

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Gustavo Silva de Oliveira

**Manutenção do Sistema de Cadastro de
Atividades Docente**

Uberlândia, Brasil

2024

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Gustavo Silva de Oliveira

**Manutenção do Sistema de Cadastro de Atividades
Docente**

Trabalho de conclusão de curso apresentado
à Faculdade de Computação da Universidade
Federal de Uberlândia, como parte dos requi-
sitos exigidos para a obtenção título de Ba-
charel em Sistemas de Informação.

Orientador: Bruno Augusto Nassif Travençolo

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Sistemas de Informação

Uberlândia, Brasil

2024

Resumo

Este trabalho aborda a manutenção corretiva e evolutiva do Sistema de Cadastro de Atividades Docente (SCAD), amplamente utilizado por docentes da Faculdade de Computação (FACOM) para a elaboração de relatórios de atividades, essenciais nos processos de progressão e promoção de carreira. A proposta deste projeto é identificar problemas e oportunidades de melhoria com base em *feedback* dos usuários, com foco na usabilidade da interface e na experiência de uso. Foram realizadas análises de pontos reportados e implementadas soluções que aprimoram a navegabilidade, a organização visual e a eficiência do sistema. O trabalho contribui para a continuidade do aperfeiçoamento do sistema, alinhando-o às necessidades dos usuários e mantendo sua relevância no contexto acadêmico.

Palavras-chave: relatório, atividade, docente, progressão, manutenção.

Lista de ilustrações

Figura 1 – Tela de Login.	11
Figura 2 – Menu Pessoal.	11
Figura 3 – Tela da página Lista de Relatórios.	12
Figura 4 – Tela do formulário de Próximo Nível.	13
Figura 5 – Tela do formulário das Informações do Relatório.	13
Figura 6 – Tela da página Lista de Atividades.	14
Figura 7 – Tela das Pendências.	14
Figura 8 – Tela da página Nova Atividade.	15
Figura 9 – Tela do Resumo de Pontuação.	16
Figura 10 – Tela do Resumo de Pontuação.	16
Figura 11 – Menu Avaliador.	17
Figura 12 – Tela da aba Relatórios em Andamento.	17
Figura 13 – Tela da aba Relatórios Encerrados.	18
Figura 14 – Tela da aba Relatórios em Avaliação.	18
Figura 15 – Tela da página de lista de atividades para avaliação.	19
Figura 16 – Tela da aba Reelatórios Avaliados.	19
Figura 17 – Menu Administrador.	20
Figura 18 – Tela da página Gerenciamento de Professores.	21
Figura 19 – Tela dos formulários de Nível Atual e Login da página Cadastro de Novo Professor.	21
Figura 20 – Tela dos formulários das Informações Gerais do Professor e Permissões da página Cadastro de Novo Professor.	22
Figura 21 – Tela da página Tabela de Pontuação.	22
Figura 22 – Tela da página do cadastro de uma Nova Tabela de Pontuação.	23
Figura 23 – Tela da página de Versões da Tabela de Pontuação.	23
Figura 24 – Tela da página de Tipo de Atividade da Versão.	23
Figura 25 – Tela da página de cadastro de um Novo Tipo de Atividade da Versão.	24
Figura 26 – Tela para selecionar a versão de origem para a cópia.	24
Figura 27 – Classe, denominação, titulação, nível e pontuação de referência da car- reira do magistério superior para docentes no regime de dedicação ex- clusiva.	26
Figura 28 – Campos de data da atividade antes do preenchimento.	35
Figura 29 – Campos de data da atividade ao inserir um valor em Data de Início.	36
Figura 30 – Campos de data da atividade ao alterar Data de Início substituindo automaticamente o valor de Data Final.	36

Figura 31 – Campos de data da atividade ao alterar Data de Início mantendo o valor já presente em Data Final	37
Figura 32 – Atividade com data inicial maior que data final	37
Figura 33 – Atividade com pontuação negativa	37
Figura 34 – Validação que impede salvar atividade com data inicial maior que data final.	38
Figura 35 – Campo de descrição no formulário de cadastro de atividade.	38
Figura 36 – Descrição sem quebra de linhas na lista de atividades.	39
Figura 37 – Descrição com quebra de linhas na lista de atividades.	40
Figura 38 – Falha ao buscar pelo termo “palestra” na lista de atividades.	41
Figura 39 – Sucesso ao buscar pelo termo “palestra” na lista de atividades.	42
Figura 40 – Seleccionando data pelo <i>datepicker</i>	43
Figura 41 – Data com formatação inválida.	43
Figura 42 – Data com formatação válida usando máscara.	44
Figura 43 – Data inválida impedindo salvar formulário.	44
Figura 44 – Data válida permitindo salvar formulário.	45
Figura 45 – Modal de confirmação ao encerrar relatório.	45
Figura 46 – Nova modal de alerta ao encerrar relatório.	46
Figura 47 – Busca com acento.	47
Figura 48 – Falha ao buscar sem acento.	47
Figura 49 – Sucesso ao buscar sem acento.	48
Figura 50 – Carregamento inicial de 10 atividades na página.	48
Figura 51 – Caixa de seleção de quantidade de itens a serem exibidos na página. . .	49
Figura 52 – Até 20 itens listados na página.	49
Figura 53 – Página 2 da Lista de atividades.	51

Lista de abreviaturas e siglas

API	Interface de Programação de Aplicações (<i>Application Programming Interface</i>)
CLI	Interface de Linha de Comando (<i>Command-Line Interface</i>)
CONDIR	Conselho Diretor
CSS	Folhas de Estilo em Cascata (<i>Cascading Style Sheets</i>)
FACOM	Faculdade de Computação
HTML	Linguagem de Marcação de HiperTexto (<i>Hypertext Markup Language</i>)
JS	JavaScript
LINQ	Consulta Integrada à Linguagem (<i>Language-Integrated Query</i>)
ORM	Mapeamento Objeto-Relacional (<i>Object-Relational Mapping</i>)
SCAD	Sistema de Cadastro de Atividades Docente
SEI	Sistema Eletrônico de Informações
SQL	Linguagem de Consulta Estruturada (<i>Structured Query Language</i>)
TS	TypeScript
UFU	Universidade Federal de Uberlândia

Sumário

1	INTRODUÇÃO	8
1.1	Objetivos	9
1.1.1	Objetivos Específicos	9
1.2	Justificativa	9
1.3	Organização do Trabalho	9
2	REFERENCIAL TEÓRICO	10
2.1	SCAD	10
2.1.1	Perfil Pessoal	10
2.1.2	Perfil Avaliador	16
2.1.3	Perfil Administrador	19
2.2	Resolução nº 03/2017 - CONDIR - Progressão e promoção docente	25
2.3	Tecnologias	26
2.3.1	.NET	26
2.3.2	C#	27
2.3.3	PostgreSQL	27
2.3.4	Entity Framework	28
2.3.5	LINQ	28
2.3.6	Angular	29
2.3.7	Bootstrap	30
3	TRABALHOS CORRELATOS	31
4	DESENVOLVIMENTO	34
4.1	Método	34
4.1.1	Especificação de Requisitos	34
4.1.1.1	Requisitos Corretivos	34
4.1.1.2	Requisitos Evolutivos	35
4.2	Implementação dos Requisitos	35
4.2.1	Ajustar preenchimento automático nos campos de data final das atividades	35
4.2.1.1	Resolução	36
4.2.2	Validar consistência das datas cadastradas nas atividades	37
4.2.2.1	Resolução	38
4.2.3	Manter apresentação de quebra de linha na descrição das atividades na listagem	38
4.2.3.1	Resolução	39

4.2.4	Corrigir mecanismo de busca	40
4.2.4.1	Resolução	41
4.2.5	Adicionar máscara nos campos de data	42
4.2.5.1	Resolução	43
4.2.6	Remover limitação de valor máximo no campo de data final das atividades .	44
4.2.6.1	Resolução	44
4.2.7	Melhorar modal de encerramento de relatório	45
4.2.7.1	Resolução	46
4.2.8	Desconsiderar acentuação nas buscas	46
4.2.8.1	Resolução	47
4.2.9	Persistir a quantidade de itens por página exibidos na lista de atividades .	48
4.2.9.1	Resolução	49
4.2.10	Persistir a página selecionada na lista de atividades	50
4.2.10.1	Resolução	51
5	CONCLUSÃO	52
	REFERÊNCIAS	54

1 Introdução

A progressão de carreira é um elemento que valoriza o trabalho dos professores, incentivando não apenas a continuidade, mas também o aprimoramento acadêmico e profissional. Conforme destacado por [Bauer, Cassettari e Oliveira \(2017\)](#), a progressão se faz importante para manter os docentes motivados e comprometidos com a melhoria contínua da educação. Com isso, a Universidade Federal de Uberlândia (UFU) estabelece, por meio da Resolução Nº 03/2017 ([CONDUR \(UFU\), 2017](#)), um regulamento específico para a avaliação docente, abordando aspectos relacionados à progressão, promoção e aceleração da promoção nas carreiras de magistério superior e de ensino básico, técnico e tecnológico, por meio da avaliação de desempenho.

Essa progressão e promoção é realizada, na UFU, por meio do Sistema Eletrônico de Informações (SEI), que gerencia todos os processos da Universidade ([ALVES, 2022](#)). Contudo, apesar do processo ser realizado em um ambiente eletrônico, ainda era um processo muito manual e despadronizado, em que cada professor utiliza diferentes ferramentas para seus relatórios ([SILVA, 2019](#)). Pensando em automatizar e padronizar esse processo, a Faculdade de Computação (FACOM) utiliza o Sistema de Cadastro de Atividades Docente (SCAD) ([BRAGA, 2019](#)). Este sistema foi desenvolvido com o objetivo de auxiliar os docentes na elaboração dos relatórios das atividades realizadas para fins de progressão e promoção de carreira.

O SCAD permite o registro das atividades e da documentação pertinente, além de possibilitar o acompanhamento do andamento do processo, ter sua pontuação calculada automaticamente e garantir que as informações estejam sempre atualizadas. O sistema também conta com uma área específica para os avaliadores (os docentes responsáveis pela análise dos relatórios submetidos), oferecendo ferramentas para a avaliação e correção da documentação enviada pelos professores.

Vários estudos têm investigado como a tecnologia da informação melhora os processos administrativos, ressaltando vantagens como a diminuição dos custos operacionais, a economia de tempo e a melhoria na tomada de decisões. A automatização possibilita que tarefas anteriormente feitas manualmente sejam realizadas de maneira mais ágil e com menos falhas ([PESSOA; MELO, 2024](#)). Indo ao encontro dos autores, sistemas como o SCAD, minimizam o risco de erros manuais, oferecendo uma plataforma unificada onde atividades possam ser cadastradas de maneira padronizada. Isso facilita a inclusão dessas atividades nos relatórios necessários para a progressão de carreira, tornando o processo mais ágil e preciso.

1.1 Objetivos

Este projeto tem como objetivo corrigir inconsistências previamente identificadas e implementar novas funcionalidades demandadas pelos usuários.

1.1.1 Objetivos Específicos

- Corrigir problemas existentes, aprimorando as funcionalidades já implementadas.
- Implementar novas funcionalidades identificadas como necessárias pelos docentes.

1.2 Justificativa

Como destacado por [Pressman \(2010\)](#), a manutenção de *software* é essencial para corrigir defeitos, adaptar o sistema para atender um ambiente em constante mudança e melhorar suas funcionalidades para atender às necessidades dos usuários. Apesar de estar em funcionamento desde 2019, o SCAD se beneficia de processos contínuos de manutenção.

Com o uso contínuo da plataforma, surge a necessidade de aprimorá-la com base nas sugestões dos próprios docentes, que identificaram erros a serem corrigidos e melhorias a serem implementadas. Esse processo de atualização visa não apenas solucionar problemas técnicos, mas também garantir que o sistema evolua para atender de forma mais eficiente as demandas institucionais.

1.3 Organização do Trabalho

Este trabalho está estruturado em cinco capítulos, começando com a introdução (Capítulo 1), que apresenta os objetivos do estudo, a justificativa para a escolha do tema e a organização geral do conteúdo. O Capítulo 2 aborda o referencial teórico, discutindo algumas normas do CONDIR relacionadas à progressão e promoção na carreira docente e as tecnologias utilizadas no desenvolvimento e manutenção do SCAD. No Capítulo 3, são apresentados os trabalhos correlatos, que discutem outros estudos que também trabalharam no aprimoramento do SCAD, oferecendo uma visão das funcionalidades já implementadas.

O Capítulo 4 descreve o desenvolvimento do trabalho, abordando a implementação das melhorias corretivas e evolutivas no SCAD, a metodologia utilizada, as ações realizadas e os resultados obtidos. Por fim, o Capítulo 5 apresenta as conclusões, que sintetizam as contribuições do trabalho, discutem suas limitações e propõem sugestões para futuras melhorias.

2 Referencial Teórico

Neste capítulo, serão apresentados os conceitos e bases teóricas que sustentam este trabalho. Inicialmente, será mostrado o sistema SCAD. Em seguida, será discutida, brevemente, a Resolução nº 03/2017 - CONDIR, que regulamenta a progressão e promoção docente, fornecendo o contexto institucional necessário para compreender as demandas dos docentes ao utilizarem o SCAD. Em seguida, serão apresentados os conceitos das principais tecnologias utilizadas na manutenção do SCAD. *No back-end*, destacam-se o C#, .NET, PostgreSQL, Entity Framework e o LINQ. Já no *front-end*, o foco recai sobre o Angular e o Bootstrap. O objetivo é oferecer uma visão geral sobre os recursos empregados para aprimorar o sistema.

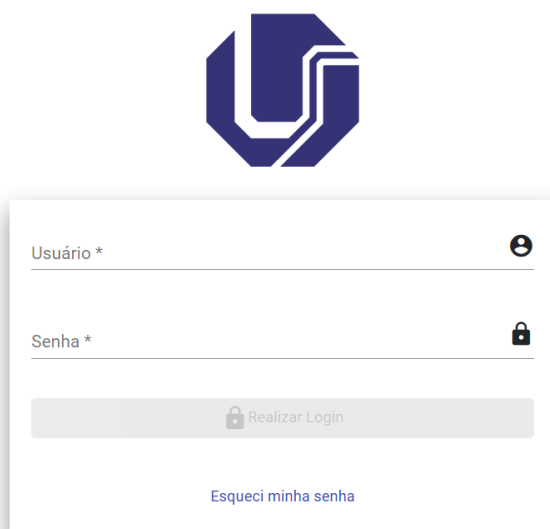
2.1 SCAD

O Sistema de Cadastro de Atividades Docente é uma plataforma utilizada pela FACOM para auxiliar processos administrativos relacionados à progressão de carreira docente. Ele facilita a gestão dos relatórios das atividades acadêmicas, garantindo mais agilidade e precisão nas tarefas relacionadas ao acompanhamento da carreira dos professores. O sistema possui três perfis: Pessoal, Avaliador, Administrador. Os três perfis serão descritos nas subseções subsequentes.

2.1.1 Perfil Pessoal

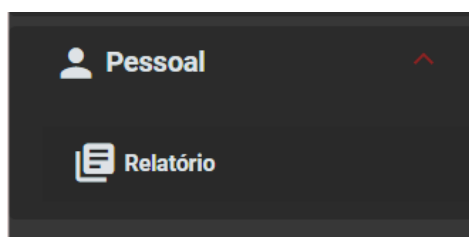
O perfil Pessoal é destinado ao professor da FACOM, que utiliza o SCAD para cadastrar suas atividades, acompanhá-las e gerar relatórios. Para acessar o sistema, é necessário possuir um login e senha criados pelo administrador do sistema, conforme ilustrado na Figura 1. Após autenticar-se, o professor tem acesso ao menu Pessoal (Figura 2), onde é possível, em Relatórios, criar ou gerenciar os relatórios, além de pesquisar – que está disponível em todas as páginas com listagens – e acessar a lista de atividades (Figura 3).

Figura 1 – Tela de Login.



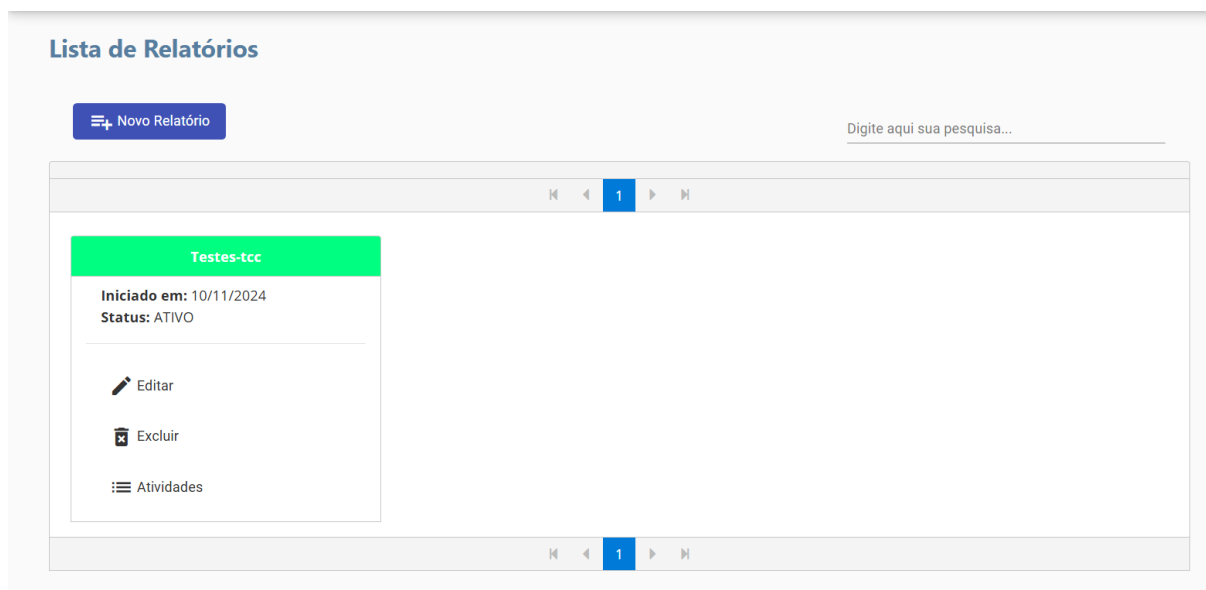
Fonte: Imagem do SCAD.

Figura 2 – Menu Pessoal.



Fonte: Imagem do SCAD.

Figura 3 – Tela da página Lista de Relatórios.



Fonte: Imagem do SCAD.

No item Novo Relatório, o professor pode cadastrar um novo relatório para a promoção/progressão de carreira. O sistema utiliza dados previamente cadastrados, permitindo que o professor selecione opções de forma progressiva, com filtros que ajustam automaticamente as escolhas subsequentes. Após essa seleção, os dados filtrados são exibidos para confirmação (Figura 4). Em seguida, o professor completa as informações do relatório, selecionando a versão da tabela de pontuação, informando o nome do relatório e definindo a data de início e término do interstício (Figura 5).

Figura 4 – Tela do formulário de Próximo Nível.

The screenshot shows a web form titled "Novo Relatório" with a tab labeled "Próximo Nível". The form contains several dropdown menus: "Regime *" with the selected value "Superior - Dedicção exclusiva e 40 horas", "Classe *" with "A", "Denominação *" with "Adjunto A", "Titulação *" with "D", and "Nível *" with "1". A "Filtrar" button is located on the right. Below the form, a section titled "Selecione a opção:" lists the selected values: "Regime: Superior - Dedicção exclusiva e 40 horas", "Classe: A", "Denominação: Adjunto A" (which is highlighted with a red circle), "Titulação: D", "Nível: 1", and "Pontuação de referência:".

Fonte: Imagem do SCAD.

Figura 5 – Tela do formulário das Informações do Relatório.

The screenshot shows a web form titled "Informações do Relatório". It features a dropdown menu for "Versão da Tabela de Pontuação *" with the selected value "1.00 - Resolução 03/2017". Below this is a text input field for "Nome do Relatório *". At the bottom, there are two date input fields: "Data Início de interstício *" and "Data Final de interstício *", each with a calendar icon. Navigation buttons "Voltar" (with a left arrow) and "Salvar" (with a save icon) are located at the bottom left and right respectively.

Fonte: Imagem do SCAD.

No botão Lista de Atividades, o professor pode realizar diversas ações, como criar uma nova atividade, consultar o resumo de pontuação, gerar relatórios, encerrar um relatório, pesquisar atividades ou visualizar a lista completa delas (Figura 6), podendo editar, excluir e verificar pendências na aba Ações (Figura 7). Ao selecionar Nova Atividade, o professor insere os detalhes da atividade (Figura 8), como o tipo (escolhido em uma lista). Após a seleção, o sistema exibe as exigências e a pontuação da atividade nas observações. Além disso, o professor pode incluir uma descrição, duração, anexar um novo comprovante da atividade ou optar por reutilizar um documento já existente.

Figura 6 – Tela da página Lista de Atividades.

Lista de atividades

Relatório iniciado em: 10/11/2024

Digite aqui sua pesquisa...

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
A1.2 - ATIVIDADES DE ORIENTAÇÃO	21 - Orientação de Tese de Doutorado / Supervisão de estágio de Pós-Doutorado. 4 ponto(s) Por aluno ou estagiário/mês completo	Início: 29/11/2024 Fim: 29/11/2024 0 mes(es)	0 ponto(s)	  
A1.3 - PRODUÇÃO INTELCTUAL	35 - Publicação de material hipermediático (texto, áudio e/ou vídeo) sobre divulgação científica em mídias variadas para o público. 15 ponto(s) Por publicação	Início: 28/11/2024 Fim: 28/11/2024	15 ponto(s)	  

Fonte: Imagem do SCAD.

Figura 7 – Tela das Pendências.

Atenção:

- Comprovante não anexado.

X Fechar

A1.1 - ATIVIDADE DE ENSINO	4 - Assessoramento à classe comum (Decreto 5671/2008); com atendimento junto ao professor e ao aluno na classe comum, realizada por docentes da área da educação especial da Unidade Especial de Ensino no turno e/ou contra-turno de aula. 1 ponto(s)	Início: 29/11/2024 Fim: 29/11/2024	1 ponto(s)	  
A1.1 - ATIVIDADE DE ENSINO	4 - Assessoramento à classe comum (Decreto 5671/2008); com atendimento junto ao professor e ao aluno na classe comum, realizada por docentes da área da educação especial da Unidade Especial de Ensino no turno e/ou contra-turno de aula. 1 ponto(s) Por hora-aula ministrada	Início: 28/11/2024 Fim: 28/11/2024	1 ponto(s)	  

Fonte: Imagem do SCAD.

Figura 8 – Tela da página Nova Atividade.

The screenshot shows a web form titled "Nova atividade". At the top, there is a label "Tipo de atividade *" followed by a text input field containing "2 - Oficina Psicoeducacional ministrada em modalidade de aula para alunos com deficiência e demais discentes da educação básica, profissional ou em" and a close icon (X). Below this is a box containing the text "Obs.: Comprovação conforme especificações 1 e 2 do item 1." and "1 ponto(s) Por hora-aula ministrada". Further down is a "Descrição" label followed by a large text area with a character count "0 / 500". Below the description are three input fields: "Quantidade", "Data de Início *" with a calendar icon, and "Data Final *" with a calendar icon. At the bottom, there is a "Reuso de Documento" label, a dropdown menu with a close icon (X), a button labeled "Escolher Arquivo", and the text "Nenhum arquivo escolhido".

Fonte: Imagem do SCAD.

Na opção Resumo da Pontuação, o professor visualiza uma lista de atividades realizadas com suas respectivas pontuações, incluindo o total acumulado (Figura 9). Já em Gerar Relatório, o relatório final é gerado e baixado em formato HTML, assim como os comprovantes em pdf (Figura 10).

Figura 9 – Tela do Resumo de Pontuação.

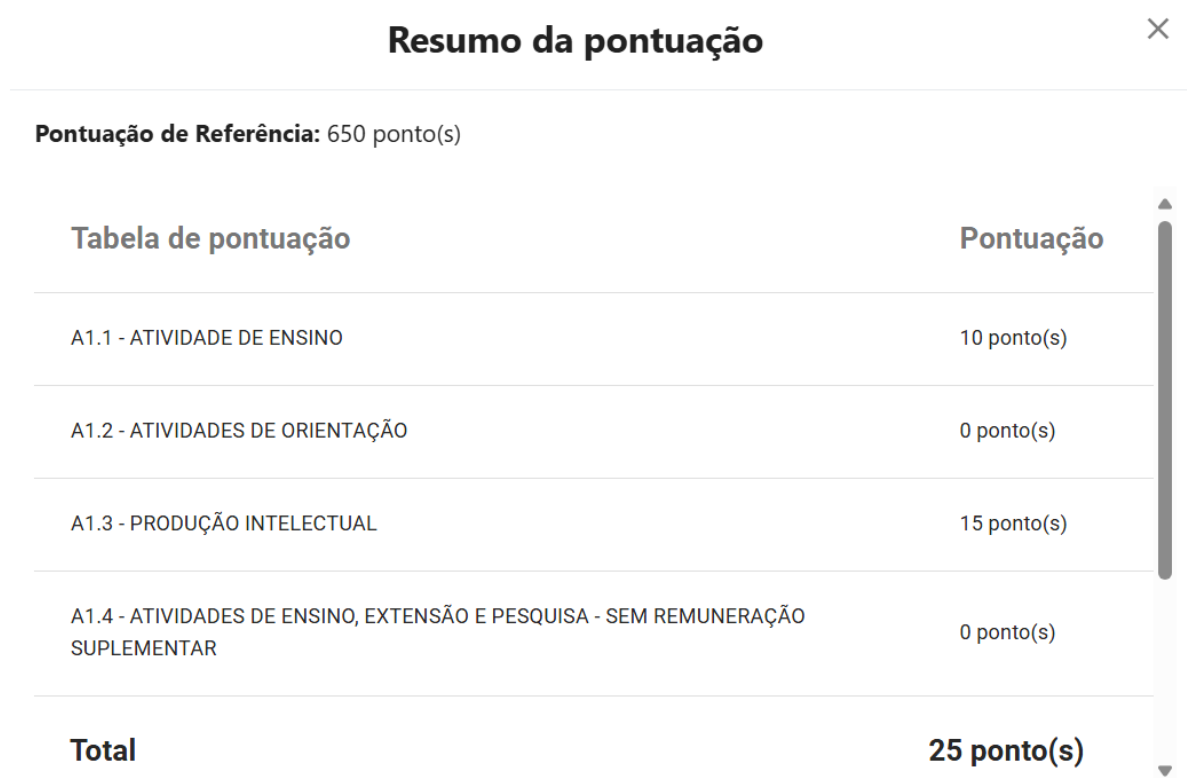
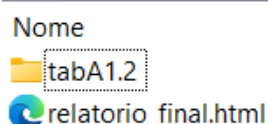


Tabela de pontuação	Pontuação
A1.1 - ATIVIDADE DE ENSINO	10 ponto(s)
A1.2 - ATIVIDADES DE ORIENTAÇÃO	0 ponto(s)
A1.3 - PRODUÇÃO INTELECTUAL	15 ponto(s)
A1.4 - ATIVIDADES DE ENSINO, EXTENSÃO E PESQUISA - SEM REMUNERAÇÃO SUPLEMENTAR	0 ponto(s)
Total	25 ponto(s)

Fonte: Imagem do SCAD.

Figura 10 – Tela do Resumo de Pontuação.

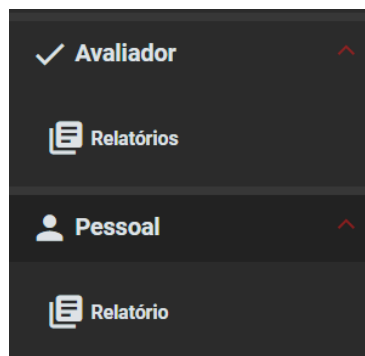


Fonte: Imagem do SCAD.

2.1.2 Perfil Avaliador

O Perfil Avaliador é responsável por conferir a consistência dos documentos e relatórios submetidos, além de contatar os professores em caso de necessidade. Após realizar o login, o avaliador acessa o menu Avaliador (Figura 11), onde pode navegar pelos relatórios por meio das abas disponíveis no menu superior. Essas abas incluem: Relatórios em Andamento, Relatórios Encerrados, Relatórios em Avaliação e Relatórios Avaliados, todas apresentando listas detalhadas de cada categoria (Figura 12).

Figura 11 – Menu Avaliador.



Fonte: Imagem do SCAD.

Figura 12 – Tela da aba Relatórios em Andamento.

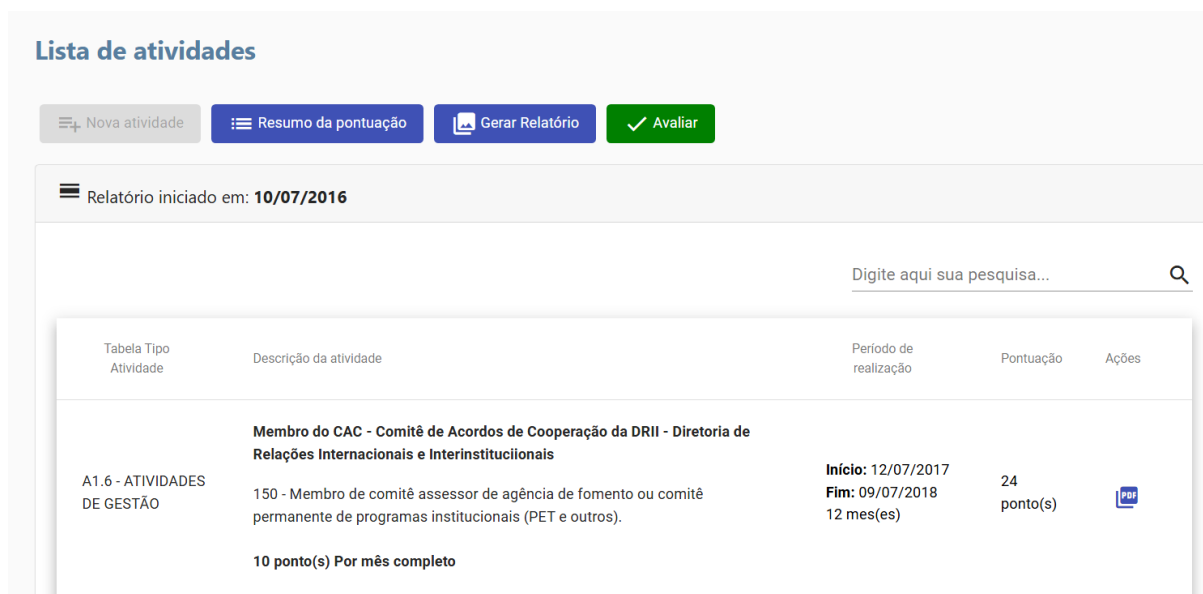
Relatórios em andamento Relatórios encerrados Relatórios em avaliação Relatórios avaliados			
Digite aqui sua pesquisa... 			
Nome do Relatório	Professor	Data Início/Fim interstício	Atividades
Relatório para progressão de C1 para C2	João das Neves	Início: 01/12/2010 Fim: 30/07/2022	
Relatório de progressão adjunto	Mari Maria	Início: 01/01/2018 Fim: 31/12/2019	

Fonte: Imagem do SCAD.

Na aba Relatórios Encerrados, ao clicar no botão de ação localizado no canto direito de um item, é aberta uma nova janela que exibe as atividades associadas ao relatório (Figura 13). Nessa janela, o avaliador pode:

- Visualizar a lista de atividades do relatório;
- Gerar o comprovante em PDF;
- Consultar o resumo da pontuação;
- Gerar o relatório final;
- Iniciar o processo de avaliação por meio do botão Avaliar.

Figura 13 – Tela da aba Relatórios Encerrados.

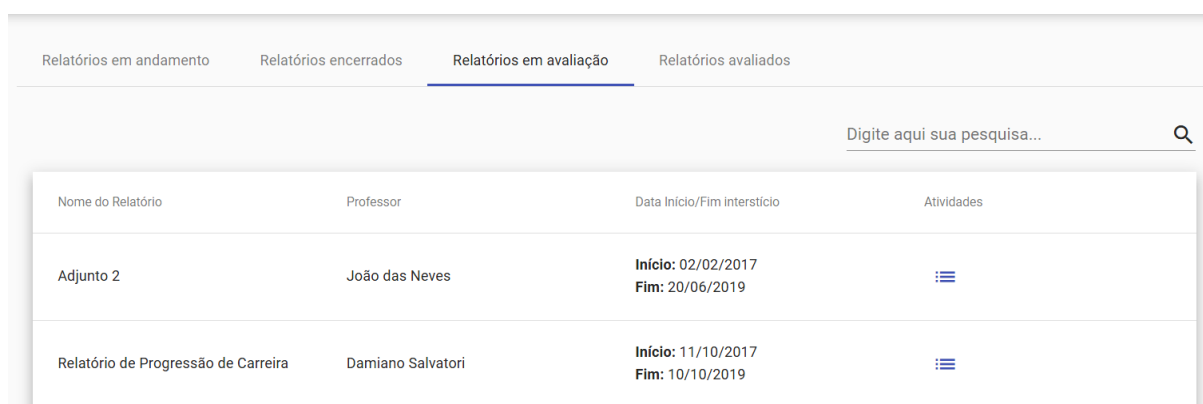


Fonte: Imagem do SCAD.

Na aba Relatórios em Avaliação (Figura 14), o avaliador encontra a lista de relatórios que estão atualmente em processo de avaliação. Ao clicar no ícone de ação relacionado às atividades de um relatório, a página de detalhes do relatório é exibida (Figura 15). Nessa página, o avaliador pode:

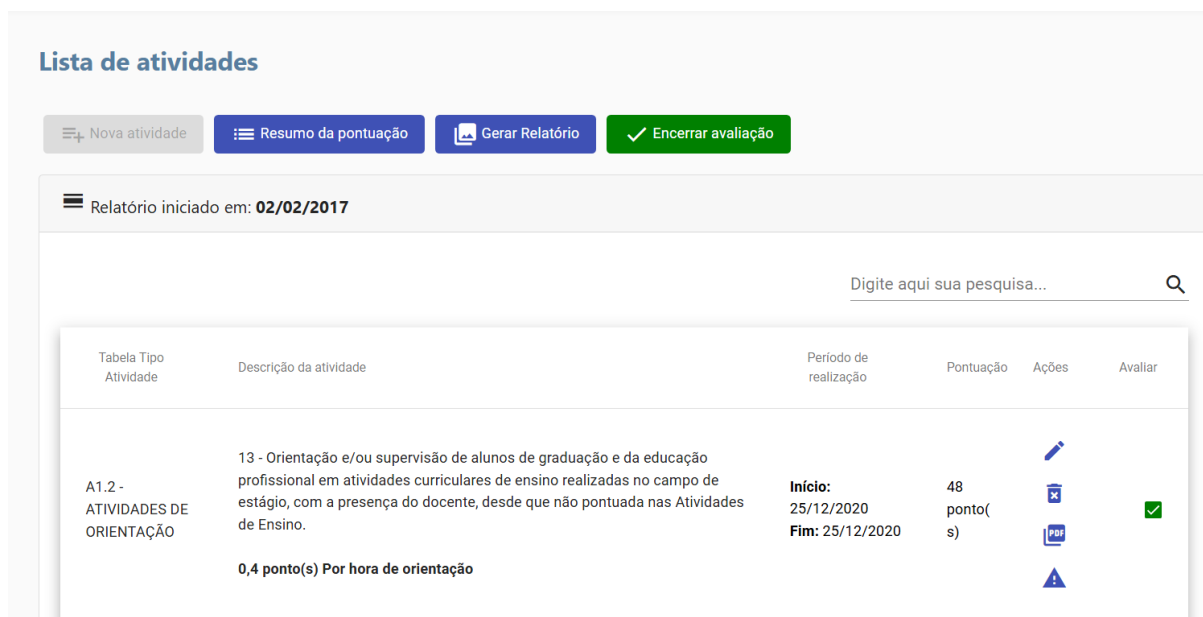
Gerenciar as atividades; Conferir as informações das atividades; Marcar como verificadas por meio do botão Check após confirmar que estão corretas. Ao concluir a análise de todas as atividades, o avaliador utiliza o botão Encerrar Avaliação para finalizar o processo.

Figura 14 – Tela da aba Relatórios em Avaliação.



Fonte: Imagem do SCAD.

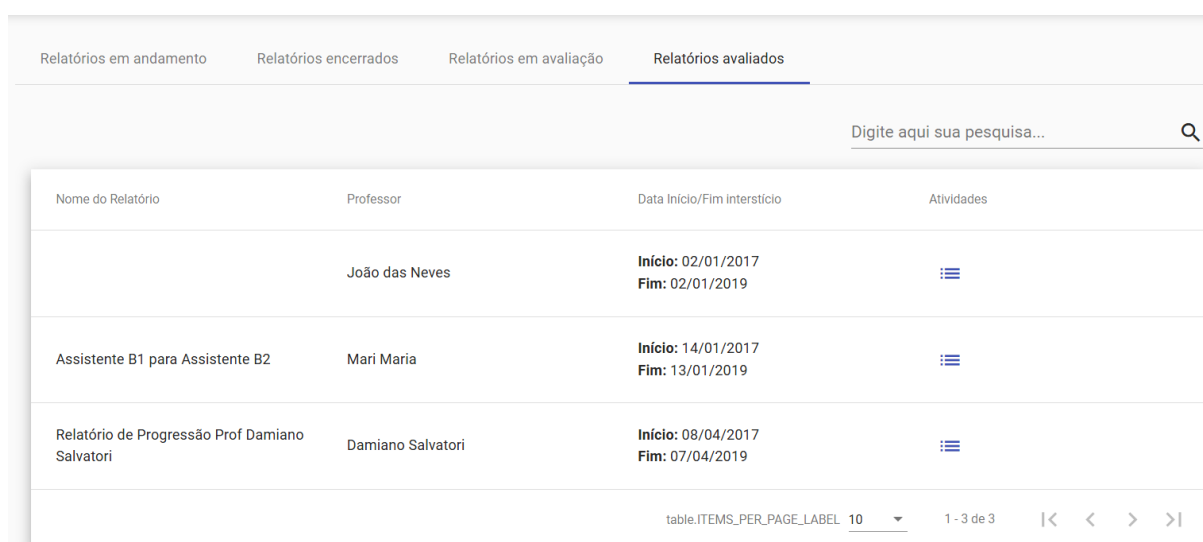
Figura 15 – Tela da página de lista de atividades para avaliação.



Fonte: Imagem do SCAD.

Na aba Relatórios Avaliados (Figura 16), é apresentada uma lista de relatórios já avaliados pelo professor avaliador. Para cada relatório, é possível visualizar os detalhes das atividades por meio do ícone de ação correspondente.

Figura 16 – Tela da aba Relatórios Avaliados.



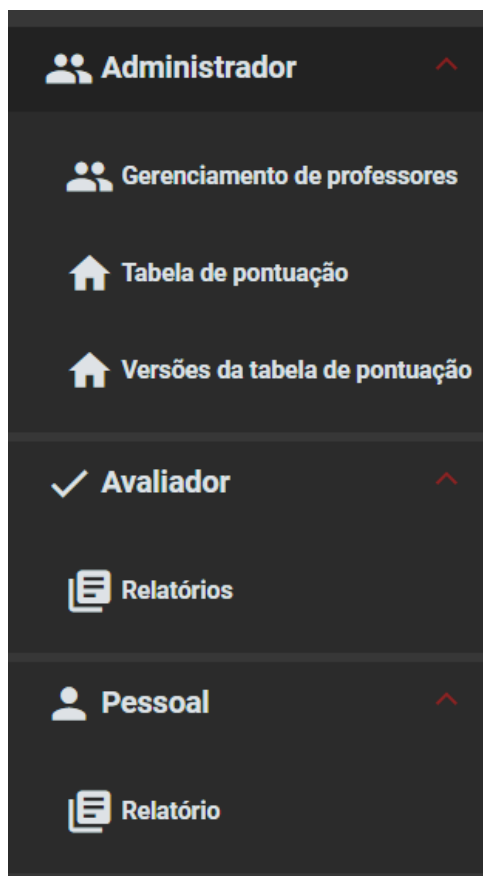
Fonte: Imagem do SCAD.

2.1.3 Perfil Administrador

O Perfil Administrador permite ao professor realizar o gerenciamento de usuários, dados e configurações do sistema, incluindo o cadastro de pontuações e a criação de versões

da tabela vigente. Assim como os demais usuários, após realizar o login, o administrador acessa seu menu principal (Figura 17).

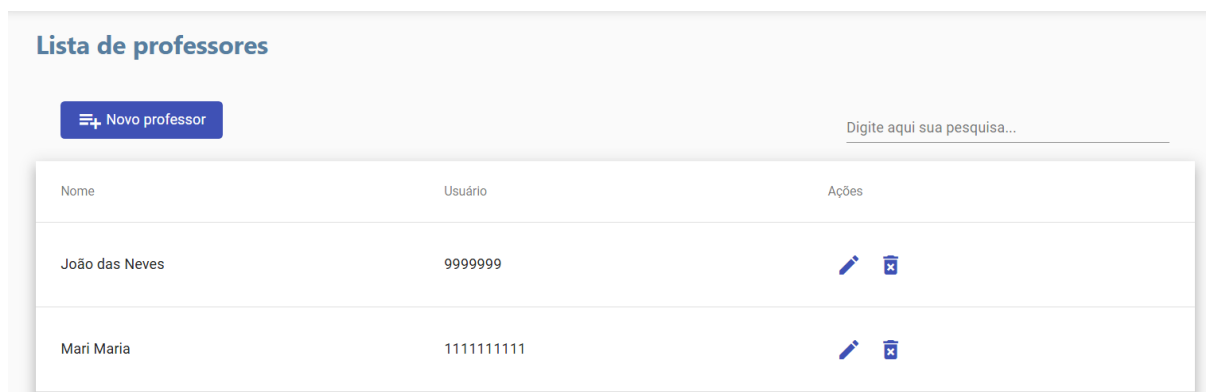
Figura 17 – Menu Administrador.



Fonte: Imagem do SCAD.

Ao clicar em Gerenciamento de Professores, o administrador é redirecionado para uma página onde pode cadastrar novos professores, visualizar a lista dos já cadastrados e gerenciar suas informações (Figura 18). Na opção Novo Professor, ele informa o nível atual do docente, seleciona opções pré-cadastradas, cria o login e senha (Figura 19) e insere os dados pessoais necessários. Além disso, é possível configurar as permissões de acesso do professor (Figura 20).

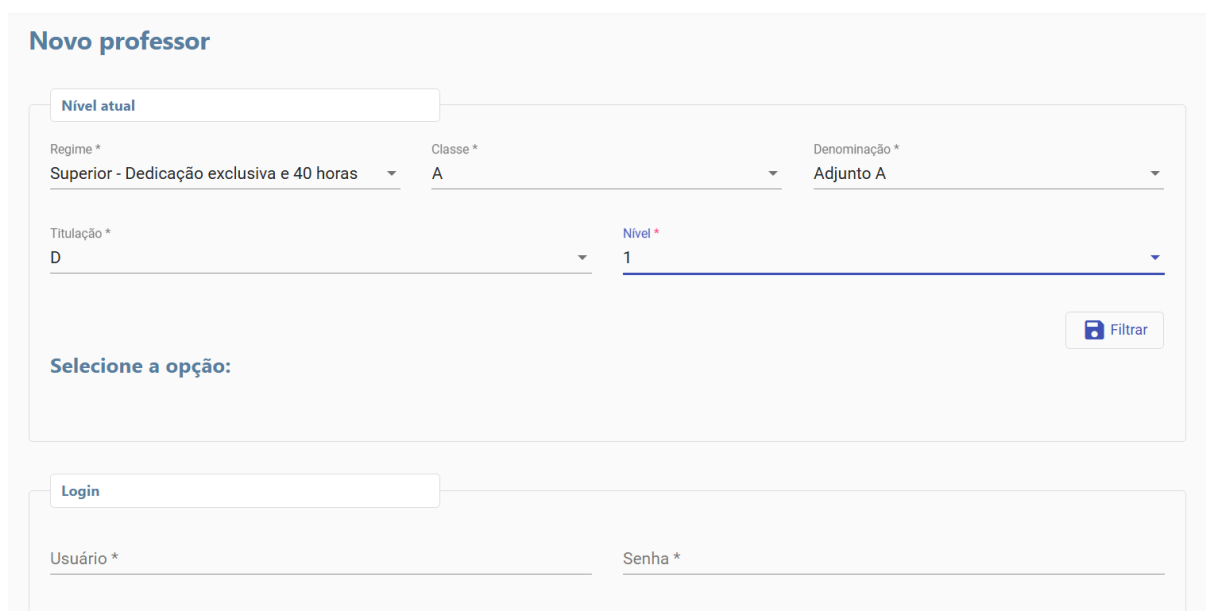
Figura 18 – Tela da página Gerenciamento de Professores.



Nome	Usuário	Ações
João das Neves	9999999	 
Mari Maria	1111111111	 

Fonte: Imagem do SCAD.

Figura 19 – Tela dos formulários de Nível Atual e Login da página Cadastro de Novo Professor.



Novo professor

Nível atual

Regime *
Superior - Dedicação exclusiva e 40 horas

Classe *
A

Denominação *
Adjunto A

Titulação *
D

Nível *
1

Selecione a opção:

Login

Usuário *
Senha *

Fonte: Imagem do SCAD.

Figura 20 – Tela dos formulários das Informações Gerais do Professor e Permissões da página Cadastro de Novo Professor.

Informações gerais do professor

Siape * Nome *

Carga atual Lotação Status

Data ingresso Data nascimento Data exoneração

Data aposentadoria Data saída

Permissões

☐ Afastado ☐ Administrativo ☐ Avaliador

[Voltar](#) [Salvar](#)

Fonte: Imagem do SCAD.

Na seção Tabela de Pontuação (Figura 21), o administrador visualiza a lista de atividades cadastradas, identificadas por código e nome. Ele pode editar as atividades existentes ou criar uma nova tabela, definindo seu código e nome (Figura 22).

Figura 21 – Tela da página Tabela de Pontuação.

Tabelas de Pontuação

[Nova tabela de pontuação](#)

Código	Nome	Ações
A1.2	ATIVIDADES DE ORIENTAÇÃO	✎ ✕
A1.3	PRODUÇÃO INTELECTUAL	✎ ✕

Fonte: Imagem do SCAD.

Figura 22 – Tela da página do cadastro de uma Nova Tabela de Pontuação.

Nova tabela de pontuação

Código * _____ Nome * _____

[← Voltar](#) [Salvar](#)

Fonte: Imagem do SCAD.

Em Versões da Tabela de Pontuação (Figura 23), o administrador tem acesso à lista de versões cadastradas, podendo editar ou criar uma nova versão. Para cada nova versão, ele define um código e um título. Ao clicar no ícone de ação de uma versão, o administrador é redirecionado para a página que detalha os Tipos de Atividade associados a ela (Figura 24).

Figura 23 – Tela da página de Versões da Tabela de Pontuação.

Versões da tabela de pontuação

[+ Nova versão da tabela de pontuação](#)

Código	Título	Ações
1.00	Resolução 03/2017	⋮ ✎ 🗑

Itens por página: 10 1 - 1 de 1 [|<](#) [<](#) [>](#) [>|](#)

Fonte: Imagem do SCAD.

Figura 24 – Tela da página de Tipo de Atividade da Versão.

Tipo de atividades

[+ Novo tipo de atividade](#) [Copiar versão anterior](#)

Código	Nome	Tabela de Pontuação	Ações
1	Aula teórica ou prática de componentes curriculares ministrados na modalidade presencial ou a distância, na Educação Básica, na Educação Profissional, em cursos de graduação ou pós-graduação stricto e lato sensu da UFU, aprovadas pelo Conselho da Unidade.	A1.1	✎ 🗑
2	Oficina Psicoeducacional ministrada em modalidade de aula para alunos com deficiência e demais discentes da educação básica, profissional ou em cursos de graduação, aprovada pelo Conselho da Unidade.	A1.1	✎ 🗑

Fonte: Imagem do SCAD.

Na página de tipos de atividade, o administrador pode criar um novo tipo de atividade (Figura 25) indicando informações como a tabela correspondente, código, quantidade de pontos, unidade (como por ação, obra, mês, etc.), nome, descrição e as quantidades mínima e máxima permitidas. Ele também pode editar os tipos de atividade já cadastrados. Além disso, existe a opção de copiar uma versão anterior da tabela, indicando a versão de origem a ser replicada (Figura 26).

Figura 25 – Tela da página de cadastro de um Novo Tipo de Atividade da Versão.

O formulário, intitulado "Novo tipo de atividade", contém os seguintes campos:

- Tabela de Pontuação ***: Campo de seleção com uma seta para baixo.
- Código ***: Campo de texto.
- Quantidade de pontos**: Campo de texto.
- Unidade ***: Campo de seleção com uma seta para baixo.
- Nome ***: Campo de texto com limite de 500 caracteres (0 / 500).
- Descrição ***: Campo de texto com limite de 1000 caracteres (0 / 1000).
- Quantidade mínima**: Campo de texto.
- Teto**: Campo de texto.

Na base do formulário, há dois botões: "Voltar" (com uma seta para trás) e "Salvar" (com um ícone de disco).

Fonte: Imagem do SCAD.

Figura 26 – Tela para selecionar a versão de origem para a cópia.

A tela apresenta o título "Selecione a versão de origem" e um campo de seleção rotulado "Versão da Tabela de Pontuação *".

Na base da tela, há dois botões: "Copiar" (com um ícone de disco) e "Fechar" (com um ícone de X).

Fonte: Imagem do SCAD.

2.2 Resolução nº 03/2017 - CONDIR - Progressão e promoção docente

A Resolução nº 03/2017 ([CONDIR \(UFU\), 2017](#)) estabelece as diretrizes para a progressão e promoção de docentes na Universidade, regulamentando os procedimentos que devem ser seguidos pelos professores para avançar em sua carreira. A norma é composta por 17 artigos, com diversos parágrafos e anexos, detalhando os critérios, prazos e documentação necessária para solicitar a avaliação de progressão. Esta seção oferece uma visão geral da resolução, focando nos pontos essenciais para a compreensão do processo de progressão e promoção docente.

Para a progressão e promoção na carreira docente, é preciso solicitar formalmente a avaliação à sua unidade acadêmica após cumprir o interstício de 24 meses — tempo necessário em um determinado nível ou classe antes de poder requerer a avaliação para avançar ou ser promovido. A solicitação deve ser acompanhada de um requerimento, um relatório de atividades docentes, uma declaração de avaliação de desempenho e a documentação comprobatória necessária. A avaliação será realizada por uma comissão composta por docentes da unidade e levará em consideração critérios como assiduidade, responsabilidade, qualidade do trabalho e a participação em atividades de ensino, extensão, pesquisa e gestão acadêmica. Para que seja protocolado, o docente precisa atingir a pontuação mínima correspondente à classe e ao nível que ocupa ([CONDIR \(UFU\), 2017](#)).

As classes da Carreira do Magistério Superior para Docentes são divididas em cinco classes, conforme ilustrado na Figura 27. Nessa mesma figura, estão indicados as denominações, as titulações (G: Graduado; E: Especialista; M: Mestre; D: Doutor), bem como os níveis e as pontuações mínimas exigidas para progressão ou promoção para a classe e nível correspondentes, no regime de dedicação exclusiva.

Figura 27 – Classe, denominação, titulação, nível e pontuação de referência da carreira do magistério superior para docentes no regime de dedicação exclusiva.

Classe	Denominação	Titulação	Nível			
			I	II	III	IV
A	Auxiliar	G, A ou E	-	600	-	-
	Assistente A	M	-	610	-	-
	Adjunto A	D	-	630	-	-
B	Assistente	G, A ou E	620	630	-	-
	Assistente	M	630	650	-	-
	Assistente	D	650	670	-	-
C	Adjunto	G, A e E	640	650	660	670
	Adjunto	M	660	680	700	720
	Adjunto	D	700	730	760	790
D	Associado	D	840	880	920	960
E	Titular	D	1000			

Fonte: ([CONDIR \(UFU\), 2017](#)).

2.3 Tecnologias

Este trabalho dá continuidade ao uso das tecnologias que fundamentam o SCAD, adotadas nos trabalhos anteriores sobre o sistema. A manutenção corretiva e evolutiva proposta segue a mesma base tecnológica. Para este trabalho, o foco principal foi o aprimoramento da interface, priorizando melhorias no *front-end*. Nesse contexto, destacou-se o aprofundamento no uso do Angular, uma das tecnologias centrais, com o objetivo de melhorar a usabilidade e tornar a plataforma mais intuitiva.

2.3.1 .NET

O .NET é uma plataforma de desenvolvimento gratuita, de código aberto e multiplataforma criada pela Microsoft. Ela permite o desenvolvimento de diversos tipos de aplicações, como *web*, *desktop*, *mobile* e jogos. Um dos principais diferenciais do .NET é a sua capacidade de executar programas escritos em várias linguagens, sendo o C# a mais popular. A plataforma utiliza um *runtime* de alto desempenho, projetado para aplicativos que demandam alta escalabilidade e confiabilidade ([Microsoft, 2024c](#)).

Entre os recursos que tornam o .NET uma escolha poderosa estão, como explicitado na documentação, o gerenciamento automático de memória, realizado pelo coletor de lixo (*Garbage Collector*), e o suporte a programação assíncrona. A plataforma também prioriza

segurança, produtividade e desempenho, sendo *type-safe* (previne operações incompatíveis entre tipos de dados) e protegida contra problemas de memória. Além disso, seus princípios de design garantem:

- Produtividade: Integração de runtime, bibliotecas, linguagem e ferramentas, otimizando a experiência do desenvolvedor.
- Interoperabilidade: Suporte eficiente para integração com código nativo e intrínsecos de *hardware*.
- Portabilidade: Aplicações podem ser executadas em diferentes plataformas (como Windows, Linux e macOS).
- Flexibilidade: Adaptabilidade para diversos domínios, como nuvem, cliente e jogos.
- Adoção de padrões: Foco em padrões como OpenTelemetry - *framework* de código aberto que permite coletar dados de telemetria (métricas, *logs* e rastreamentos) para monitoramento e análise de desempenho -, ao invés de soluções personalizadas.

A plataforma é mantida pela Microsoft em colaboração com a comunidade, recebendo atualizações regulares para garantir segurança e confiabilidade em produção (Microsoft, 2024c).

2.3.2 C#

O C# é uma linguagem de programação multiplataforma e de uso geral, projetada para oferecer alta produtividade ao permitir a criação de código de alto desempenho. Com uma grande base de desenvolvedores, o C# é a linguagem principal do ecossistema .NET. Sua versatilidade é garantida pelo suporte amplo em diversas cargas de trabalho do .NET. A linguagem é baseada em princípios de programação orientada a objetos, contudo, o C# também incorpora muitos recursos da programação funcional, oferecendo uma flexibilidade considerável aos desenvolvedores (Microsoft, 2024a).

O C# também se beneficia do suporte a coletores de lixo, que gerenciam automaticamente a memória, otimizando o desempenho das aplicações. Essa integração entre a linguagem e o ambiente de execução permite que os desenvolvedores escrevam algoritmos de maneira produtiva e eficiente, utilizando fluxos de trabalho intuitivos que são moldados pelas bibliotecas do .NET (Microsoft, 2024c).

2.3.3 PostgreSQL

O PostgreSQL é um sistema de gerenciamento de banco de dados relacional e objeto-relacional de código aberto, desenvolvido originalmente no Departamento de Ciência da Computação da Universidade da Califórnia, em Berkeley. Ele surgiu a partir

do projeto POSTGRES e, ao longo do tempo, incorporou várias inovações que se tornaram comuns em sistemas de bancos de dados comerciais apenas anos depois ([PostgreSQL Global Development Group, 2024](#)).

O PostgreSQL é conhecido por sua capacidade de lidar com consultas complexas, chaves estrangeiras, gatilhos, visões atualizáveis, integridade transacional e controle de concorrência multiversão. Ele também oferece grande flexibilidade, permitindo aos usuários adicionar novos tipos de dados, funções, operadores, funções agregadas, métodos de índice e linguagens procedurais ([PostgreSQL Global Development Group, 2024](#)).

A documentação conta que, em 1996, o nome “Postgres95” foi alterado para PostgreSQL para refletir a adição de funcionalidades SQL (Structured Query Language) ao projeto original. O PostgreSQL é altamente extensível e pode ser utilizado, modificado e distribuído gratuitamente, sendo amplamente adotado em ambientes acadêmicos, comerciais e privados. Sua licença aberta facilita sua utilização e adaptação de acordo com as necessidades do usuário. Esse sistema se destaca por sua estabilidade, conformidade com os padrões SQL e pela possibilidade de personalização.

2.3.4 Entity Framework

O Entity Framework (EF) é uma ferramenta moderna de mapeamento objeto-relacional (ORM) que facilita o desenvolvimento de uma camada de acesso a dados limpa e eficiente em aplicações .NET, como no C#. Ele permite a comunicação com diversos bancos de dados, incluindo SQL Server, SQLite, MySQL, PostgreSQL, além de suportar o Azure Cosmos DB e bancos de dados no Azure. O EF oferece suporte a consultas LINQ, o que pode tornar a consulta de dados mais intuitiva, além de funcionalidades robustas para controle de alterações, gerenciamento de atualizações e migrações de esquema de banco de dados ([Microsoft, 2024b](#)).

Essa capacidade de abstração e reutilização é característica de um *framework*, que pode ser definido como um design reutilizável de um programa ou parte de um programa expresso como um conjunto de classes. Um *framework* fornece uma estrutura básica para o desenvolvimento de software, permitindo que os desenvolvedores se concentrem em resolver problemas específicos, enquanto reutilizam componentes prontos que garantem maior produtividade e consistência ([JOHNSON, 1992](#)). No caso do Entity Framework, ele abstrai detalhes complexos de acesso a dados, permitindo maior foco na lógica da aplicação.

2.3.5 LINQ

Uma consulta é uma expressão usada para buscar dados em uma fonte específica. Diferentes fontes, como bancos de dados relacionais e XML, utilizam suas próprias linguagens de consulta, como SQL e XQuery. O LINQ (Language-Integrated Query) resolve

esse problema ao permitir que os desenvolvedores usem C# para consultar diversos tipos de dados, como bancos de dados SQL, coleções .NET e documentos XML, de forma consistente, utilizando o mesmo modelo de consulta ([Microsoft, 2024d](#)).

O LINQ ainda oferece uma API de funções de ordem superior – funções que recebem ou retornam outras funções como parâmetros ou saída, como `where` e `select`. Essa abordagem proporciona flexibilidade ao escrever código mais declarativo C#, facilitando a manipulação de dados de diversas fontes de maneira consistente ([Microsoft, 2024e](#)).

2.3.6 Angular

O Angular é um framework de código aberto desenvolvido pelo Google, amplamente utilizado para criar aplicativos web dinâmicos e interativos. Ele é baseado em TypeScript (TS), uma extensão do JavaScript (JS) que adiciona tipagem estática e outros recursos avançados, tornando o desenvolvimento mais seguro e eficiente. O Angular adota uma abordagem orientada a componentes, na qual a aplicação é dividida em pequenas partes reutilizáveis que, juntas, formam a interface final. Cada componente é composto por três arquivos principais: um arquivo de estilo, geralmente com extensão CSS ou SCSS, que define a aparência; um arquivo HTML, que descreve a estrutura visual do componente; e um arquivo TypeScript, que contém a lógica de comportamento ([Angular Team, 2024b](#)).

Antes de começar a utilizar o Angular, é essencial ter conhecimentos prévios em HTML, CSS e JavaScript. O HTML (HyperText Markup Language) é a linguagem de marcação usada para estruturar o conteúdo das páginas da web. Ele define elementos como títulos, parágrafos, botões e links, sendo a base para qualquer aplicação web. Já o CSS (Cascading Style Sheets) é a linguagem de estilos que define a aparência visual desses elementos, permitindo configurar cores, tamanhos, fontes, espaçamentos e muito mais. Por sua vez, o JavaScript é uma linguagem de programação que adiciona interatividade às páginas, como validar formulários, atualizar conteúdos dinamicamente e responder a eventos do usuário, sendo uma peça-chave na construção de aplicativos web modernos ([Angular Team, 2024a](#)).

Além do conhecimento dessas tecnologias, é importante ter familiaridade com o TypeScript, já que ele é a linguagem padrão utilizada no Angular. O TypeScript é um superconjunto do JS, ou seja, é uma linguagem que estende o JS adicionando novos recursos, mas que ainda é totalmente compatível com o JS padrão. Isso facilita a detecção de erros durante o desenvolvimento e melhora a manutenção do código, tornando-o mais legível e robusto. Por padrão, a Interface de Linha de Comando (CLI) do Angular gera todos os arquivos de configuração necessários em TS, o que simplifica a inicialização de projetos e proporciona uma integração fluida com o framework ([GEETHA et al., 2022](#)).

Outro ponto de destaque no Angular é sua CLI, que automatiza diversas tarefas de desenvolvimento. Com a CLI, é possível criar rapidamente a estrutura de um projeto, incluindo pastas, arquivos de configuração e dependências necessárias. Ela também facilita o gerenciamento de componentes, serviços e módulos, além de simplificar a compilação, os testes e a implantação da aplicação. Ao criar um componente por meio da CLI do Angular, também é gerado um arquivo para testes unitários, incentivando boas práticas no desenvolvimento ([Angular Team, 2024b](#)).

2.3.7 Bootstrap

O Bootstrap é o framework de front-end mais popular e amplamente utilizada para o desenvolvimento de sites responsivos e compatíveis com dispositivos móveis. Totalmente gratuito e de código aberto, o Bootstrap facilita a criação de páginas web modernas e adaptáveis, tornando o desenvolvimento mais rápido e acessível. Ele inclui uma série de modelos de design baseados em HTML, CSS e JavaScript, que cobrem uma variedade de elementos essenciais para a construção de interfaces, como tipografia, formulários, botões, tabelas, navegação, modais, carrosséis de imagens, entre outros ([GAIKWAD; ADKAR, 2019](#)).

O Bootstrap oferece uma variedade de classes predefinidas que simplificam a criação de layouts e estilos. Ele também permite a integração de plug-ins JavaScript, que adicionam funcionalidades interativas como modais e carrosséis de imagens, melhorando a experiência do usuário ([Bootstrap, 2024](#)). Um de seus componentes mais utilizados é sistema de grades fluidas e responsivas, que se ajustam automaticamente ao tamanho da tela do dispositivo, permitindo que o layout se adapte ([GAIKWAD; ADKAR, 2019](#)).

3 Trabalhos correlatos

Esta seção apresenta trabalhos relacionados ao Sistema de Cadastro de Atividades Docente (SCAD), implementado na Universidade Federal de Uberlândia (UFU). Desde sua criação, o SCAD tem sido objeto de estudos que abordam diferentes aspectos de sua funcionalidade e impacto. Os trabalhos descritos a seguir investigam sua aplicação prática, limitações identificadas e melhorias propostas, fornecendo uma base sólida para o desenvolvimento de novas contribuições, como as abordadas neste trabalho.

Braga (2019) desenvolveu o Sistema de Cadastro de Atividades Docente (SCAD), com a finalidade de automatizar o processo de criação e avaliação de relatórios das atividades acadêmicas dos docentes da UFU. O SCAD visa otimizar e padronizar o fluxo de atividades, promovendo a eficiência na progressão de carreira dos docentes. Aqui, o SCAD utiliza tecnologias como Java EE, Spring MVC, Hibernate no *back-end*, e AngularJS, HTML, CSS no *front-end*, com PostgreSQL como banco de dados. A arquitetura adotada segue o padrão MVC (*Model-View-Controller*), que organiza a aplicação em camadas distintas, garantindo maior modularidade e facilidade de manutenção. O sistema é composto por diversos módulos, cada um com funções específicas para otimizar a gestão das atividades acadêmicas e a avaliação dos docentes.

O Módulo Principal centraliza o acesso dos usuários redirecionando-os para a Tela de Autenticação, onde inserem seu código SIAPE e senha. Após a autenticação, são direcionados para a Tela Inicial, que exibe os módulos disponíveis conforme o perfil do usuário, facilitando a navegação. O Módulo Administrativo é destinado aos gestores e permite o controle das informações dos docentes e parâmetros do sistema. Inclui a Gerência de Professores, onde é possível cadastrar, editar e remover docentes, e a Gerência de Versões da Tabela de Pontuação e Tipos de Atividade, que assegura uma avaliação padronizada e flexível.

Já o Módulo Pessoal foi criado para os docentes, permitindo a criação, edição e encerramento de relatórios, bem como o acompanhamento das pontuações das atividades realizadas. Também inclui a Gerência de Atividades, que facilita o cadastro e a edição das atividades acadêmicas. Por fim, o Módulo Avaliação é destinado aos avaliadores, permitindo o acesso aos relatórios para análise e correção. Ele organiza os relatórios em categorias de “a avaliar” e “avaliados” e possibilita a Correção de Atividades, com justificativas obrigatórias, garantindo um processo de avaliação transparente e documentado.

Silva (2019) apresentou uma versão atualizada do Sistema de Cadastro de Atividades Docente (SCAD), com o objetivo de modernizar o processo de produção de relatórios utilizados na progressão e promoção de carreira dos docentes. A nova versão foi desen-

volvida com tecnologias modernas para garantir maior desempenho, melhor manutenção e uma experiência mais eficiente. No *back-end*, foram utilizados C#, .NET e serviços web RESTful, enquanto o *front-end* foi implementado com Angular e Bootstrap. O banco de dados relacional PostgreSQL foi escolhido para gerenciar informações como dados dos professores, atividades realizadas, documentos comprobatórios e tabelas de pontuação.

A arquitetura do sistema seguiu o modelo DDD (*Domain Driven Design*), organizado em camadas que segmentam a lógica de aplicação, regras de negócio, serviços e persistência de dados. Entre as camadas principais estão: aplicação, domínio, serviço e infraestrutura. Essa estrutura garante modularidade e escalabilidade, facilitando a evolução do sistema.

A nova versão do sistema também incorporou requisitos adicionais. Entre as atualizações, destaca-se a inclusão de informações detalhadas sobre o próximo nível de progressão do docente, o avaliador responsável e as datas do interstício ao gerar relatórios. Além disso, o sistema passou a exibir o próximo nível do docente ao criar um novo relatório, facilitando o planejamento das atividades necessárias para a progressão. Outro aprimoramento significativo foi a adição de telas específicas para cadastro e visualização das tabelas de pontuação. Essas atualizações complementam as funcionalidades já existentes nos diversos módulos, contribuindo para maior eficiência e praticidade no gerenciamento das atividades docentes.

Poucos anos depois, Reis (2021) buscou aprimorar a usabilidade do sistema e ampliar suas funcionalidades, tornando-o mais eficiente e intuitivo. No módulo de gerenciamento de atividades, foram implementadas diversas melhorias voltadas à eficiência e à usabilidade do sistema. Uma das funcionalidades introduzidas foi a possibilidade de reutilizar documentos já existentes em relatórios anteriores para preencher os documentos necessários para novas atividades. Além disso, foi implementada a remoção de relações entre documentos referenciados que sejam de reuso. Foi corrigido um bug relacionado às datas de início e fim do interstício. Por fim, a navegação na escolha do nível de promoção ou progressão de carreira foi aprimorada, tornando o processo mais fluido e intuitivo.

No módulo de relatórios, foi adicionado um recurso que exibe a quantidade de horas lançadas em cada atividade, proporcionando maior clareza na visualização dos dados registrados. Além disso, o submódulo de relatórios foi reformulado, com melhorias estruturais e visuais que facilitam a utilização. No módulo de gerenciamento para avaliadores, foram incluídos novos meios para consulta de resoluções e foi realizada uma melhoria na interface desse módulo, com o objetivo de torná-la mais intuitiva.

O autor utilizou as mesmas tecnologias mencionadas por Silva (2019), como C#, .NET, Angular e PostgreSQL, mas incorporou o uso do Entity Framework para o mapeamento objeto-relacional (ORM), facilitando a manipulação e persistência dos dados no banco de dados. Além disso, adotou os princípios do SOLID no desenvolvimento, garan-

tindo um código mais modular, manutenível e alinhado às boas práticas de engenharia de software.

4 Desenvolvimento

Neste capítulo, será apresentado o método adotado para o desenvolvimento do trabalho, abrangendo a especificação dos requisitos e a implementação do sistema. Nas seções a seguir, serão destacados o método utilizado, os requisitos definidos e o processo de implementação realizado.

4.1 Método

Uma vez disposto de uma lista de requisitos sugeridos pelos usuários da aplicação, o fluxo de trabalho foi organizado em ciclos. Esses ciclos incluíram reuniões periódicas com o administrador do sistema, elencamento de requisitos prioritários e a subsequente implementação das correções e melhorias destes requisitos.

Para otimizar o gerenciamento e monitoramento das demandas, utilizou-se a ferramenta Trello¹. Essa plataforma proporciona maior organização visual das demandas a serem implementadas, facilitando o acompanhamento do progresso do trabalho.

4.1.1 Especificação de Requisitos

As tarefas de manutenção realizadas neste trabalho foram baseadas em sugestões de melhorias e correções enviadas pelos usuários do SCAD, os professores da FACOM, ao administrador do sistema. Foram priorizadas atividades voltadas para o aperfeiçoamento da usabilidade e a correção de problemas na interação do sistema. A seguir, são descritos os requisitos tratados neste trabalho.

4.1.1.1 Requisitos Corretivos

Esta seção apresenta requisitos voltados para corrigir falhas ou problemas existentes no sistema, como comportamentos inesperados que comprometem a funcionalidade ou a experiência do usuário.

- Ajustar preenchimento automático nos campos de data final das atividades;
- validar consistência das datas cadastradas nas atividades;
- manter apresentação de quebra de linha na descrição das atividades na listagem;
- corrigir mecanismo de busca.

¹ <https://trello.com/>

4.1.1.2 Requisitos Evolutivos

Esta seção apresenta requisitos voltados para implementar novas funcionalidades, melhorar funcionalidades existentes ou aprimorar a experiência do usuário.

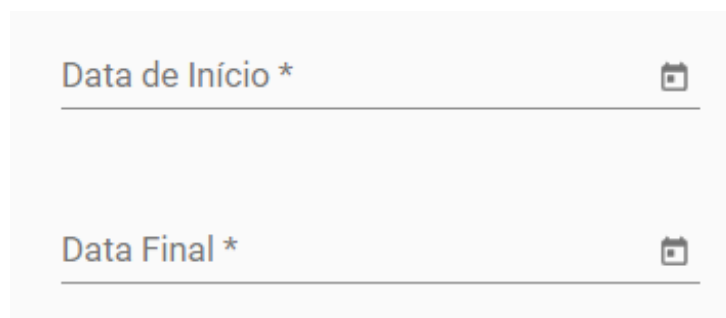
- Adicionar máscara nos campos de data;
- remover limitação de valor máximo no campo de data final das atividades;
- melhorar modal de encerramento de relatório;
- desconsiderar acentuação nas buscas;
- persistir a quantidade de itens por página exibidos na lista de atividades;
- persistir a página selecionada na lista de atividades.

4.2 Implementação dos Requisitos

4.2.1 Ajustar preenchimento automático nos campos de data final das atividades

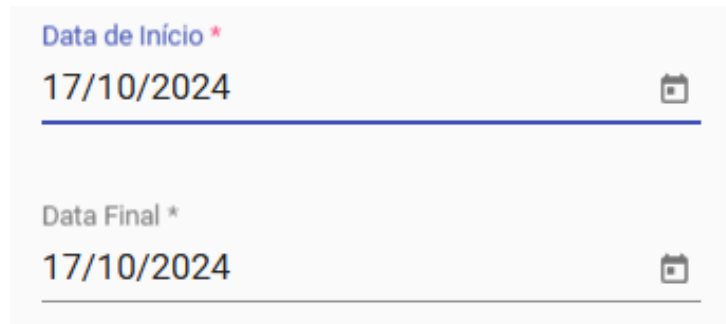
Ao preencher um valor no campo Data de Início no formulário de cadastro e edição de uma atividade, o campo Data Final é automaticamente preenchido com o mesmo valor. Esse comportamento é uma regra previamente implementada no SCAD e funciona como descrito. A finalidade é facilitar o preenchimento para o usuário, uma vez que a data final muitas vezes é composta de pelo menos parte da data inicial. As Figuras 28 e 29 ilustram, respectivamente, os campos vazios antes de serem preenchidos e após inserir manualmente uma data apenas em Data de Início.

Figura 28 – Campos de data da atividade antes do preenchimento.

A imagem mostra dois campos de entrada de data, um acima do outro. O campo superior é rotulado "Data de Início *" e o campo inferior é rotulado "Data Final *". Ambos os campos são vazios e possuem um ícone de calendário no canto superior direito. Os rótulos são em uma cor azulada e o asterisco indica que são campos obrigatórios.

Fonte: Imagem do SCAD.

Figura 29 – Campos de data da atividade ao inserir um valor em Data de Início.



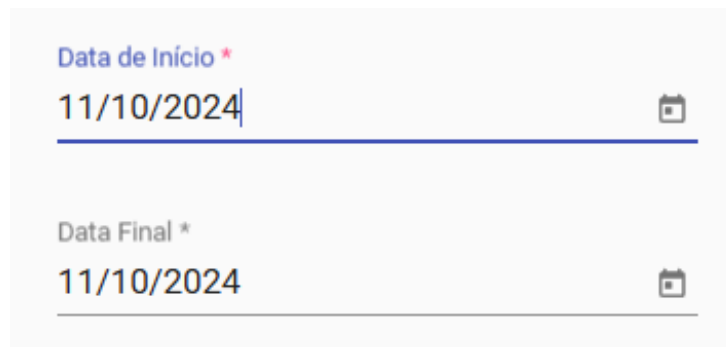
Formulário com dois campos de data:

- Data de Início ***: 17/10/2024
- Data Final ***: 17/10/2024

Fonte: Imagem do SCAD.

Contudo, registrou-se cenários onde esse comportamento é indesejável. Por exemplo, ao editar uma atividade e corrigir uma data inicial que estava anteriormente errada, o campo Data Final tinha seu valor substituído pela nova data inicial e o valor anterior era perdido (Figura 30).

Figura 30 – Campos de data da atividade ao alterar Data de Início substituindo automaticamente o valor de Data Final.



Formulário com dois campos de data:

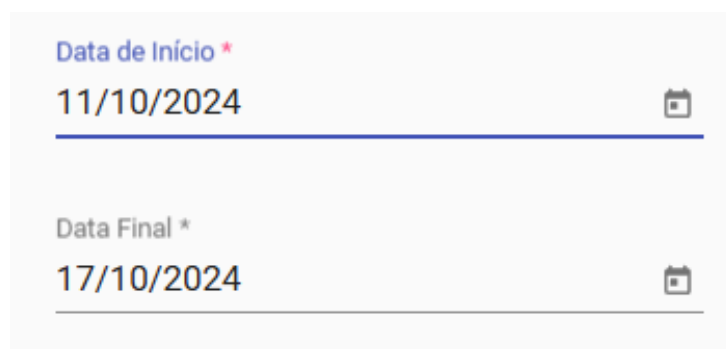
- Data de Início ***: 11/10/2024
- Data Final ***: 11/10/2024

Fonte: Imagem do SCAD.

4.2.1.1 Resolução

Foi implementado uma verificação condicional na função que executava este preenchimento automático. Com a alteração, a função examina antes se o nó correspondente à Data Final já está preenchido, atualizando-o caso o teste retorne falso. Desta forma ao alterar o valor de Data de Início com o campo Data Final já populado, o preenchimento automático de Data Final não ocorre, permanecendo o valor anterior (Figura 31).

Figura 31 – Campos de data da atividade ao alterar Data de Início mantendo o valor já presente em Data Final



Data de Início *
11/10/2024

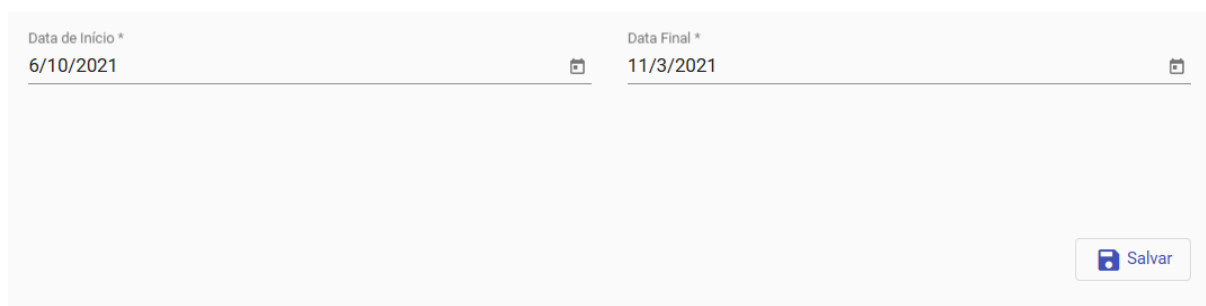
Data Final *
17/10/2024

Fonte: Imagem do SCAD.

4.2.2 Validar consistência das datas cadastradas nas atividades

A duração de uma atividade é um dos itens que define os pontos atribuídos à ela. Se um usuário do SCAD, por engano, preenche uma atividade com a Data de Início maior que Data Final (Figura 32), essa atividade acaba por calcular uma pontuação negativa (Figura 33).

Figura 32 – Atividade com data inicial maior que data final



Data de Início *
6/10/2021

Data Final *
11/3/2021

Salvar

Fonte: Imagem do SCAD.

Figura 33 – Atividade com pontuação negativa

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
A1.2 - ATIVIDADES DE ORIENTAÇÃO	Orientação Estágio João 14 - Orientação e/ou supervisão de alunos de graduação e da educação profissional em atividades curriculares de ensino, sem a presença do docente, desde que não pontuada nas Atividades de Ensino. 1 ponto(s) Por aluno/mês completo	Início: 06/10/2021 Fim: 11/03/2021 -6 mes(es)	-6 ponto(s)	   

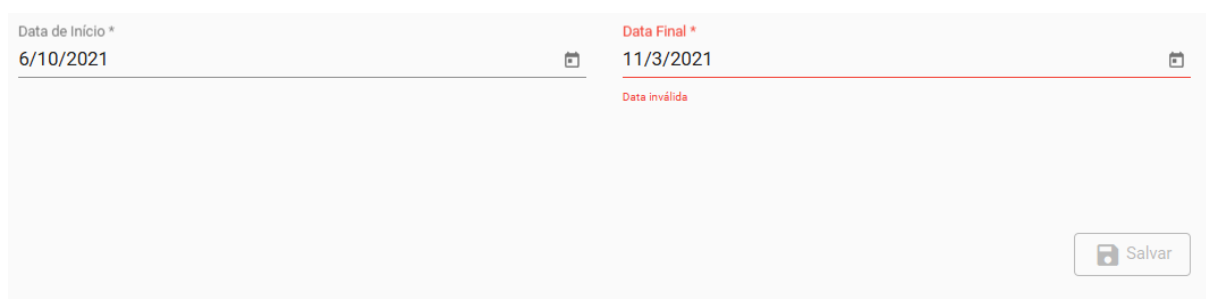
Fonte: Imagem do SCAD.

A princípio, neste requisito foi solicitado a adição de um alerta informando o usuário de que esta atividade produziria pontuação negativa. No entanto, entendeu-se que uma validação direta nos campos de data seria mais proveitosa.

4.2.2.1 Resolução

Foi adicionado uma validação ao campo Data Final para que o valor mínimo permitido nele seja igual o valor presente em Data de Início. Caso a validação falhe, o campo é destacado em vermelho, indicando a inconsistência e o botão de salvar é desabilitado, como mostra a Figura 34, impedindo o usuário de cadastrar uma atividade que gere pontuação negativa.

Figura 34 – Validação que impede salvar atividade com data inicial maior que data final.

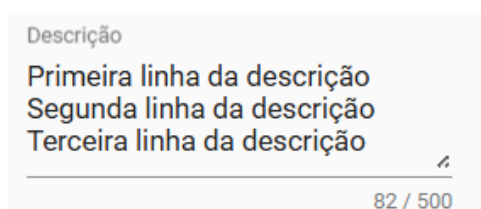


Fonte: Imagem do SCAD.

4.2.3 Manter apresentação de quebra de linha na descrição das atividades na listagem

O campo Descrição no formulário de cadastro de atividade permite quebra de linhas como no exemplo da Figura 35.

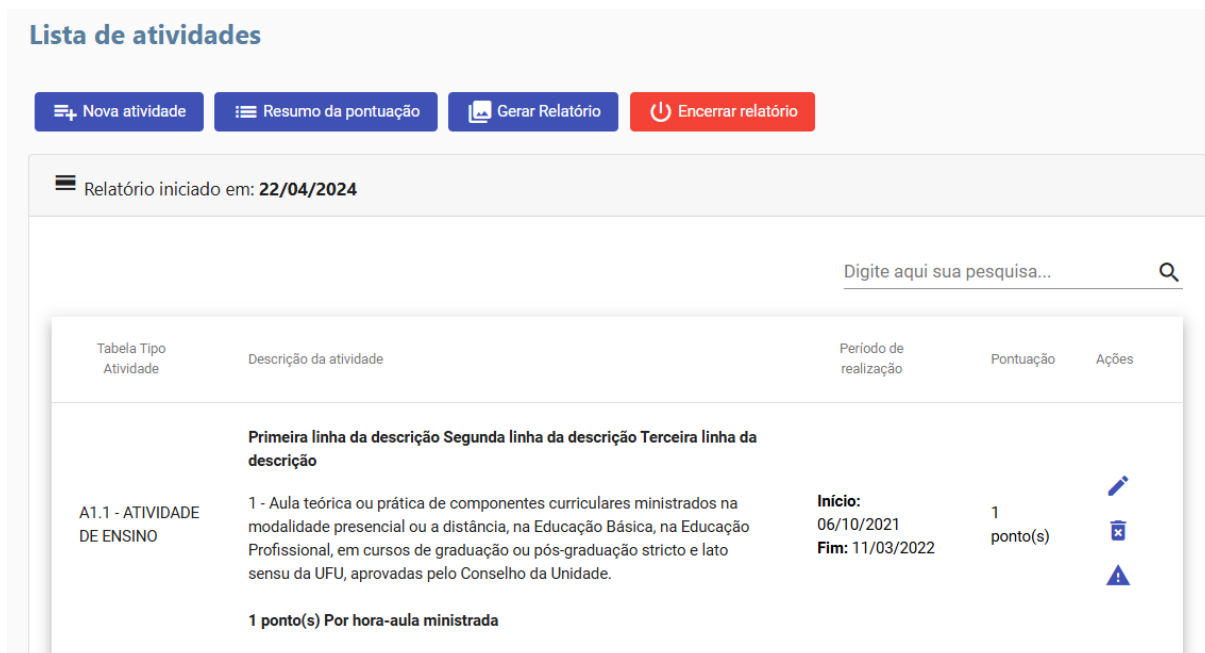
Figura 35 – Campo de descrição no formulário de cadastro de atividade.



Fonte: Imagem do SCAD.

No entanto, na tela de lista de atividades, as quebras de linhas não são mantidas e são substituídas por espaços em branco (Figura 36).

Figura 36 – Descrição sem quebra de linhas na lista de atividades.



Fonte: Imagem do SCAD.

Ao editar a atividade em questão o campo Descrição apresenta normalmente o texto com as quebras de linhas. O problema é, de fato, a impersistência dessa configuração quando a atividade é exibida como um item na página de lista de atividades.

4.2.3.1 Resolução

Notou-se que a quebra de linha persiste na página de edição de uma atividade, pois esse campo é definido com a *tag* `<textarea>` do HTML que, de fato, representa uma caixa de texto que respeita a formatação nela inserida. Porém, na lista de atividades essa descrição é renderizada como um texto simples contido em uma *tag* `<div>`. Ao inspecionar o código HTML do elemento é possível confirmar que o conteúdo ainda contém as quebras de linha, mas, por padrão, a apresentação de texto em uma `<div>` substitui quebras de linha por espaços em branco.

O comportamento de substituição ou não das quebras de linha em uma página *web* é definida pela propriedade CSS `white-space`, que é usada para descrever como os espaços em branco dentro dos elementos são manipulados. Seu valor inicial é `normal` e é esse valor que a propriedade assume em casos omissos. Um dos comportamentos implicados por `white-space: normal` é justamente o recolhimento de quebra de linhas, isto é, a exibição delas como espaços em branco.

Para a correção, foi identificado no projeto *front-end* do SCAD o seletor CSS utilizado para estilizar a descrição na lista de atividades e adicionado explicitamente à ele a

propriedade `white-space` com o valor `pre-line`. Esse valor se diferencia do `normal` precisamente na forma que é tratada a quebra de linha, preservando-a neste caso (Figura 37).

Figura 37 – Descrição com quebra de linhas na lista de atividades.

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
A1.1 - ATIVIDADE DE ENSINO	Primeira linha da descrição Segunda linha da descrição Terceira linha da descrição 1 - Aula teórica ou prática de componentes curriculares ministrados na modalidade presencial ou a distância, na Educação Básica, na Educação Profissional, em cursos de graduação ou pós-graduação stricto e lato sensu da UFU, aprovadas pelo Conselho da Unidade.	Início: 06/10/2021 Fim: 11/03/2022	1 ponto(s)	

Fonte: Imagem do SCAD.

4.2.4 Corrigir mecanismo de busca

Algumas páginas estavam com as buscas desfuncionais, sem de fato filtrar os resultados com o termo inserido no campo de busca. A Figura 38 exemplifica a busca pelo termo “palestra” sem sucesso, com a lista continuando a exibir todas as atividades daquele relatório.

Figura 38 – Falha ao buscar pelo termo “palestra” na lista de atividades.

Relatório iniciado em: 22/04/2024

Digite aqui sua pesquisa...
palestra

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
A1.2 - ATIVIDADES DE ORIENTAÇÃO	Coorientação Jorge 20 - Coorientação de Dissertação de Mestrado. 1,25 ponto(s) Por aluno/mês completo	Início: 10/05/2024 Fim: 10/11/2024 6 mes(es)	7,5 ponto(s)	  
A1.3 - PRODUÇÃO INTELECTUAL	Minicurso Techweek 40 - Palestras e conferências proferidas, minicursos ministrados, seminários, oficinas, participação em mesas redondas ou em painéis de debate em reuniões científicas não promovidas por associações ou sociedades científicas. 10 ponto(s) Por tema	Início: 10/10/2024 Fim: 11/10/2024	20 ponto(s)	  
A1.1 - ATIVIDADE DE ENSINO	atendimento 3 - Atendimento educacional especializado oferecido em modalidade de aula para alunos da Educação Especial (Decreto 5671/2008 e 7611/2011) em decorrência de quadro de deficiência e que estejam regularmente matriculados na educação básica, na educação profissional, em cursos de graduação ou pós-graduação stricto e lato sensu da UFU, aprovado pelo Conselho da Unidade. 1 ponto(s) Por hora-aula ministrada	Início: 02/11/2024 Fim: 02/11/2024	1 ponto(s)	  

Fonte: Imagem do SCAD.

4.2.4.1 Resolução

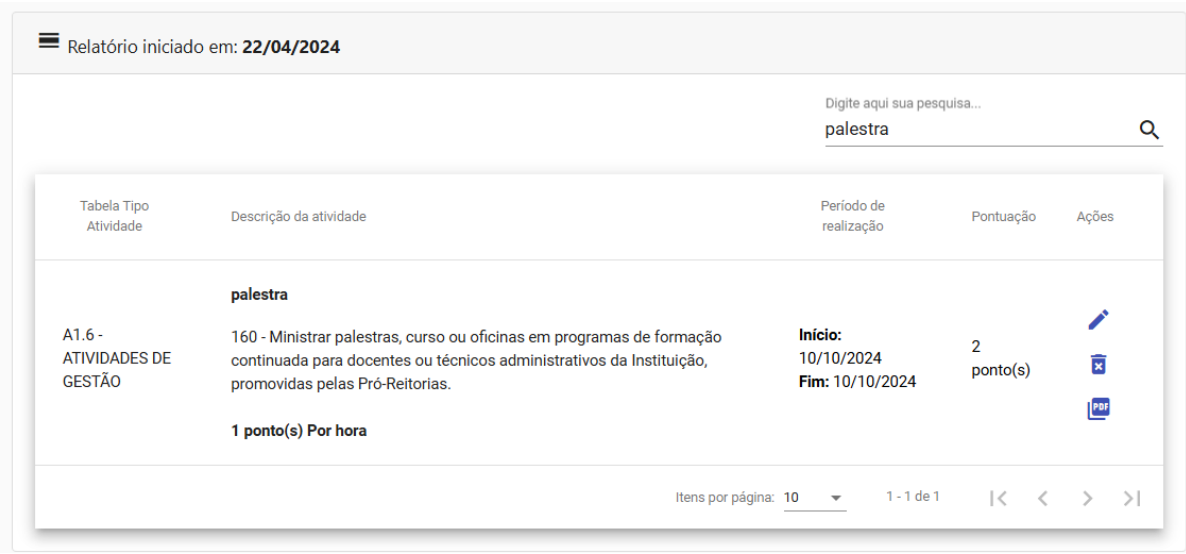
Constatou-se que as páginas em que a busca não funcionava eram aquelas em que eram feitas requisições para o *back-end* com o termo da busca. Essas requisições sempre retornavam um erro 500, que indica que o servidor não conseguiu executar a solicitação. De forma contrária, as páginas em que a busca funcionava normalmente, eram aquelas em que todo o algoritmo da busca foi implementado no *front-end*.

Para que a busca opere sem realizar requisições ao servidor, é necessário que os dados de todos os itens daquele módulo sejam carregados ao cliente. É possível que esses módulos com buscas desfuncionais, como a Lista de Atividades, realizavam consultas paginadas ao servidor, trazendo ao cliente apenas os dados exibidos naquela página no momento. Contudo, observando a resposta do servidor quando é carregado o módulo, percebe-se que esse não é mais o caso, já que todos os dados são carregados uma única vez ao acessar Lista de Atividades.

Tendo em vista que todos os dados estão disponíveis ao *front-end* ao carregar o

módulo e considerando a pequena escala do sistema, não há contraindicação em continuar operando dessa forma. Foi implementado, então, um algoritmo de busca usando componentes próprios do Angular. A principal vantagem de uma busca dessa forma é que, como os dados já estão no lado do cliente a busca será muito rápida. A Figura 39 ilustra a mesma tentativa de busca com o termo “palestra”, mas, desta vez, com sucesso, retornando apenas as atividades que contém o termo.

Figura 39 – Sucesso ao buscar pelo termo “palestra” na lista de atividades.



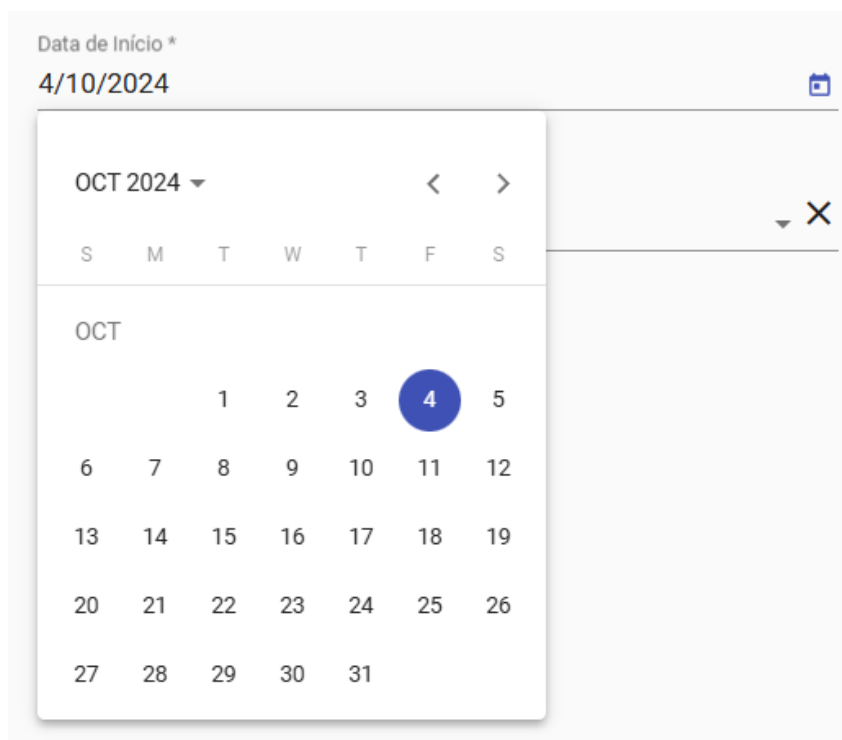
The screenshot shows a web application interface. At the top, a header bar indicates 'Relatório iniciado em: 22/04/2024'. Below this is a search bar with the placeholder 'Digite aqui sua pesquisa...' and the search term 'palestra' entered. A search icon is to the right of the input. Below the search bar is a table with the following columns: 'Tabela Tipo Atividade', 'Descrição da atividade', 'Período de realização', 'Pontuação', and 'Ações'. The table contains one row for the activity 'A1.6 - ATIVIDADES DE GESTÃO'. The description is '160 - Ministrar palestras, curso ou oficinas em programas de formação continuada para docentes ou técnicos administrativos da Instituição, promovidas pelas Pró-Reitorias.' The period is 'Início: 10/10/2024' and 'Fim: 10/10/2024'. The score is '2 ponto(s)'. The actions column contains three icons: a pencil, a trash can, and a PDF icon. Below the table, there is a footer bar with 'Itens por página: 10', '1 - 1 de 1', and navigation arrows.

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
A1.6 - ATIVIDADES DE GESTÃO	160 - Ministrar palestras, curso ou oficinas em programas de formação continuada para docentes ou técnicos administrativos da Instituição, promovidas pelas Pró-Reitorias.	Início: 10/10/2024 Fim: 10/10/2024	2 ponto(s)	  

Fonte: Imagem do SCAD.

4.2.5 Adicionar máscara nos campos de data

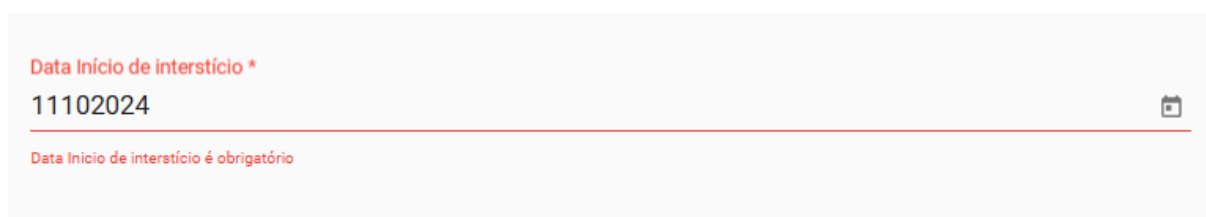
Nos campos de data do SCAD existe uma validação para checar se o valor inserido é de fato uma data. Uma das restrições é a necessidade do valor estar no formato dd/mm/aaaa. Ao selecionar a data usando o componente *datepicker* disponível, ela é devidamente formatada com as barras entre dia, mês e ano (Figura 40).

Figura 40 – Selecionando data pelo *datepicker*.

Fonte: Imagem do SCAD.

É possível, também, adicionar a data digitando diretamente no campo, mas isso introduz um distúrbio de usabilidade. Esse distúrbio acontece, pois o usuário precisa digitar as barras, do contrário, a data é considerada inválida (Figura 41).

Figura 41 – Data com formatação inválida.



Fonte: Imagem do SCAD.

Como as barras devem sempre ser inseridas e sempre nas mesmas posições, é comum nos sistemas modernos existir uma máscara no campo, de modo que elas sejam inseridas automaticamente, facultando ao usuário a ação de digitá-las.

4.2.5.1 Resolução

Para implementar essa melhoria, é necessário criar uma máscara de dados, isto é, uma pré-definição do formato de dados que o usuário pode inserir no campo. O Angular já possui diretivas próprias para datas que auxiliam nessa tarefa. Feito isso, foram

adicionadas algumas regras adicionais para assegurar bom funcionamento como, por exemplo, impedir inserção de caracteres não numéricos além da barra e assegurar o formato dd/mm/aaaa. A Figura 42 exemplifica o valor presente no campo após digitar nele o mesmo valor “11102024”, que antes gerava um erro.

Figura 42 – Data com formatação válida usando máscara.

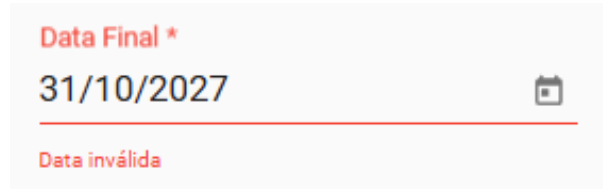
A imagem mostra uma interface de usuário com um campo de entrada de texto. No topo do campo, há o rótulo "Data Início de interstício *" em cinza. Abaixo dele, o valor "10/11/2024" está exibido em azul. À direita do campo, há um ícone de calendário. Uma linha horizontal separa o campo de entrada de uma área inferior.

Fonte: Imagem do SCAD.

4.2.6 Remover limitação de valor máximo no campo de data final das atividades

O campo Data Final nos formulários de cadastro de Relatório e Atividade não aceitavam algumas datas. A Figura 43 ilustra um caso de data considerada inválida.

Figura 43 – Data inválida impedindo salvar formulário.

A imagem mostra uma interface de usuário com um campo de entrada de texto. No topo do campo, há o rótulo "Data Final *" em vermelho. Abaixo dele, o valor "31/10/2027" está exibido em azul. À direita do campo, há um ícone de calendário. Uma linha horizontal separa o campo de entrada de uma área inferior, onde a mensagem "Data inválida" está exibida em vermelho.

Fonte: Imagem do SCAD.

4.2.6.1 Resolução

Após análise, foi localizado no código uma validação que permite a Data Final conter um valor no máximo equivalente ao final do ano atual, acrescido de dois anos. Por exemplo, estando em 2024, o campo Data Final pode assumir um valor máximo de 31/12/2026.

Essa validação foi possivelmente implementada para se adequar a regra de negócio do interstício que dura dois anos. Como o SCAD foi pensado para cadastrar relatórios e atividades para um determinado interstício, um cadastro efetivamente não teria duração de mais de dois anos.

Apesar disso, existe a intenção de aprimorar o sistema para, por exemplo, cadastrar atividades em mais de um interstício, uma vez que uma atividade possa começar em um interstício e terminar em outro. Atualmente isso é feito cadastrando atividades duplicadas

em múltiplos relatórios. Além disso, um professor pode querer planejar uma atividade futura e já deixá-la cadastrada, mas a validação atual pode impedir isso.

Quando o SCAD passar a permitir uma atividade em diferentes interstícios, convém reavaliar a validação de data máxima. Por ora, a validação foi mantida, porém, foi ajustada para uma data arbitrariamente alta (31/12/2099). Na prática, isso equivale à remoção da validação. Dessa forma, a data de exemplo 31/10/2027, que antes era inválida, passa a ser válida (Figura 44).

Figura 44 – Data válida permitindo salvar formulário.



Fonte: Imagem do SCAD.

4.2.7 Melhorar modal de encerramento de relatório

Na página Lista de atividades ao clicar no botão Encerrar Relatório uma modal de confirmação é exibida (Figura 45). Visto que a ação de encerrar relatório implica irreversivelmente na impossibilidade de sua alteração a partir deste momento, elencou-se a necessidade dessa modal passar a ser uma modal de alerta reforçando essa informação.

Figura 45 – Modal de confirmação ao encerrar relatório.

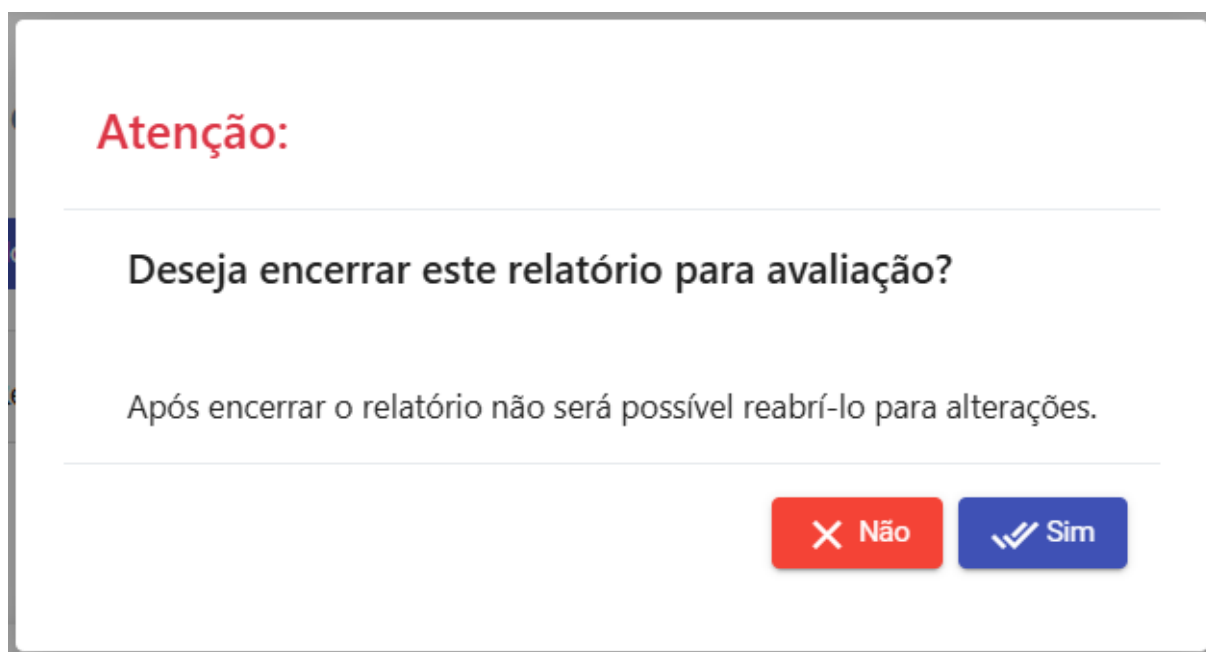


Fonte: Imagem do SCAD.

4.2.7.1 Resolução

A modal de confirmação foi substituída por uma modal de alerta, com o texto “Atenção” em destaque e vermelho, e uma mensagem que enfatiza a consequência do encerramento como ilustrado na Figura 46.

Figura 46 – Nova modal de alerta ao encerrar relatório.



Fonte: Imagem do SCAD.

4.2.8 Desconsiderar acentuação nas buscas

O sistema de busca do SCAD traz resultados que contêm exatamente o termo buscado. Embora a exatidão do termo buscado possa parecer desejável para evitar que a busca retorne itens não correspondentes, é comum realizar buscas desconsiderando a acentuação e outros sinais diacríticos, como til e cedilha.

A Figura 47 ilustra o caso de uma busca pelo termo “pós-graduação” que retorna apenas um resultado, Já na Figura 48, é possível notar que a mesma busca, mas sem os sinais diacríticos, não retorna nenhum resultado.

Figura 47 – Busca com acento.

Digite aqui sua pesquisa...
pós-graduação

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
A1.3 - PRODUÇÃO INTELLECTUAL	Banca pós-graduação 77 - Membro titular de banca de defesa de Monografia ou Trabalho de Conclusão de Curso de Pós-graduação lato sensu. 5 ponto(s) Por participação	Início: 10/10/2024 Fim: 10/10/2024	5 ponto(s)	  

Itens por página: 10 1 - 1 de 1

Fonte: Imagem do SCAD.

Figura 48 – Falha ao buscar sem acento.

Digite aqui sua pesquisa...
pos-graduacao

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
Itens por página: 10 0 de 0				

Fonte: Imagem do SCAD.

4.2.8.1 Resolução

Foi elaborado um método que normaliza uma dada **string**. A normalização consiste em remover acentos ou qualquer outro sinal diacrítico da língua portuguesa do texto passado como parâmetro.

Desta forma, normalizando tanto o termo de busca quanto os dados a serem buscados, garante-se que os resultados da busca serão satisfatórios independente de acentuação. A Figura 49 mostra a mesma busca pelo termo “pos-graduacao” agora retornando com sucesso o item esperado.

Figura 49 – Sucesso ao buscar sem acento.

Digite aqui sua pesquisa...

pos-graduacao

Tabela Tipo Atividade	Descrição da atividade	Período de realização	Pontuação	Ações
A1.3 - PRODUÇÃO INTELLECTUAL	Banca pós-graduação 77 - Membro titular de banca de defesa de Monografia ou Trabalho de Conclusão de Curso de Pós-graduação lato sensu. 5 ponto(s) Por participação	Início: 10/10/2024 Fim: 10/10/2024	5 ponto(s)	  

Itens por página: 10 1 - 1 de 1 |< < > >|

Fonte: Imagem do SCAD.

Uma medida adotada para a normalização dos dados dos itens na lista foi processar e normalizar todos os itens no momento do carregamento, armazenando as novas **strings** normalizadas em seus respectivos objetos. Isso garante que os resultados das buscas estejam sempre disponíveis, além de proporcionar um desempenho elevado durante a realização das buscas, pois a normalização não será feita toda vez que for realizada uma busca, mas já no carregamento da página.

4.2.9 Persistir a quantidade de itens por página exibidos na lista de atividades

A Lista de Atividades é paginada e, inicialmente, exibe no máximo 10 itens (Figura 50). O usuário tem a opção de aumentar o número de itens exibidos por página, com as opções de 10, 20, 50 ou 100 itens (Figura 51).

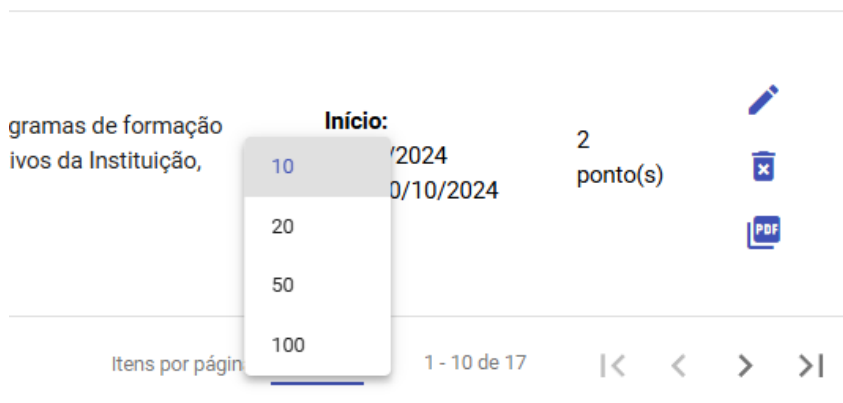
Figura 50 – Carregamento inicial de 10 atividades na página.

	palestra			
A1.6 - ATIVIDADES DE GESTÃO	160 - Ministrar palestras, curso ou oficinas em programas de formação continuada para docentes ou técnicos administrativos da Instituição, promovidas pelas Pró-Reitorias.	Início: 10/10/2024 Fim: 10/10/2024	2 ponto(s)	  
	1 ponto(s) Por hora			

Itens por página: 10 1 - 10 de 17 |< < > >|

Fonte: Imagem do SCAD.

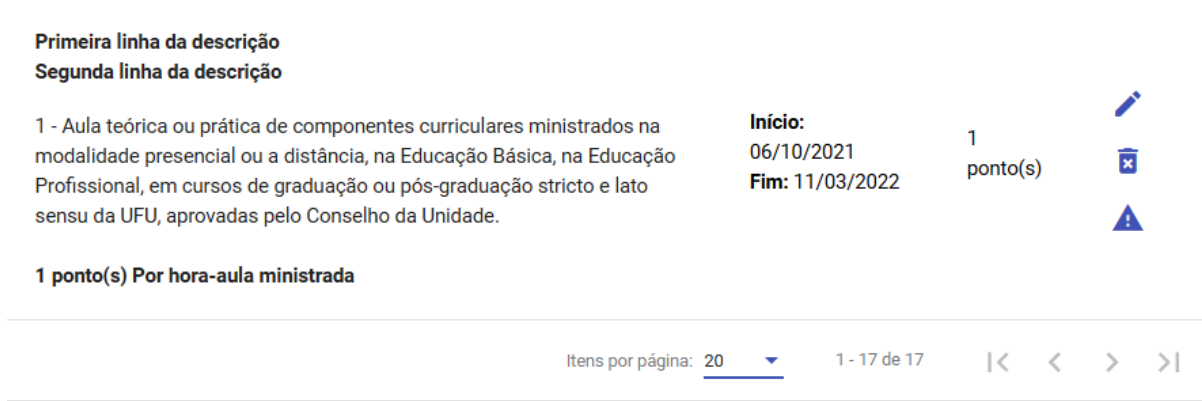
Figura 51 – Caixa de seleção de quantidade de itens a serem exibidos na página.



Fonte: Imagem do SCAD.

Ao selecionar, por exemplo, a opção de exibir 20 itens (Figura 52), a página deixa de limitar a exibição a apenas 10 itens e passa a mostrar até 20. Nesse caso, como a lista contém 17 itens, todos são exibidos.

Figura 52 – Até 20 itens listados na página.



Fonte: Imagem do SCAD.

Esta configuração permite que o usuário defina sua preferência de visualização. Porém, ao editar uma atividade e salvar as alterações, ou até mesmo cancelar a edição, a lista é recarregada com a configuração padrão de 10 itens. É objetivo desse requisito que essa preferência seja mantida enquanto estiverem navegando no contexto das atividades.

4.2.9.1 Resolução

Para atender a essa demanda, foi definido que o contexto de atividades corresponde à lista de atividades e aos seus respectivos itens. Assim, caso o usuário tenha selecionado uma quantidade de itens diferente de 10, essa preferência será preservada enquanto ele permanecer nesse contexto, incluindo edições nas atividades.

Ao retornar da edição de uma atividade, o comportamento atual do componente da listagem definido no Angular é recarregado e as preferências de exibição redefinidas. Implementou-se, então, uma solução para persistir essa informação temporariamente. Como trata-se de um dado de curto prazo, foi decidido armazená-lo no próprio navegador do cliente.

Foi criado um procedimento que escuta o evento de alteração na quantidade de itens exibidos. Sempre que essa configuração é modificada, o novo valor é armazenado em uma variável de sessão. Essa variável é destruída ao sair da página de lista de atividades, exceto quando a saída ocorre para editar uma atividade. Isso garante que, ao retornar para a lista após uma edição, a quantidade de itens exibidos permaneça conforme a última configuração do usuário. Dessa forma, a experiência do usuário é aprimorada, respeitando suas preferências enquanto navega e interage no contexto das atividades.

4.2.10 Persistir a página selecionada na lista de atividades

Este requisito é semelhante ao anterior, pois trata da persistência da página na lista de atividades. Desta vez, a solicitação é que, ao editar uma atividade e retornar à tela de lista de atividades, a página exibida seja a mesma em que o usuário estava antes de iniciar a edição.

Por exemplo, considerando o cenário do requisito anterior com 17 itens e a configuração de exibição definida para 10 itens por página, a lista será dividida em duas páginas, como na Figura 53. Se o usuário estiver na segunda página e clicar em Editar em uma atividade, ao retornar à lista, atualmente a exibição volta para a primeira página por padrão. No entanto, o comportamento esperado é que a lista volte à mesma página em que o usuário estava antes de editar a atividade.

Figura 53 – Página 2 da Lista de atividades.

Primeira linha da descrição		Segunda linha da descrição			
A1.1 - ATIVI DADE DE ENSI NO	1 - Aula teórica ou prática de componentes curriculares ministrados na modalidade presencial ou a distância, na Educação Básica, na Educação Profissional, em cursos de graduação ou pós-graduação stricto e lato sensu da UFU, aprovadas pelo Conselho da Unidade.	Início: 06/10/ 2021	1 pon to(s))		
		Fim: 11/03/ 2022			
1 ponto(s) Por hora-aula ministrada					
Itens por página: 10 11 - 17 de 17 < < > > 					

Fonte: Imagem do SCAD.

4.2.10.1 Resolução

Assim como o problema, a solução segue uma lógica semelhante à descrita no requisito anterior. Foi implementado o armazenamento da página atual em uma variável de sessão utilizando Angular. Essa variável é destruída ao sair do contexto da lista de atividades. Contudo, ao editar uma atividade, o contexto é mantido, e o valor armazenado na variável é utilizado para recarregar a lista na página correta ao retornar. Essa abordagem garante uma experiência mais fluida, preservando a navegação do usuário.

5 Conclusão

Este trabalho teve como foco aprimorar a usabilidade do sistema SCAD, direcionando esforços para melhorar a interação do usuário com a plataforma. As mudanças se concentraram principalmente no *front-end*, utilizando as funcionalidades avançadas do *framework* Angular.

Embora adotar uma nova tecnologia seja custoso no início para o progresso do desenvolvimento, o Angular mostrou-se extremamente vantajoso. Suas diretrizes se alinham às necessidades de muitos requisitos do sistema, facilitando e agilizando o desenvolvimento. Além disso, sua ampla adoção em sistemas contemporâneos oferece uma base sólida para o crescimento e a manutenção do SCAD, garantindo suporte contínuo, atualizações regulares e uma comunidade ativa, o que facilita tanto a expansão do sistema quanto a realização de manutenções futuras.

Ficou evidente, também, a importância de uma captação de requisitos eficaz, pois os usuários, como principais utilizadores do sistema, têm um conhecimento profundo sobre o que está funcionando bem e o que precisa ser aprimorado. Os usuários frequentemente identificam necessidades que, muitas vezes, não são percebidas pelos desenvolvedores ao estudar o sistema. A coleta eficaz desses requisitos é essencial para garantir que o sistema evolua de acordo com as expectativas e realidades do seu público, resultando em soluções mais precisas e eficientes.

Ao longo deste trabalho, foram identificados dois aspectos relevantes que merecem atenção em atividades futuras. O primeiro refere-se aos padrões de projeto. Embora boas práticas de programação estejam incorporadas no projeto *front-end* do SCAD, reforçar a adesão a essas práticas pode representar uma oportunidade para futuras iniciativas. Detectou-se que alguns segmentos de código contêm redundâncias, o que pode sugerir um uso limitado de lógica comum entre as funcionalidades. Apesar disso não constituir um problema crítico, eleva o trabalho de manutenção e pode resultar em inconsistências se partes redundantes forem modificadas em diferentes partes do sistema. Atuar para diminuir essas redundâncias auxiliará na criação de um sistema mais eficaz e sustentável.

Outro aspecto importante é a organização e o mapeamento do banco de dados. No momento, o sistema emprega Mapeamento Objeto-Relacional (Object-Relational Mapping - ORM), um instrumento robusto que simplifica o acesso ao banco de dados, permitindo que as operações sejam executadas de forma mais intuitiva e integrada ao código. Contudo, se o mapeamento entre objetos e tabelas não for adequadamente planejado, pode se tornar uma constante fonte de problemas. Estruturas de banco de dados indefinidas, relações complexas ou não normalizadas, e a falta de um planejamento apropriado

podem tornar as migrações particularmente desafiadoras, especialmente para os novos programadores que se juntam ao projeto.

Conflitos relacionados a atualizações de esquema são problemas que podem ocorrer ao se utilizar o ORM. Dificuldades relacionadas a chaves estrangeiras, mudanças em colunas que já possuem dados ou inconsistências entre ambientes - como o de desenvolvimento e produção - também são exemplos comuns de desafios que podem surgir. Esses problemas, frequentemente, são complexos de identificar e corrigir, pois um simples erro de mapeamento ou uma execução imprópria da sequência de migrações pode resultar em falhas graves durante a execução. Portanto, é aconselhável fazer uma avaliação detalhada da estrutura do banco de dados e do procedimento de migração empregado no SCAD. Isso pode envolver o remodelamento das tabelas já existentes, a melhoria das relações entre elas, a remoção de redundâncias e a asseguuração de que o mapeamento feito pelo ORM seja claro e funcional.

Ademais, medidas como dividir migrações extensas em fases menores e testar as mudanças em cenários isolados podem diminuir consideravelmente a probabilidade de falhas. Com a implementação dessas melhorias, o sistema se tornará mais robusto e de fácil manutenção, possibilitando que os programadores se concentrem na criação de novas funcionalidades, ao invés de consumir tempo resolvendo questões de infraestrutura.

Referências

ALVES, J. V. A. **Melhorias na infraestrutura do sistema de cadastro de atividades docente**. 2022. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Federal de Uberlândia. Disponível em: <<https://repositorio.ufu.br/handle/123456789/36031>>. Acesso em: 19 set. 2024. Citado na página 8.

Angular Team. **Essentials: A short introduction to some of Angular’s main concepts**. 2024. <<https://angular.dev/essentials>>. Acesso em: 15 nov. 2024. Citado na página 29.

_____. **Introduction: What is Angular?** 2024. <<https://angular.dev/overview>>. Acesso em: 15 nov. 2024. Citado 2 vezes nas páginas 29 e 30.

BAUER, A.; CASSETTARI, N.; OLIVEIRA, R. P. d. Políticas docentes e qualidade da educação: uma revisão da literatura e indicações de política. **Ensaio: Avaliação e Políticas Públicas em Educação**, Fundação CESGRANRIO, v. 25, n. 97, p. 943–970, Oct 2017. ISSN 0104-4036. Disponível em: <<https://doi.org/10.1590/S0104-40362017002501010>>. Acesso em: 13 nov. 2024. Citado na página 8.

Bootstrap. **Introduction**. 2024. <<https://getbootstrap.com/docs/4.1/getting-started/introduction/>>. Acesso em: 23 nov. 2024. Citado na página 30.

BRAGA, T. R. d. S. **Desenvolvimento de um Sistema de Cadastro de Atividades do Docente**. 2019. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Federal de Uberlândia. Disponível em: <<https://repositorio.ufu.br/handle/123456789/26783>>. Acesso em: 18 set. 2024. Citado 2 vezes nas páginas 8 e 31.

CONDIR (UFU). **Resolução nº 03/2017 - CONDIR - Progressão e promoção docente**. 2017. Disponível em: <https://progep.ufu.br/sites/proreh.ufu.br/files/conteudo/legislacao/leg_atacondir-2017-3.pdf>. Acesso em: 13 nov. 2024. Citado 3 vezes nas páginas 8, 25 e 26.

GAIKWAD, S. S.; ADKAR, P. A review paper on bootstrap framework. **IRE Journals**, v. 2, n. 10, p. 349–351, 2019. Citado na página 30.

GEETHA, G.; MITTAL, M.; PRASAD, K. M.; PONSAM, J. G. Interpretation and analysis of angular framework. In: **2022 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)**. [S.l.: s.n.], 2022. p. 1–6. Acesso em: 15 nov. 2024. Citado na página 29.

JOHNSON, R. E. Documenting frameworks using patterns. **SIGPLAN Not.**, Association for Computing Machinery, New York, NY, USA, v. 27, n. 10, p. 63–76, out. 1992. ISSN 0362-1340. Disponível em: <<https://doi.org/10.1145/141937.141943>>. Acesso em: 14 nov. 2024. Citado na página 28.

Microsoft. **Estratégia de linguagem do Microsoft .NET**. 2024. <<https://learn.microsoft.com/pt-br/dotnet/fundamentals/languages?source=recommendations#c>>. Acesso em: 14 nov. 2024. Citado na página 27.

_____. **Hub de documentação do Entity Framework**. 2024. <<https://learn.microsoft.com/pt-br/ef/>>. Acesso em: 14 nov. 2024. Citado na página 28.

_____. **Introdução ao .NET**. 2024. <<https://learn.microsoft.com/pt-br/dotnet/core/introduction>>. Acesso em: 14 nov. 2024. Citado 2 vezes nas páginas 26 e 27.

_____. **Introdução às consultas LINQ no C#**. 2024. <<https://learn.microsoft.com/pt-br/dotnet/csharp/linq/get-started/introduction-to-linq-queries>>. Acesso em: 15 nov. 2024. Citado na página 29.

_____. **Visão geral da consulta LINQ**. 2024. <<https://learn.microsoft.com/pt-br/dotnet/standard/linq/>>. Acesso em: 15 nov. 2024. Citado na página 29.

PESSOA, I. A.; MELO, A. L. d. A. O papel da tecnologia da informação na melhoria dos processos administrativos. **Ciências Sociais Aplicadas**, v. 28, n. 138, p. 21–21, Sep 2024. Registro DOI: 10.69849/revistaft/pa10202409212015. Citado na página 8.

PostgreSQL Global Development Group. **PostgreSQL 16 Documentation**. 2024. <<https://www.postgresql.org/files/documentation/pdf/16/postgresql-16-A4.pdf>>. Acesso em: 14 nov. 2024. Citado na página 28.

PRESSMAN, R. S. **Software Engineering: A Practitioner's Approach**. 7th. ed. New York, NY: McGraw-Hill, 2010. 928 p. ISBN 978-0-07-337597-7. Citado na página 9.

REIS, M. P. **Sistema de informação para gerenciamento de progressão e promoção funcional**. 2021. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Federal de Uberlândia. Disponível em: <<https://repositorio.ufu.br/handle/123456789/33841>>. Acesso em: 19 set. 2024. Citado na página 32.

SILVA, W. S. **Sistema de Cadastro de Atividades Docente**. 2019. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação). Disponível em: <<https://repositorio.ufu.br/handle/123456789/28962>>. Acesso em: 18 set. 2024. Citado 3 vezes nas páginas 8, 31 e 32.