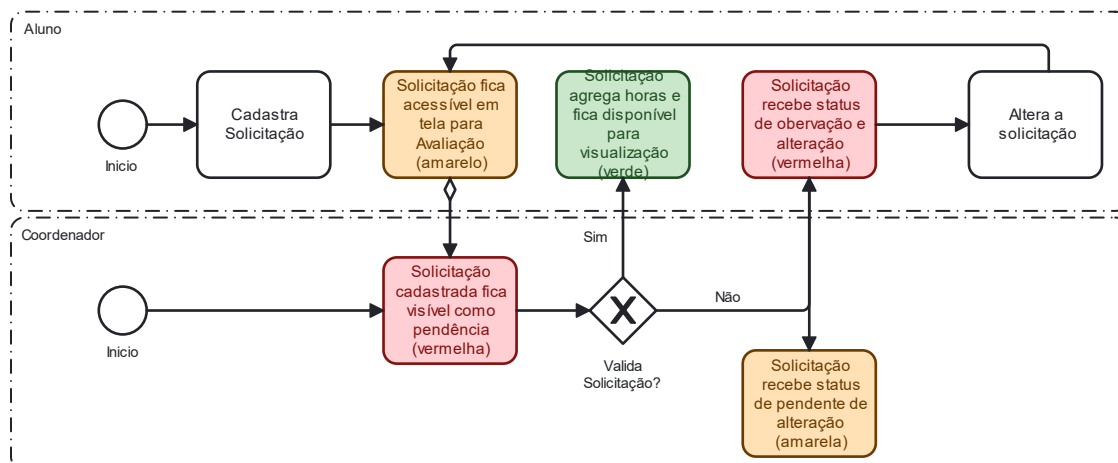


Figura 3. A solicitação de um aproveitamento e sua análise.

5.3 Validação

Pelo papel de aluno, foram testados os cadastros de solicitações com o upload dos documentos comprobatórios necessários, o monitoramento da situação das solicitações conforme a resposta do coordenador e a alteração das solicitações, quando requisitadas pela resposta do coordenador. Também foi testado o cadastro de telefone para compor o formulário preenchido automaticamente pelo sistema para formalização ao setor de protocolo.

5.4 Telas do Sistema

No Apêndice C estão capturas de tela do sistema, onde são demonstradas as etapas do segundo caso de uso. É executado o cadastro de uma solicitação e a mesma é disponibilizada para avaliação. Após a avaliação, o processo fica visível como disponível para formalização e encerramento pela coordenação do curso.

6. Considerações Finais

Através da pesquisa sobre plataformas similares, descobriu-se que as poucas referências que apresentam funcionalidades satisfatórias são próprias ou não foram implementadas. Embora não enumerados neste documento a fim de evitar a sobrecarga de informações, outros trabalhos de conclusão com propostas alternativas foram encontrados, mas evidenciam a demanda de um sistema de código aberto em âmbito nacional.

Este trabalho foi desenvolvido primeiramente com o objetivo de atender às necessidades do curso Sistemas para Internet do IFRS - Campus Porto Alegre, mas sua estrutura prevê que possa ser utilizado por outros cursos paralela ou simultaneamente, se assim for do interesse da instituição.

As dificuldades no processo de implantação concentraram-se na administração do tempo de vida dos objetos utilizados, visto que a linguagem de programação é Java e este aspecto deve ser severamente respeitado. O impedimento de realização de testes de autenticação através do LDAP também dificultou a modelagem, visto que as informações fornecidas pelo retorno são taxativas para determinar quais informações necessitam ser

complementadas em um possível pré-cadastro.

Conforme já mencionado, a arquitetura da plataforma viabiliza a gestão de aproveitamento de horas por atividades acadêmicas complementares previstas no projeto pedagógico do curso por parte interativa do coordenador e do aluno do curso. Como melhorias futuras, espera-se fornecer recursos gráficos ou personalizados conforme a necessidade com base no uso do sistema possa sugerir.

Referências

- OLIVEIRA, J. A. et al. (2022), **Controle informatizado de atividades complementares no ensino superior: uma proposta de sistema web**. Revista de Ensino, Ciência e Tecnologia, v. 12, n. 1, p. 81-98.
- BARDINELLI, T. C. B.; MARQUES, J. P. L. (2019), **A tecnologia da informação como recurso estratégico nas organizações**. Revista Científica do ITPAC, 12(2), 1-10.
- BORTOLUZZI, S. C., & VANTI, A. E. (2017), **Tecnologias educacionais e as tendências pedagógicas**. Revista de Tecnologia Aplicada, 6(2), 5-19.
- ANTUNES, A. S. A. (2019), **Atividades complementares no ensino superior: um estudo sobre o processo de reconhecimento das atividades na formação acadêmica**. Revista Brasileira de Aprendizagem Aberta e a Distância, v. 18, n. 1, p. 11-25.
- MINISTÉRIO DA EDUCAÇÃO. **Parecer CNE/CES nº 239/2008, de 6 de novembro de 2008 - Carga horária das atividades complementares nos cursos superiores de tecnologia**. Disponível em: http://portal.mec.gov.br/cne/arquivos/pdf/2008/pces239_08.pdf. Acesso em 29/04/2023.
- MIGUEL, Marlon; FRANCO, Mateus Henrique (2018), **Sistema para gerenciamento das atividades complementares**. Disponível em <http://repositorio.utfpr.edu.br/jspui/bitstream/1/15669/1/sistemagerenciamentoatividadescomplementares.pdf>. Acesso em 01/05/2023.
- BURNS, Ed; SCHALK, James (2009). **JavaServer Faces 2.0: The Complete Reference**. McGraw-Hill Osborne Media.
- SIGAA - https://docs.info.ufrn.br/doku.php?id=suporte:sigaa:visao_geral. Acessado em 4 jul.. 2023.

APÊNDICE A

CONFIGURAÇÃO E TESTES REALIZADOS

Para usar as ferramentas descritas previamente alguns problemas foram encontrados na configuração do projeto:

1. Servidor JBOSS Wildfly – definição local de serviço e JRE com versão específica para o servidor.
2. Framework JSF – utilização de Jakarta Faces 4.0. A migração do JSF 2.3 para o 3.0/4.0 demanda algum tempo, mas é bem mais performática e reduz bastante a quantidade de código.
3. Framework Primefaces – utilização da versão 12.0. A integração do Primefaces com a implementação JSF da Jakarta é muito mais estável que suas versões anteriores. Os escopos de blocos DOM para gestão do AJAX são administrados de forma bem mais simples.
4. Framework Hibernate – utilização da versão 6.1 que implementa, por sua vez o framework Jakarta Persistence 3.0 (JPA). A seguinte versão também precisou ser adaptada dos padrões de schema anteriores (xsd), mas também ficou bem mais versátil, dispensando diversos parâmetros na construção das unidades de persistência;
5. Maven – utilização do repositório para as bibliotecas não fornecidas pelo JRE. O gerenciamento das frameworks utilizadas assim como trajetórias específicas de deploy foram utilizadas através de plugins.

Ao concluir a definição da estrutura do projeto, passou-se para os testes, os quais encontram-se descritos abaixo:

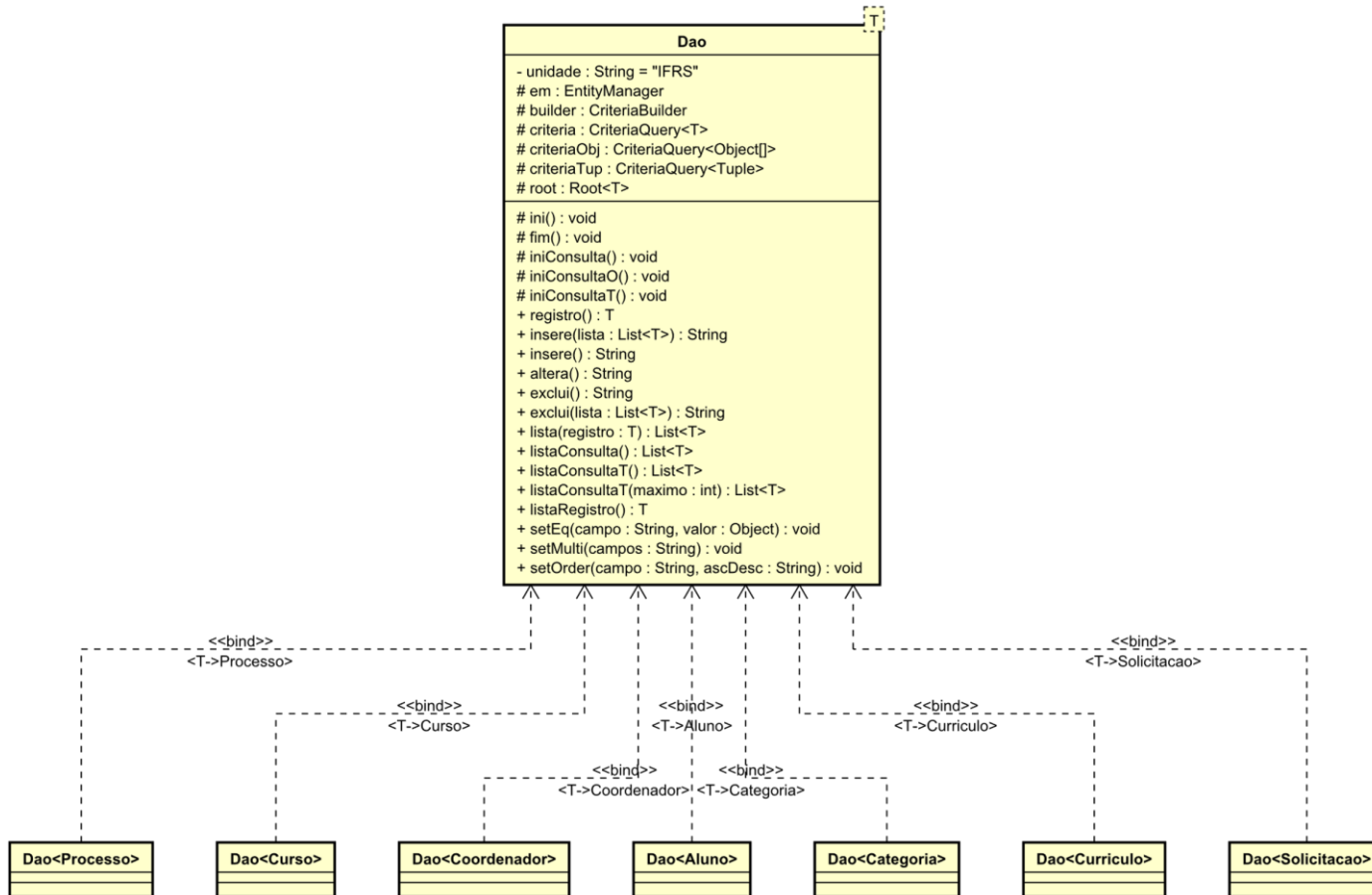
1. Construção de uma classe de conexão e de operações básicas de manipulação de dados (CRUD - *Create, Retrieve, Update e Delete* / inclusão, consulta, atualização e exclusão) construída de forma a atender a todas as classes de entidade, fornecendo o recurso de conexão às mesmas sem a necessidade de instanciar um terceiro objeto;
2. Determinação de blocos de uso das views conforme o usuário. Com a separação das classes “Beans” por utilização, pôde-se criar comportamentos personalizados do mesmo objeto para usuários diferentes relacionados aos seus ambientes de interação com o sistema.
3. Utilização de constantes nas classes entidade para diminuir o volume de código nos métodos onde são realizadas consultas personalizadas. Modificação da abordagem da consulta “nomeada” através de padrão desenvolvido na construção da classe modelo com a inclusão de método com retorno de String que simulasse a sintaxe do Hibernate para consulta. O método foi chamado de “toQuery()” em referência aos métodos padrões sobrescritos como, por exemplo, “toString()”;
4. Criação de segunda classe modelo verificar se o CRUD reflexivo iria ou não funcionar. Após o sucesso desse teste foi realizado o acoplamento da classe de conexão com a de manipulação para agilização do processo de construção e finalização de sessão;

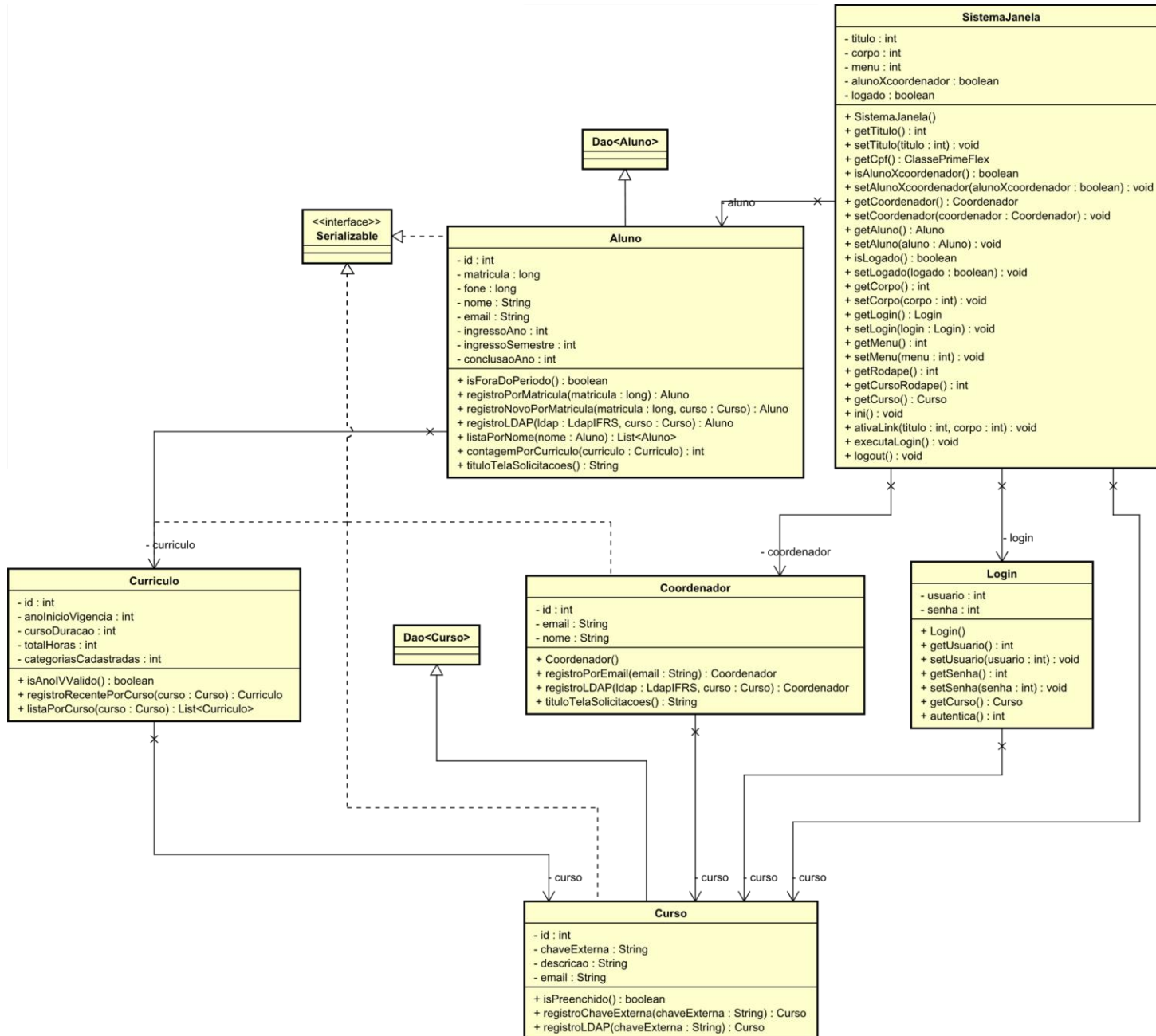
5. Iniciado recurso de construção de classe “construtora de classes” para estruturar as classes pertinentes a Objetos IoT consumidores do projeto, que ainda está em desenvolvimento

A estrutura dos arquivos XHTML foi definida com base em templates personalizados. Eles utilizam uma arquitetura que demanda de pouca adequação para cada caso e seguem uma linha de ação similar em qualquer bloco que seja aplicado.

APÊNDICE B

DIAGRAMA DE CLASSES





APÊNDICE C
TELAS CAPTURADAS DO SISTEMA

Solicitações Incluir Sol

Solicitações em andamento - 65

Categoria	Descrição
Apresentação de trabalho científico	Teste 1
Apresentação de trabalho científico	Teste 2
Atividade como pesquisador de iniciação científica	Teste 3
Atividade de monitoria de componentes	Teste 4

12/07/2023

Cadastro de Solicitação

Categoria: Atividade de monitoria de componentes

Descrição *: Teste 5

Período: 07/2023

Horas Realizadas: 10

Horas Aproveitadas: 5

Documento Comprobatório: [Documento Comprobatório](#)

Atividade de monitoria de componentes curriculares relacionados à área do curso

[✓ Salvar](#) [✗ Cancelar](#)

Administradores Sair

Solicitações

Busca

Observações

-
-
-
-

Matrícula: 2023001111 - ALUNO

Solicitações em andamento - 65 horas de 70 (92,86%) - Faltam 5 horas. (5 horas em avaliação)

Q Busca

Categoria	Descrição	Período	Horas Realizadas	Horas Aproveitadas	Observações
-----------	-----------	---------	------------------	--------------------	-------------

Atividade de monitoria de componentes

Teste 5

07/2023

10

5



Apresentação de trabalho científico

Teste 1

06/2023

10

10



Apresentação de trabalho científico

Teste 2

06/2023

100

20



Atividade como pesquisador de iniciação científica

Teste 3

06/2023

100

30



Atividade de

12/07/2023

IFRS - SSI - SISTEMAS PARA INTERNET

Matrícula: 2023001111 - ALUNO

Solicitações

Data Evento	Situação	Categoria	Aluno	
11/07/2023	Em avaliacao	Atividade de monitoria de componentes	AIRTON DA ROCHA BERNARDONI	
11/07/2023	Aprovada	Apresentação de trabalho científico	AIRTON DA ROCHA BERNARDONI	
11/07/2023	Aprovada	Apresentação de trabalho científico	AIRTON DA ROCHA BERNARDONI	
11/07/2023	Aprovada	Atividade como pesquisador de iniciação científica	AIRTON DA ROCHA BERNARDONI	
11/07/2023	Aprovada	Atividade de monitoria de componentes	AIRTON DA ROCHA BERNARDONI	

Solicitações Currículos

Em Avaliação Em correção

Solicitações

Data Evento	Situação
11/07/2023	Em avaliacao
11/07/2023	Aprovada
11/07/2023	Aprovada
11/07/2023	Aprovada
11/07/2023	Aprovada

12/07/2023

Solicitação

Categoria: Atividade de monitoria de componentes curriculares relacionados à área do curso

Descrição: Teste 5

Período: 07/2023

Horas Realizadas: 10

Horas Aproveitadas: 5

Documento Comprobatório: Documento Comprobatorio

Observação: Observação

✓ Solicitar Correção ✓ Validar ✗ Cancelar

Sair Solicitações

Relatório

Busca

Coordenador: COORDEDNADOR

Processos

Aluno	Matrícula	Horas Totais	Horas Aprov.	Horas Pendentes	Abertura	
AIRTON DA ROCHA BERNARDONI	2023004926	70	70	0	22/06/2023	