

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

JEAN LUCAS DE OLIVEIRA LIMA

**Ferramenta para análise de produção
científica: estudo de caso no Instituto de
Informática da UFRGS**

Trabalho de Graduação.

Prof. Dr. Leandro Krug Wives

Porto Alegre, junho de 2014.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Graduação: Prof. Sérgio Roberto Kieling Franco

Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb

Coordenador do Curso de Ciência da Computação: Prof. Raul Fernando Weber

Bibliotecário-Chefe do Instituto de Informática: Beatriz Haro

AGRADECIMENTOS

Agradeço primeiramente a minha família, principalmente aos meus pais, que sempre me deram todo tipo de suporte e apoio oferecendo compreensão e suporte para que fosse possível a minha graduação.

Agradeço a todos os meus amigos, os que já existiam e os ganhei durante o período letivo, estes em especial os meus companheiros do Quem Chuta Busca F.C os quais podemos aproveitar a conquista de dois Dacompeonatos.

Agradeço ao professor Leandro, que sempre foi muito solícito e atencioso, pela orientação, suporte e ajuda que me permitiram chegar ao objetivo deste trabalho.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	6
LISTA DE FIGURAS	7
RESUMO	8
ABSTRACT	10
1 INTRODUÇÃO	11
2 AMBIENTE E TÉCNICAS UTILIZADAS	13
2.1 ABORDAGEM (OU TÉCNICAS)	13
2.1.1 Engenharia da Web	13
2.1.1.1 Princípios Ágeis	15
2.2 Requisitos	16
2.2.1 Requisitos Não-Funcionais	16
2.2.2 Requisitos Funcionais	16
2.3 Componentes do Sistema	17
2.3.1 .Net Framework	17
2.3.2 Banco de Dados	18
2.3.2.1 LINQ	19
2.3.2.2 LINQ – Sintaxe	19
2.3.2.3 LINQ – Como Funciona.....	19
2.3.3 HTML	20
2.3.4 CSS	20
2.3.5 JavaScript.....	20
2.3.5.1 Google Charts.....	21
2.4 Arquitetura	22
2.4.1 Cliente-Servidor.....	22
2.4.2 Model-View-Controller	23
2.4.2.1 Model	24
2.4.2.2 Controller	25
2.4.2.3 View	25
3 DESENVOLVIMENTO DO SISTEMA	27

3.1	Modelagem do Banco de Dados.....	27
3.1.1	Entidade Curriculos	28
3.1.2	Entidade Group	28
3.1.3	Entidade CurriculosGroup	28
3.1.4	Entidade Line	29
3.1.5	Entidade CurriculosLine	29
3.1.6	Entidade Area	29
3.1.7	Entidade QualisPeriodico.....	29
3.1.8	Entidade QualisConference	29
3.2	O Sistema	29
3.2.1	Tela Inicial	29
3.2.2	Tela de Pesquisa.....	30
3.2.2.1	Palavra Chave.....	31
3.2.2.2	Pesquisar por	32
3.2.2.3	Ordenar por	34
3.2.2.4	Agrupar por	36
3.2.2.5	Contar por Qualis	37
3.2.3	Telas de Pesquisadores	38
3.2.4	Telas de Grupos	40
3.2.5	Telas Áreas de Pesquisa.....	41
3.2.6	Telas de Linhas de Pesquisa	42
4	CONCLUSÃO	45
	REFERÊNCIAS.....	46

LISTA DE ABREVIATURAS E SIGLAS

API	Application Program Interface
BD	Banco de Dados
CLR	Common Language Runtime
CSS	Cascading Style Sheet
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
LINQ	Language Integrated Query
MVC	Model View Controller
NoSQL	Not Only Standard Query Language
SGBD	Sistema Gerenciador de Banco de Dados
SQL	Standard Query Language
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

LISTA DE FIGURAS

<i>Figura 2.1: Processo da Engenharia da Web.</i>	14
<i>Figura 2.2: Componentes do .NET Framework.</i>	18
<i>Figura 2.3: Arquitetura do LINQ.</i>	20
<i>Figura 2.4: Interação entre JavaScript, HTML e CSS.</i>	21
<i>Figura 2.5: Gráficos disponibilizados pelo Google Charts.</i>	22
<i>Figura 2.6: Modelo Cliente-Servidor.</i>	23
<i>Figura 2.7: Interação entre os componentes do MVC.</i>	24
<i>Figura 2.8: Organização de pastas e classes do Modelo da aplicação.</i>	24
<i>Figura 2.9: Classes do componente Controller da aplicação.</i>	25
<i>Figura 2.10: Páginas do componente View e organização de pastas.</i>	26
<i>Figura 3.1: Diagrama ER do banco de dados do sistema.</i>	28
<i>Figura 3.2: Tela inicial do sistema.</i>	30
<i>Figura 3.3: Tela de Pesquisa.</i>	31
<i>Figura 3.4: Pesquisa utilizando uma palavra chave.</i>	32
<i>Figura 3.5: Pesquisa sem selecionar filtro 'Pesquisar por'.</i>	32
<i>Figura 3.6: Pesquisa por Linha de Pesquisa.</i>	33
<i>Figura 3.7 Pesquisa por Grupo de Pesquisa.</i>	34
<i>Figura 3.8: Pesquisa ordenanda por ano.</i>	35
<i>Figura 3.9: Pesquisa ordenanda por trieno.</i>	35
<i>Figura 3.10: Gráfico evolutivo e pesquisa ordenanda por ano.</i>	36
<i>Figura 3.11: Pesquisa agrupada pelo tipo completo/resumo.</i>	37
<i>Figura 3.12: Pesquisa agrupada pelo tipo periódico/evento.</i>	37
<i>Figura 3.13: Gráfico de somatório dos Qualis.</i>	38
<i>Figura 3.14: Listagem de pesquisadores do sistema.</i>	39
<i>Figura 3.15: Diferenças entre telas de inserção e edição de pesquisadores.</i>	40
<i>Figura 3.16: Listagem de grupos de pesquisa.</i>	40
<i>Figura 3.17: Tela de inserção e edição de grupos de pesquisa.</i>	41
<i>Figura 3.18: Tela de listagem de área de pesquisa.</i>	42
<i>Figura 3.19: Tela de inserção e edição de área de pesquisa.</i>	42
<i>Figura 3.20: Tela de listagem das linhas de pesquisa.</i>	43
<i>Figura 3.21: Tela de inserção e edição de linha de pesquisa.</i>	44

RESUMO

De acordo com dados da FAPESP, o Brasil é o país com maior representatividade na América do Sul em relação a quantidade de publicações científicas, nos últimos anos. A elaboração e o uso de ferramentas de análise de produção científica recebem atenção crescente como instrumentos para medição dos resultados da atividade no setor científico. O Instituto de Informática da Universidade Federal do Rio Grande do Sul possui um diversificado corpo docente de professores os quais trabalham com o ensino e, ao mesmo tempo, com a pesquisa acadêmica. O propósito deste trabalho consiste no desenvolvimento de uma aplicação que possibilite a análise dos dados acadêmicos dos pesquisadores do Instituto de Informática. Esses dados são analisados com base no currículo Lattes dos pesquisadores. A aplicação permite a consulta sobre as publicações de um professor/pesquisador, grupo, ou linha de pesquisa. Também emite relatórios de consolidação, facilitando a análise dos dados.

Palavras-Chave: publicação científica, currículo Lattes, análise.

A Tool to Analyze the Scientific Production: Case Study on the Institute of Informática at UFRGS

ABSTRACT

According to data published by FAPESP in 2011, Brazil is the most notability country in South America in respect of the number of scientific paper publications in the last years. The use of analysis tools, for scientific paper publications, is getting more relevance to measure the results of the activities in scientific area. The Information Technology Department of Universidade Federal do Rio Grande do Sul has a miscellaneous set of professors which work in the graduation and, at the same time, with academic research. The purpose of this publication consists of a software development that allows analyzing the academic data of researches of Information Technology Department. This data are going to be analyzed based on researches curriculum Lattes. The system allows to seek over publications of professors/researches, group or line of research. It also print reports making the analyze easier.

Keywords: scientific publication, curriculum Lattes, analyze.

1 INTRODUÇÃO

O Brasil lidera a produção de artigos científicos em relação aos principais países da América Latina, tendo publicado 94.622 trabalhos em periódicos científicos internacionais indexados pelo Web of Science no período 2008 a 2010 (FAPESP, 20011).

A elaboração e o uso de ferramentas de análise de produção científica recebem atenção crescente como instrumentos para medição dos resultados da atividade no setor científico. A produção científica ganha importância crescente para o crescimento da ciência, tecnologia e inovação. Indicadores de produção podem contribuir, por exemplo, para a análise dos resultados da infraestrutura disponível e das políticas de investimento em pesquisa científica e tecnológica. Também são úteis na análise da dinâmica das diferentes áreas científicas, inclusive na identificação e compreensão de áreas emergentes ou consolidadas. Diversas agências governamentais nacionais e internacionais de fomento à pesquisa científica e tecnológica elaboram e utilizam indicadores de produção científica para a formulação, execução e acompanhamento de políticas públicas de ciência e tecnologia. Outro fator que contribui para a crescente atenção dada aos indicadores de produção científica é a maior disponibilidade de metodologias e recursos eletrônicos para sua elaboração, com indicadores muitas vezes pré-elaborados, acessíveis com relativa facilidade.

Esta demanda por análise de produção científica, também, ocorre dentro dos departamentos de Ciência da Computação e Engenharia da Computação, do Instituto de Informática da Universidade Federal do Rio Grande do Sul (UFRGS).

O Instituto de Informática possui um diversificado corpo docente de professores. Todos professores possuem seus trabalhos de licenciatura concomitante ao trabalho de pesquisa acadêmica. Pesquisa, Inovação e desenvolvimento tecnológico possuem papel de destaque dentro do Instituto de Informática.

Neste trabalho é proposta a implementação de um sistema que possibilite a análise dos dados acadêmicos dos pesquisadores do Instituto de Informática. Estes dados estão presentes no currículo Lattes destes pesquisadores. Esta análise será feita com buscas a um banco de dados o qual armazenará todos os currículos Lattes. Serão avaliados dados do tipo, quantidade de Qualis, publicações por linha de pesquisa ou por grupo de pesquisa, entre outras variáveis que serão analisadas nos capítulos seguintes.

O presente documento está organizado em 4 capítulos. No segundo capítulo é descrito o ambiente e as técnicas utilizadas para o desenvolvimento da aplicação, tais como, técnicas de abordagem para o desenvolvimento, requisitos funcionais e não-funcionais, componentes do sistema e arquitetura utilizada. Em seguida, no quarto capítulo, a implementação do sistema é detalhada, contemplando a modelagem do banco de dados, a organização das classes dentro da solução, análise das interfaces e da

navegação entre as interfaces. Por fim, no quinto e último capítulo, são avaliados os resultados obtidos, bem como as dificuldades encontradas durante o desenvolvimento do trabalho e sugestões de melhoria para trabalhos futuros.

2 AMBIENTE E TÉCNICAS UTILIZADAS

Nesta seção será descrito a abordagem para o desenvolvimento do sistema, as funcionalidades necessárias para a criação da aplicação. Em seguida serão apresentados e justificados os componentes utilizados na implementação da aplicação e, por último, será mostrada a arquitetura utilizada.

2.1 ABORDAGEM (OU TÉCNICAS)

Para o desenvolvimento do sistema foi utilizado uma técnica de abordagem utilizada a qual fornece maiores garantias de entrega do sistema com qualidade e dentro do prazo estipulado.

2.1.1 Engenharia da Web

Para a criação da aplicação foram analisadas algumas formas de abordagem para o desenvolvimento do sistema e a engenharia de software focada em aplicações web foi a que demonstrou maior probabilidade de sucesso e menores taxas de risco para a construção de sistemas web.

Engenharia Web propõe um modelo de fluxo de processo ágil, porém disciplinado, para a construção de aplicações web. (PRESSMAN, 2009). Negócios comerciais modernos demandam fácil adaptação as mudanças que ocorrem a todo instante, usuários finais do sistema se preocupam que a aplicação esteja disonível quando eles precisarem, mesmo mudando as regras do sistema constantemente. Um time de desenvolvimento deve focar em agilidade.

Um time ágil pode ser definido como flexível quando estiver apto a responder a mudanças. Engenharia de software se trata de mudanças. Mudanças no software que está sendo construído, mudanças nos membros do time, mudanças devido a tecnologia, mudanças de todos os tipos que podem impactar no produto ou em parte do projeto que está criando o produto. Suporte a mudanças deve ser construído em todas as partes do desenvolvimento (JACOBSON, ANO, PÁGINA).

Um modelo de processo estabelece a base para o processo completo da engenharia de software indentificando um um pequeno número de atividade aplicáveis a todo tipo de projetos. Além disso, o modelo engloba um conjunto de atividades que devem ser aplicadas durante todo o processo de desenvolvimento.

Cada atividade é composta por um conjunto de de ações, as quais são um conjunto de ações que produzem o produto, design é um tipo de ação, por exemplo. Cada ação é

composta por tarefas individuais que efetuam parte do trabalho que deve ser realizado pela ação. Este nívelamento de atividades, ações e tarefas pode ser visto na Figura 2.1 a seguir.

WebE process

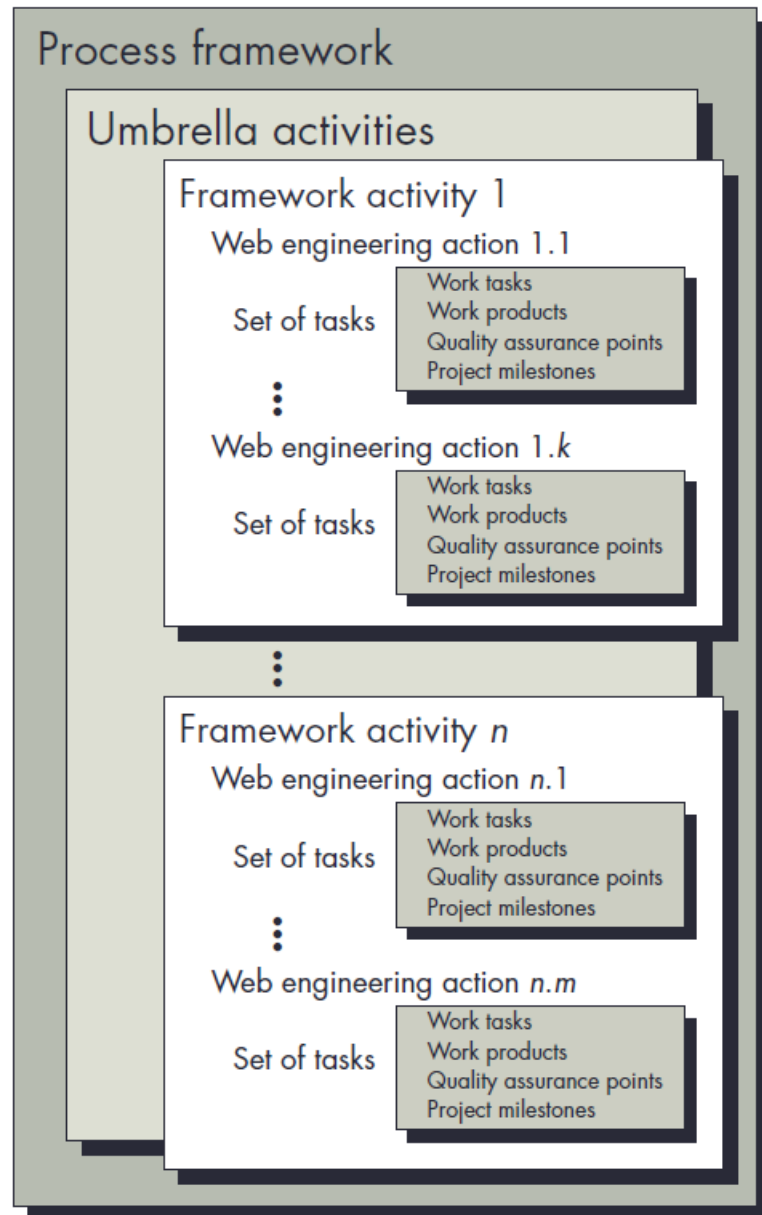


Figura 2.1: Processo da Engenharia da Web.

Fonte: Pressman (2009, Pg. 14)

As atividades a seguir são parte de um modelo de processo genérico e são aplicáveis a grande maioria dos projetos web.

- **Comunicação.** Esta atividade envolve muita interação e colaboração com o cliente – e outros usuários interessados – e engloba coleta de requisitos e outras atividades relacionadas.

- **Planejamento.** Estabelece um plano incremental – assume-se que a aplicação será entregue em uma série de incrementos que provêm, sucessivamente, robustos requisitos a cada entrega – para o projeto. Descreve as ações que irão ocorrer, as tarefas a serem realizadas, os possíveis riscos que poderão ser encontrados ao decorrer desta entrega incremental, os recursos que serão necessários, o produto que será desenvolvido e a escala de trabalho.
- **Modelamento.** Esta atividade engloba a criação de modelos que irão ajudar o desenvolvedor e o cliente na melhor compreensão dos requisitos e o design que deverá ser alcançado com estes requisitos.
- **Construção.** Corresponde a criação das páginas, com a combinação dos elementos de uma página web e que foram utilizados neste projeto, como HTML, JavaScript e .NET, já anteriormente abordados. Esta atividade também contém a fase de testes, a qual é uma etapa que tem como objetivo descobrir erros no código que não tinham sido detectados pelo desenvolvedor.
- **Publicação.** Corresponde à entrega do incremento do projeto para o cliente que irá avaliá-la e, posteriormente, fornecer um parecer baseado na sua avaliação.

Estas cinco atividades genéricas podem ser usadas durante o processo de desenvolvimento de aplicações, não importando o tamanho e complexidade desta aplicação. Atividades como gerenciamento de risco, garantia de qualidade e gerenciamento de conteúdo, são aplicadas durante todo o processo. Toda aplicação que utilizar qualquer modelo de processo deve perceber que adaptação deve ser aplicada durante todo o processo para se obtenha sucesso no projeto. Adaptação do problema, do projeto, do time de desenvolvimento e da cultura organizacional são exemplos de adaptações que devem estar presente ao desenvolver uma aplicação.

2.1.1.1 Princípios Ágeis

O modelo de processo adaptado para projetos voltados para Web deve enfatizar agilidade e seguir os doze princípios adotados pela Agile Alliance.

1. A maior prioridade é satisfazer o cliente através de entregas contínuas e valiosas – que contenham significância dentro do projeto - do software.
2. Aceitar mudanças nos requisitos, mesmo quando elas ocorrerem tardiamente no decorrer do processo de desenvolvimento.
3. Entregar incrementos do software que sejam funcionais. Que estas entregas ocorram de semanas em semanas ou de meses em meses. Dando preferência para o menor tempo para as entregas.
4. Desenvolvedores e pessoas que trabalharam na área de negócios devem trabalhar juntos diariamente durante todo o projeto.
5. Construir o projeto em volta de pessoas motivadas. Proporcionar um bom ambiente e suporte necessário, acreditar no comprometimento de entrega destas pessoas.

6. O melhor e mais eficiente modo de troca de informações para o time de desenvolvimento são conversas que se façam pessoalmente.
7. Software funcionando é a primeira medida de progresso.
8. Processo ágil estimula o desenvolvimento sustentável. Os desenvolvedores e clientes devem ser capazes de manter um ritmo constante.
9. Prestar atenção sempre em qualidade de software e um bom design aumenta a agilidade.
10. Simplicidade é essencial. Realizar entregas contínuas com incrementos que estejam livres de erros e com boa qualidade.
11. As melhores arquiteturas, os melhores requisitos e melhores designs provêm de times auto gerenciáveis.
12. Em intervalos, regulares o time reflete como pode se tornar mais efetivo e proíbe metas para alcançar novas metas.

2.2 Requisitos

A análise de requisitos tem por objetivo auxiliar os engenheiros de software a compreender melhor o problema que eles irão trabalhar para solucionar, gerando um entendimento escrito do mesmo. Isso pode ser atingido através de cenários de uso, listas de funções e de características, modelos de análise ou uma especificação (PRESSMAN, 2006).

Nas subseções seguintes serão mostrados os principais requisitos funcionais e não-funcionais elaborados para aplicação proposta. Eles são as funcionalidades básicas para o bom funcionamento do sistema desenvolvido.

2.2.1 Requisitos Não-Funcionais

Abaixo seguem os requisitos não-funcionais da aplicação:

- O sistema deve ser disponibilizado livremente;
- O sistema deve estar disponibilizado para rodar em uma máquina qualquer, estando ela na Web, em um servidor, ou não;
- O sistema deve fazer uso de um banco de dados.

2.2.2 Requisitos Funcionais

Abaixo seguem os requisitos funcionais da aplicação:

- A utilização do sistema deve ser livre da necessidade de realizar qualquer tipo de cadastro;
- O sistema deve possibilitar a busca e a listagem de publicações de pesquisadores do Instituto de Informática da UFRGS;
- Deve ser possível ordenar o resultado por período, área ou linha de pesquisa;

- Na busca ou listagem deve ser possível visualizar e contabilizar os QUALIS de cada publicação, de acordo com o tipo da publicação e ano;
- O sistema deve possibilitar a filtragem por tipo de publicação;
- A busca ou listagem pode ser feita baseada em pesquisadores contidos em grupos ou linhas de pesquisa.
- O sistema deve possibilitar a inserção e edição de pesquisadores;
- O sistema deve possibilitar a inserção e edição de grupos de pesquisa;
- O sistema deve possibilitar a inserção e edição de linhas de pesquisa;
- O sistema deve possibilitar o upload de novas versões do QUALIS;
- O sistema deve possibilitar o upload de XML Lattes contendo a lista de publicações dos usuários.

2.3 Componentes do Sistema

Nesta seção são apresentados os componentes que se fizeram necessários para que o sistema atingisse os requisitos abordados na seção anterior. Primeiramente, é abordado o framework utilizado. Em seguida são demonstrados os motivos da escolha do SGBD (Sistema de Gerenciamento de Banco de Dados). E por último os elementos de uma página Web e como eles interagem entre si.

2.3.1 .Net Framework

O framework .NET é uma plataforma desenvolvida pela Microsoft que suporta desenvolvimento e execução de sistemas e aplicações. Toda aplicação desenvolvida para ser executada pelo .NET framework é executada sobre um ambiente de execução independente de linguagem (Common Language Runtime - CLR). Esta CLR, atualmente, é capaz de executar mais de 30 linguagens de programação. Neste trabalho, a linguagem utilizada foi o C#. O .NET provém, também, uma vasta biblioteca de classes.

A figura 2.2 ilustra os recursos do .NET framework.

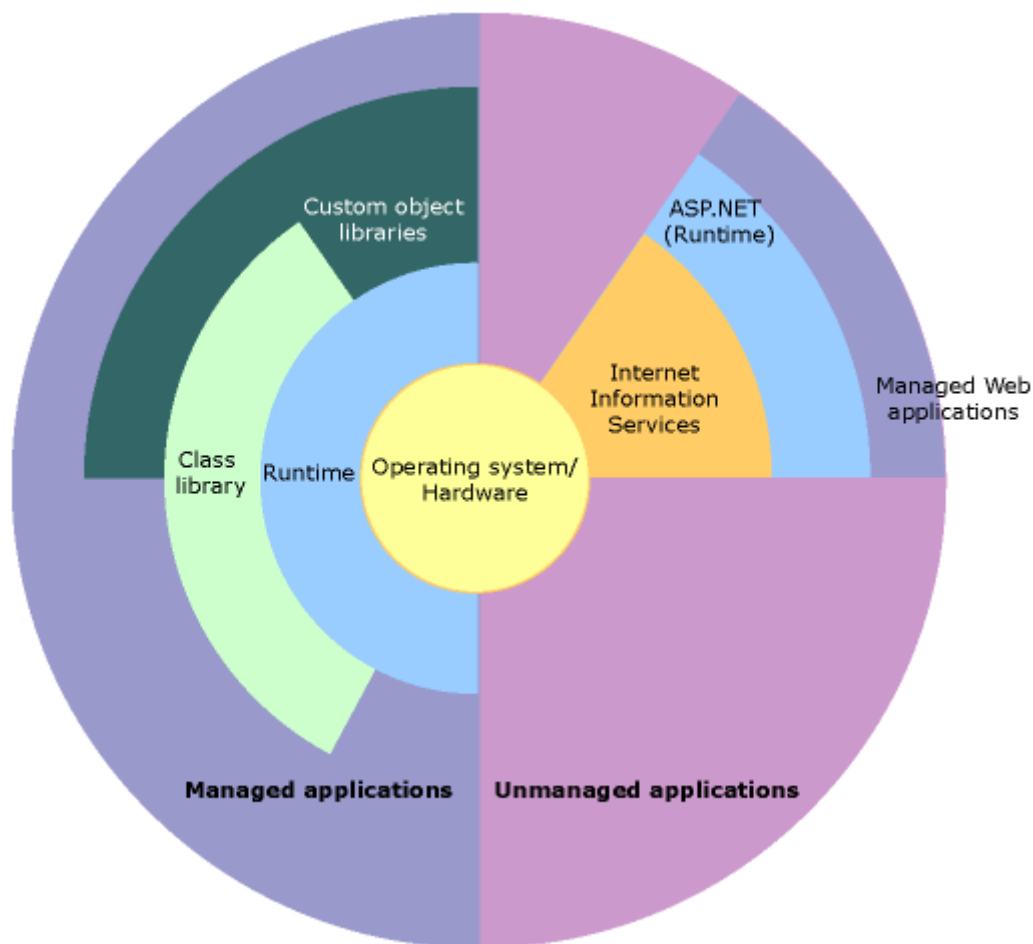


Figura 2.2: Componentes do .NET Framework.

Fonte: DEVELOPER NETWORK (2014)

2.3.2 Banco de Dados

O SGBD utilizado foi o SQL Server. SQL Server foi desenvolvido pela Microsoft e há diversas edições destinadas a diferentes públicos e diferentes cargas. Na aplicação foi utilizado a edição SQL Server 2008 R2 Express Edition.

Segundo especificação da Microsoft o SQL Server 2008 R2 foi desenvolvido sobre uma versão do SQL Server 2008 a qual possui alta confiabilidade e é de fácil gerenciamento, ele deriva com uma maior escalabilidade e com serviços de relatórios e dados análises. Desenvolvido, especificamente, para prover serviços para organizações que hospedam aplicações Web, o SQL Server 2008 R2 oferece um ótimo balanço entre os serviços de escalabilidade e manutenção. Ele não possui restrições quanto ao tamanho do banco de dados.

O autor optou por utilizar esta ferramenta de banco de dados para o repositório pela sua experiência com a mesma, pela fácil integração com o .NET Framework, pois ambos são desenvolvidos pela Microsoft e por haver uma vasta documentação.

2.3.2.1 LINQ

A fim de otimizar o desenvolvimento da aplicação foi utilizada a Language Integrated Query (LINQ). LINQ é uma tecnologia do .NET Framework que provém um mecanismo de suporte a nível de linguagem para consultar qualquer tipo de dados. As consultas são integradas a linguagem, isto resolve o problema de detecção de erro em uma sintaxe do SQL somente em tempo de processamento do código. Além de facilitar ao adicionar recursos que facilitam o desenvolvimento de consultas, LINQ prove suporte na formatação e validação dos dados.

2.3.2.2 LINQ – Sintaxe

Existem duas sintaxes possíveis para escrever uma expressão com LINQ, são elas:

- Expressão padrão (usa expressões Lambda):
stringList.Where(s => s.Length == 3).Select(s => s)
- Expressão de consulta:
from str in strings where str.Length == 3 select str

A mesma sintaxe acima pode ser aplicada a um banco de dados relacional, DataSets, arquivos XML ou qualquer outra coleção de dados que seja do tipo IEnumerable ou IQueryable.

2.3.2.3 LINQ – Como Funciona

LINQ funciona com tradução sintática (syntactic sugar). O compilador obtém a consulta, no formato que foi demonstrado acima, e transforma para outra sintaxe que possui funções, métodos e objetos. As principais operações são de tradução sintática que irão transformar as consultas para linguagem de máquina.

Ele irá operar sobre a coleção que irá sofrer as ações, para cada tipo de operação existe um método relativo para o tipo de coleção que está sofrendo a ação. Sobre o resultado de cada ação será retornado um elemento dos tipos IEnumerable ou IQueryable, os quais podem sofrer novas ações.

A figura 2.3 a seguir mostra uma visão geral da arquitetura de LINQ.

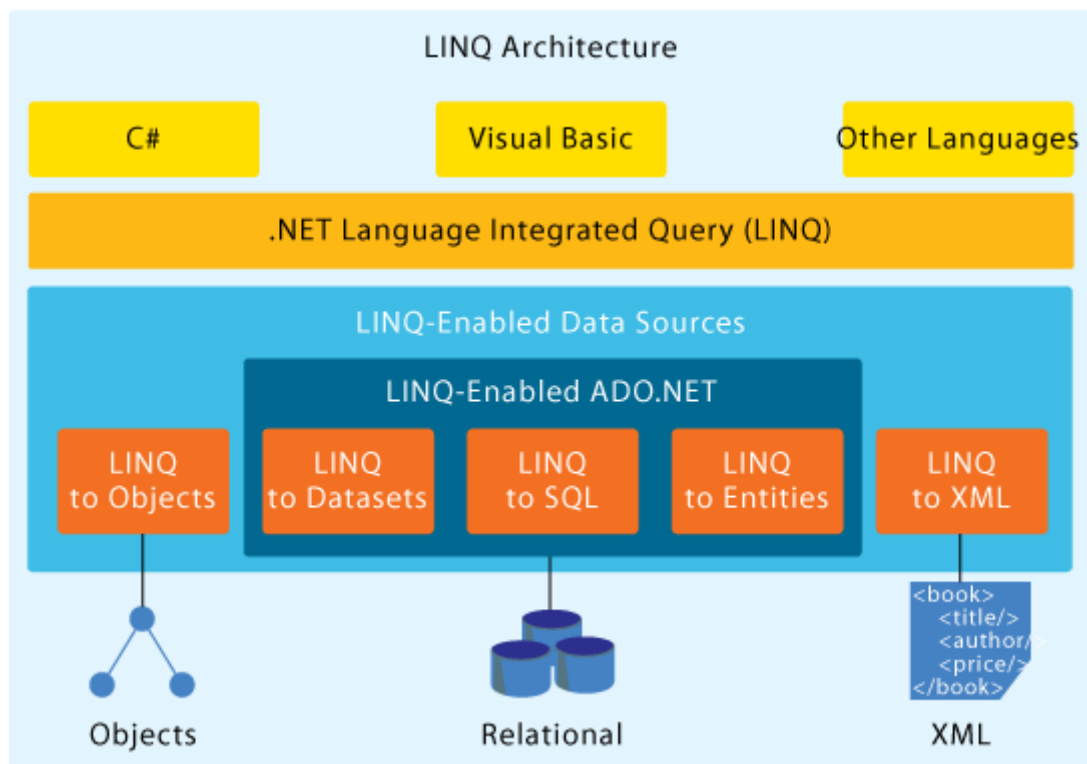


Figura 2.3: Arquitetura do LINQ.

Fonte: MSDN MAGAZINE (2014)

2.3.3 HTML

Hyper Text Markup Language (HTML) é a principal linguagem de marcação estruturada utilizada para a construção de páginas WEB. As marcações indicam aos Navegadores como as imagens e conteúdos da página WEB. HTML é recomendado pela World Wide Web Consortium (W3C) e é interpretado pelos maiores navegadores. Na aplicação, será usado para denotar os elementos da página, tais como caixas de seleção, caixas de texto, botões, etc.

2.3.4 CSS

Cascading Style Sheet é uma folha de estilo utilizada para descrever o layout de páginas WEB. É uma linguagem de estilos que define a apresentação de documentos escritos em uma linguagem de marcação. É normalmente utilizada para definir o layout de páginas em HTML, porém pode ser utilizada para qualquer linguagem de marcação. Na aplicação é utilizada para definir cores, formato das letras, etc.

2.3.5 JavaScript

JavaScript é uma linguagem de programação interpretada a qual é executada pelos navegadores. Tem como principal objetivo ser executada do lado do cliente para que o usuário possa interagir, com a página WEB, sem que o fluxo da aplicação tenha que passar pelo lado do servidor. JavaScript é assíncrono e tem controle sobre os objetos de uma página WEB tornando-a mais dinâmica, pois é capaz de alterar o conteúdo da página.

A figura 2.4 ilustra o comportamento de HTML, CSS e JavaScript na aplicação.



Figura 2.4: Interação entre JavaScript, HTML e CSS.

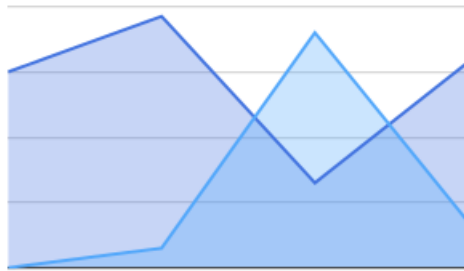
Fonte: Criada pelo autor

2.3.5.1 Google Charts

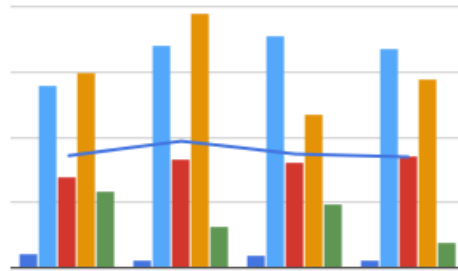
Para a criação de alguns gráficos da aplicação, tais como, gráfico evolutivo da quantidade de publicações realizadas e gráfico de contagem de qualis, foi utilizado a API do Google Chart. Esta API cria, dinamicamente, gráficos e tabelas que são facilmente anexadas a páginas HTML. É uma API gratuita, desde que usado de acordo com os termos da Google. Não possuem componentes, a serem instalados no lado cliente ou no lado servidor, é feita apenas uma requisição ao serviço de Charts da Google que retorna uma imagem do gráfico, o que contribui com o não processamento do gráfico, pois ele é feito pelos serviços da Google.

A Figura .25 ilustra alguns exemplos de gráficos fornecidos pela API.

Area Chart



Combo Chart



Geo Chart



Scatter Chart

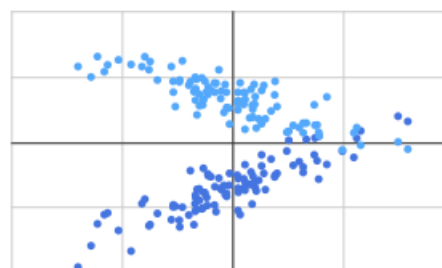


Figura 2.5: Gráficos disponibilizados pelo Google Charts.

Fonte: GOOGLE DEVELOPERS (2014).

2.4 Arquitetura

Nesta seção será abordada a arquitetura do sistema proposto. A aplicação desenvolvida é baseada na arquitetura Cliente-Servidor e segue o padrão MVC como modelo estrutural, ambos detalhados nas subseções que seguem.

2.4.1 Cliente-Servidor

O modelo Cliente-Servidor é uma estrutura de aplicação distribuída composta por dois componentes chamados Cliente e Servidor que se comunicam através de uma rede de computadores. A função do Servidor é fornecer recursos ou serviços ao Cliente e, para tanto, deve receber requisições de uma ou mais instâncias de um Cliente, processá-las e enviar uma resposta baseada na requisição. Os Clientes apenas geram requisições e, após enviá-las, aguardam pela resposta (TANEMBAUM, 2003).

A figura 2.6 ilustra o comportamento de uma aplicação Cliente-Servidor.

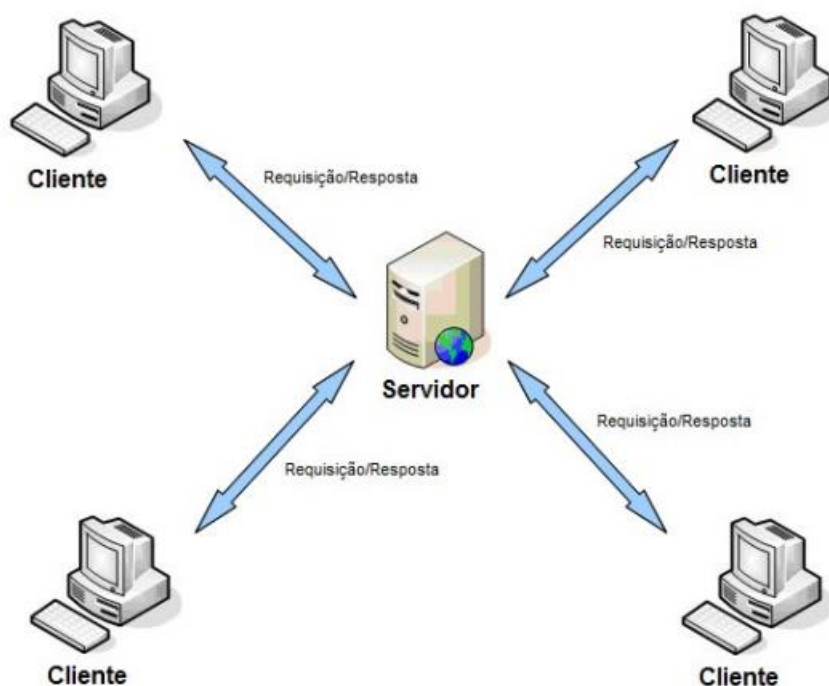


Figura 2.6: Modelo Cliente-Servidor.

Fonte: Nunes, 2013.

No sistema proposto neste trabalho, o navegador web pelo qual o usuário irá acessar o repositório faz o papel de Cliente que interagirá com o Servidor através do protocolo HTTP, requisitando serviços disponíveis na aplicação como efetuar uma busca ou adicionar um novo objeto ao repositório.

2.4.2 Model-View-Controller

Model-view-controller (MVC) é um padrão arquitetural que consiste em dividir as responsabilidades da aplicação em três tipos de componentes. O Modelo representa o objeto da aplicação, a Visualização representa as telas do sistema e o Controlador define o caminho que a interface irá seguir conforme uma ação do usuário. Esta abordagem reduz a dependência entre o Modelo e as Visualizações, aumentando a flexibilidade e o uso do código (GAMMA, HELM, JOHNSON, VLISSIDES, 1995).

Na arquitetura MVC o modelo representa os dados da aplicação e as regras do negócio que governam o acesso e a modificação dos dados. O modelo mantém o estado persistente do negócio.

Um componente de visualização renderiza o conteúdo de uma parte particular do modelo e encaminha para o controlador as ações do usuário. Ele, também, acessa os dados do modelo via controlador e define como esses dados devem ser apresentados.

Um controlador define o comportamento da aplicação. É ele que interpreta as ações do usuário e as mapeia para chamadas do modelo. Em um cliente de aplicações Web essas ações do usuário poderiam ser cliques de botões ou seleções de menus. As ações realizadas pelo modelo incluem ativar processos de negócio ou alterar o estado do modelo. Com base na ação do usuário e no resultado do processamento do modelo, o controlador seleciona uma visualização a ser exibida como parte da resposta a solicitação do usuário. Há normalmente um controlador para cada conjunto de funcionalidades relacionadas.

A Figura 2.7 ilustra as interações entre os componentes do MVC.

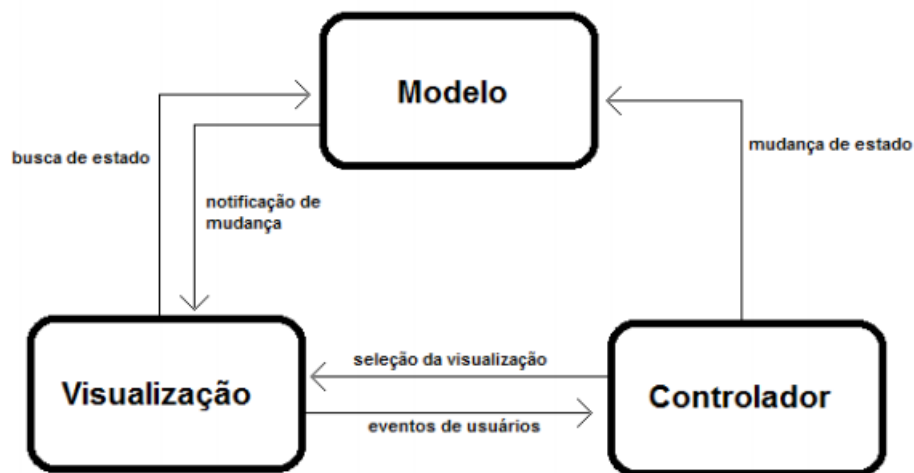


Figura 2.7: Interação entre os componentes do MVC.

Fonte: Nunes, 2013.

2.4.2.1 Model

A Figura 2.8 ilustra as classes que representam o componente Modelo do padrão MVC e a estrutura de organização de pastas dentro da aplicação.

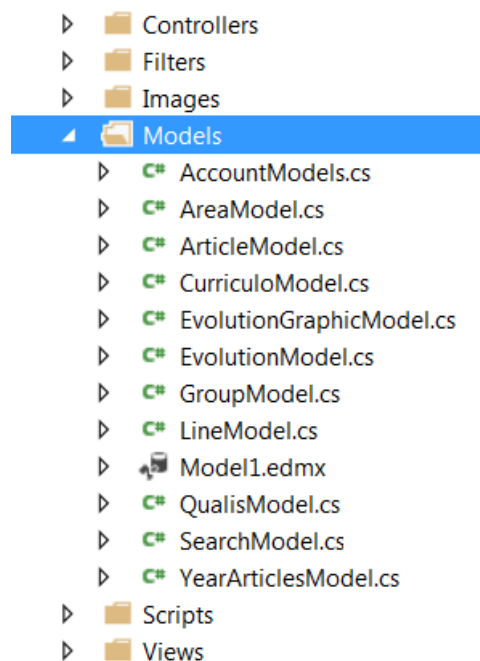


Figura 2.8: Organização de pastas e classes do Modelo da aplicação.

Fonte: Imagem criada pelo autor.

Como pode ser observado na Figura 2.8 a pasta **Models** encapsula todos os documentos do tipo **model** os quais contêm os dados e as classes que serão manipulados pelas **controllers** e mapeada para um elemento de visualização do compenten **View**, permitindo a exibição dessas informações para o usuário.

Cada classe desta pasta não representa, diretamente, uma tabela do banco de dados. As classes as quais correspondem às tabelas estão contidas no documento Model1.edmx. Este é o documento que contém todos os entities que são gerados pelo Entity Framework do .Net que representam cada tabela do banco de dados.

2.4.2.2 Controller

A Figura 2.9 ilustra as classes que representam o componente Controller do padrão MVC na aplicação desenvolvida.

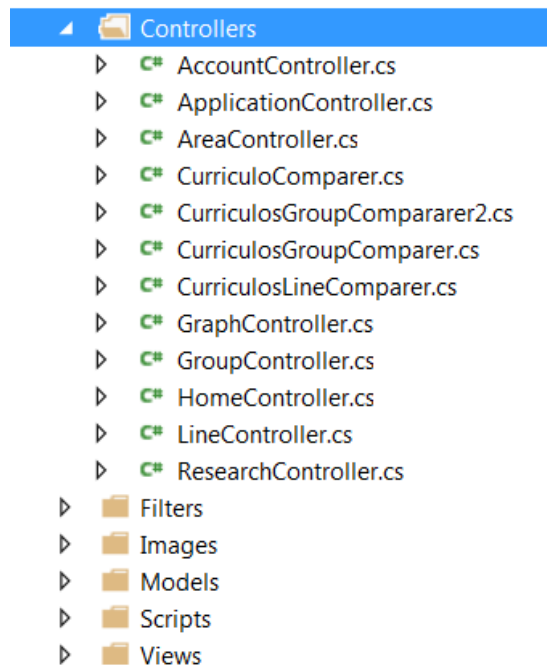


Figura 2.9: Classes do componente Controller da aplicação.

Fonte: Imagem criada pelo autor.

Como pode ser observado na Figura 2.9 a organização de pastas dentro do componente Controller é muito simples. Ele só contém as classes as quais são usadas pelas aplicação com o propósito de identificar as ações do usuário e executar a operação por ele desejada utilizando as informações de algum elemento da Model e repassar o elemento da View correspondente a esta ação. Assim, o usuário obterá a resposta desejada que foi manipulada pelo componente do elemento Controller.

2.4.2.3 View

A Figura 2.10 ilustra as páginas, que são documento no formato HTML, as quais representam o componente View do padrão MVC na aplicação desenvolvida e a organização de pastas dentro deste componente.

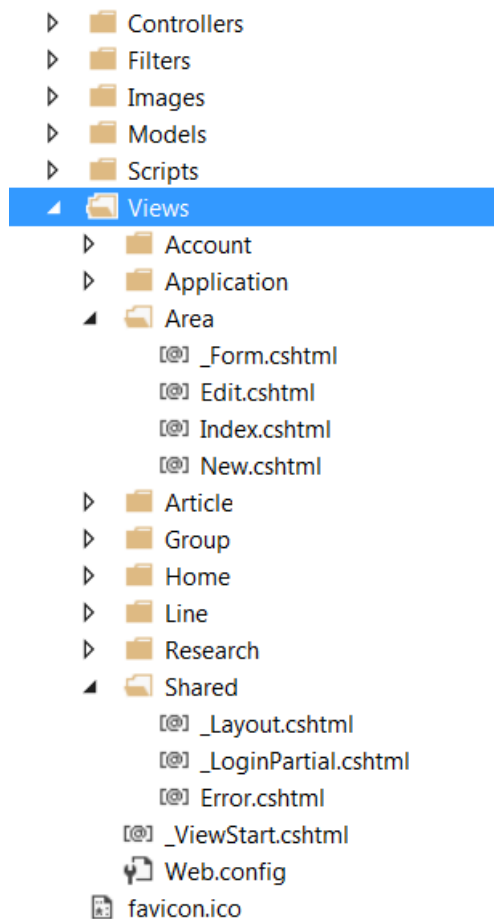


Figura 2.10: Páginas do componente View e organização de pastas.

Fonte: Imagem criada pelo autor.

A pasta Views contém subpastas que correspondem a cada classe do elemento Controller. Todas as subpastas contém páginas ou parte de páginas que servem para formar um elemento de visualização para o usuário, ou seja, uma tela final. Por exemplo, na subpasta Area possui a tela de Index que serve para listar as áreas contidas no sistema, porém também contém o documento _Form.cshtml que é um formulário usado tanto na criação quanto na edição de uma área. Assim, não é uma página completa. E a subpasta Shared contém o layout geral da aplicação ela sempre é utilizada para qualquer tela do sistema.

3 DESENVOLVIMENTO DO SISTEMA

No capítulo anterior foram apontados os principais requisitos para o desenvolvimento do sistema proposto, que implementasse análises baseado em um repositório que armazene os currículos lattes dos professores do Instituto de Informática da UFRGS. Também foram vistos os componentes utilizados no desenvolvimento deste sistema, sua arquitetura e as atividades adotadas como forma de abordagem para o desenvolvimento da aplicação.

Este capítulo detalha a solução abordada do sistema proposto. Primeiramente, será apresentada a estrutura da base de dados utilizada na aplicação. Posteriormente, será abordado a estrutura de classes resultante desse modelo. E por fim será mostrado imagens das telas de navegação do sistema apresentando uma análise de toda a aplicação.

3.1 Modelagem do Banco de Dados

Com o objetivo de manter os dados, utilizados pela aplicação, em somente um banco de dados foi utilizado apenas um SGBD, o qual foi abordado no capítulo anterior. Sendo assim, a modelagem inclui os currículos lattes dos pesquisadores no formato XML e as relações entre linha e grupo de pesquisa com os pesquisadores

O diagrama ER (Entidade-Relacionamento) da base de dados modelado para o sistema proposto está ilustrado na Figura 3.1. Cada entidade e seus respectivos atributos estão descritos nas subseções seguintes.

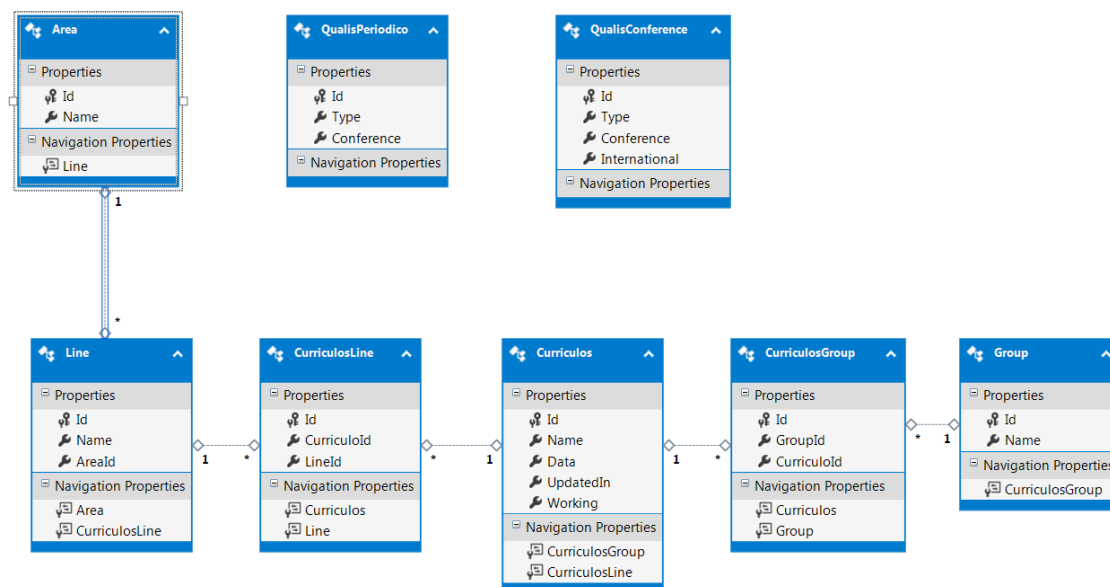


Figura 3.1: Diagrama ER do banco de dados do sistema.

3.1.1 Entidade Currículos

Esta entidade é a principal do sistema. Ela contém as informações referentes de todos os pesquisadores cadastrados no sistema. Estas informações são utilizadas para realizar as pesquisas e aplicar os filtros que irão retornar as publicações que o usuário deseja encontrar.

Esta tabela possui um campo como identificador único para cada objeto, este identificador é também uma chave estrangeira para as entidades *CurriculosGroup* e *CurriculosLine*. Possui também um campo para o nome do pesquisador, um para informar a última vez que foi atualizado o currículo do pesquisador, um campo para informar se o pesquisador ainda está ativo junto ao Instituto de Informático. E por fim um campo onde é armazenado o currículo lattes do pesquisador no formato XML, este é campo mais importante desta tabela, visto que todas as pesquisas, em algum momento, utilizam o currículo dos professores como fonte.

3.1.2 Entidade Group

Esta entidade contém os dados dos grupos de pesquisa cadastrados no sistema. Esta tabela possui um campo identificador único para os objetos contidos nela. Este campo é, também, uma chave estrangeira para a tabela `CurriculosGroup`. Esta entidade ainda possui o campo `nome` para cada objeto.

3.1.3 Entidade CurrículosGroup

Esta entidade possui as relações entre as entidades Currículos e Group. Ela possui um identificador único para cada objeto e os identificadores únicos das entidades Currículos e Group. Cada grupo pode conter um núemro n de pesquisadores que pertencem ao grupo e cada pesquisador pode connter um número n de grupos os quais ele pertence.

3.1.4 Entidade Line

Esta entidade contém os dados das linhas de pesquisa cadastradas no sistema. Esta tabela possui um campo identificador único para os objetos contidos nela. Este campo é, também, uma chave estrangeira para a tabela CurruculosLine. Esta entidade também possui o campo nome para cada objeto. Ela ainda possui um campo indentificador de área, pois cada linha de pesquisa tem uma área de pesquisa relacionada.

3.1.5 Entidade CurruculosLine

Esta entidade possui as relações entre as entidades Curruculos e Line. Ela possui um identificador único para cada objeto e os identificadores únicos das entidades Curruculos e Line. Cada grupo pode conter um númeo n de pesquisadores que pertencem a linha e cada pesquisador pode connter um número n de linhas os quais ele pertence.

3.1.6 Entidade Area

Esta entidade contém os dados das áreas de pesquisa cadastradas no sistema. Esta tabela possui um campo identificador único para os objetos contidos nela. Este campo é, também, uma chave estrangeira para a tabela Line. Esta entidade ainda possui o campo nome para cada objeto.

3.1.7 Entidade QualisPeriodico

Esta entidade contém os dados dos QUALIS por periódicos cadastradas no sistema. Esta tabela possui um campo identificador único para os objetos contidos nela. Além disso, possui um campo para o título periódico e para o tipo de QUALIS.

3.1.8 Entidade QualisConference

Esta entidade contém os dados dos QUALIS por periódicos cadastradas no sistema. Esta tabela possui um campo identificador único para os objetos contidos nela. Além disso, possui um campo para o título periódico e outro campo para o tipo de QUALIS, por fim, ainda possui um campo para identificação se a conferência é nacional ou internacional.

3.2 O Sistema

Nesta seção serão abordadas as telas do sistema e como ocorre a navegação entre elas. Para o desenvolvimento do sistema foi adotado um layout padrão que será notado nas figuras das telas do sistema que serão mostradas nas subseções seguintes. Para acessar qualquer tela do sistema não é necessário efetuar cadastro e/ou login no sistema.

3.2.1 Tela Inicial

A página inicial é a primeira visão do sistema exibida para o usuário ao acessar a URL onde está hospedado a aplicação.

Análise Produção Científica

Inicial Aplicação Pesquisadores Grupos Linhas de Pesquisa Areas de Pesquisa

Ferramenta de Análise de Produção Científica. O propósito desta aplicação consiste no desenvolvimento de uma aplicação que possibilite a análise dos dados acadêmicos dos pesquisadores do Instituto de Informática.

Esses dados são analisados com base no currículo Lattes dos pesquisadores. A aplicação permite a consulta sobre as publicações de um professor/pesquisador grupo, ou linha de pesquisa. Também emite relatórios de consolidação, facilitando a análise dos dados.

Funcionalidades:

Professor/Pesquisador

Funcionalidade que permite que o cadastro e edição de pesquisadores. Ver mais..

Pesquisa

Funcionalidade que permite que a pesquisa por pesquisadores, linha e grupo de pesquisas. Além disso, é possível contabilizar a quantidade de quais das publicações. Ver mais..

Linhas/Grupo de Pesquisa

Funcionalidade que permite cadastro e edição dos grupos e linhas de pesquisa. Grupos e Linhas.

© 2014 - Desenvolvido por Jean Lucas Lima

Figura 3.2: Tela inicial do sistema.

Fonte: Imagem criada pelo autor.

Como pode ser observado na Figura 3.2 a tela inicial tem como objetivo informar ao usuário as informações básicas do sistema ao qual ele está acessando, tais como o objetivo da aplicação que é fornecer informações referentes as publicações dos pesquisadores que possuem vínculo com o Instituto de Informática da UFRGS. Ela contém, também, informações para entrar em contato com o Instituto de Informática e do criador do sistema.

3.2.2 Tela de Pesquisa

A tela de pesquisa é a principal página do sistema. É nesta página que o usuário poderá realizar todas as pesquisas necessárias para buscar informações referentes as publicações dos pesquisadores cadastrados no sistema.

Como pode ser observado na Figura 3.3 a pesquisa pode ser realizada de diversas formas com a combinação das opções de pesquisa disponibilizadas na página. Após selecionar as opções desejadas para a pesquisa o usuário deve clicar no botão 'Filtrar', o qual irá submeter o formulário e o usuário será redirecionado para a mesma tela, porém com as informações da busca desejada e as opções, escolhidas anteriormente, ainda selecionadas. A tela possui, também, um botão 'Desfazer', este botão tem como objetivo desfazer as opções de pesquisa que haviam sido selecionadas pelo usuário. Cada opção da pesquisa tem um comportamento que será explicado a seguir.

Palavra Chave

Pesquisar por

☐ Pesquisador ☐ Linha ☐ Grupo

Ordenar por

☐ Ano ☐ Trieno ☐ Evolução

Agrupar por

☐ Completo/Resumo ☐ Periódico/Evento

Contar por Qualis:

☐ Todos que contêm ☐ Todos que não contêm ☐ Todos detalhadamente ☐ A1 ☐ A2 ☐ B1 ☐ B2 ☐ B3 ☐ B4 ☐ B5 ☐ C

Filtrar

Desfazer

© 2014 - My ASP.NET MVC Application

Figura 3.3: Tela de Pesquisa.

Fonte: Imagem criada pelo autor.

3.2.2.1 Palavra Chave

A opção ‘Palavra Chave’ é um campo onde o usuário pode escrever o nome de um pesquisador para encontrar publicações que tenham a autoria vinculada a este pesquisador. Este campo funciona associado ao filtro de ‘Pesquisar por’, pois se as opções ‘Linha’ e ‘Grupo’ não forem selecionadas a busca será feita utilizando a opção ‘Pesquisador’, assim, assumindo que a pesquisa deverá ser realizada em relação ao nome dos pesquisadores cadastrados no sistema. Este comportamento é ilustrado na Figura 3.5. A pesquisa é feita de forma não sensível, isto é, não terá relevância a forma como o usuário escrever o nome do pesquisador, se com letra maiúscula no início ou se não constar letra maiúscula no início do nome. Por exemplo, a busca por ‘Paulo’ ou ‘paulo’ resultará o mesmo resultado. A busca pode ser feita não só pelo nome, mas também pelos sobrenomes dos pesquisadores. Porém, é de extrema importância que a grafia esteja correta para que seja possível a busca pelo nome do autor. Uma pesquisa feita com o sobrenome ‘Menezes’ não resultará no mesmo resultado feito com o sobrenome ‘Meneses’, por exemplo.

Como pode ser observado na Figura 3.4 o resultado da pesquisa é uma lista de publicações onde consta o nome do autor, o título da publicação e a classificação do Qualis, se possuir.

Palavra Chave

Pesquisar por
☒ Pesquisador ☐ Linha ☐ Grupo

Ordenar por
☐ Ano ☐ Trieno ☐ Evolução

Agrupar por
☐ Completo/Resumo ☐ Periódico/Evento

Contar por Qualis:
☐ Todos que contêm ☐ Todos que não contêm ☐ Todos detalhadamente ☐ A1 ☐ A2 ☐ B1 ☐ B2 ☐ B3 ☐ B4 ☐ B5 ☐ C

Total: 87 publicações(s) - Artigos: 9 - Eventos: 78

	Autor	Título	Qualis
<input type="checkbox"/>	Renata Galante	Temporal and Versioning Model for Schema Evolution in Object-Oriented Databases	A2
<input type="checkbox"/>	Renata Galante	Type-safe Versioned Object Query Language. (Versão estendida do artigo publicado no SBLP 2006)	Não possui
<input type="checkbox"/>	Renata Galante	TVX - Time and Versions in XML	Não possui
<input type="checkbox"/>	Renata Galante	XML: Some Papers in a Haystack	A1
<input type="checkbox"/>	Renata Galante	Supporting Temporal Queries on XML Keyword Search Engines	B3

Figura 3.4: Pesquisa utilizando uma palavra chave.

Fonte: Imagem criada pelo autor.

Palavra Chave

Pesquisar por
☐ Pesquisador ☐ Linha ☐ Grupo

Ordenar por
☐ Ano ☐ Trieno ☐ Evolução

Agrupar por
☐ Completo/Resumo ☐ Periódico/Evento

Contar por Qualis:
☐ Todos que contêm ☐ Todos que não contêm ☐ Todos detalhadamente ☐ A1 ☐ A2 ☐ B1 ☐ B2 ☐ B3 ☐ B4 ☐ B5 ☐ C

Total: 87 publicações(s) - Artigos: 9 - Eventos: 78

	Autor	Título	Qualis
<input type="checkbox"/>	Renata Galante	Temporal and Versioning Model for Schema Evolution in Object-Oriented Databases	A2
<input type="checkbox"/>	Renata Galante	Type-safe Versioned Object Query Language. (Versão estendida do artigo publicado no SBLP 2006)	Não possui

Figura 3.5: Pesquisa sem selecionar filtro ‘Pesquisar por’.

Fonte: Imagem criada pelo autor.

3.2.2.2Pesquisar por

O filtro de ‘Pesquisar por’ é o que define qual será o tipo de pesquisa que o sistema irá realizar. Ele é fundamental para todas as buscas que forem feitas na página de pesquisa. Ele funciona concomitante com a opção de ‘Palavra Chave’. Quando não selecionado ele assume que a pesquisa por publicações deverá ser feita em relação ao

nome do autor, como já demonstrado anteriormente. Porém, ainda há outras duas opções para se realizar a pesquisa:

- **Linha** – Ao selecionar a opção Linha uma nova opção de filtro irá aparecer para o usuário, logo abaixo do filtro ‘Pesquisar por’. Esta nova opção contém um campo o qual lista todas as Linhas de Pesquisa cadastradas no sistema, que deverão ser as Linhas de Pesquisa do Instituto de Informática. Após selecionar a Linha de Pesquisa desejada a pesquisa retornará todas as publicações dos pesquisadores que estão inclusos na Linha de Pesquisa selecionada pelo usuário, anteriormente. A Figura 3.6 demonstra uma busca feita utilizando o campo Linha.

Palavra Chave

Pesquisar por

☐ Pesquisador ☒ Linha ☐ Grupo

Selecione a Linha de Pesquisa

Ordenar por

☐ Ano ☐ Trieno ☐ Evolução

Agrupar por

☐ Completo/Resumo ☐ Periódico/Evento

Contar por Qualis:

☐ Todos que contêm ☐ Todos que não contêm ☐ Todos detalhadamente ☐ A1 ☐ A2 ☐ B1 ☐ B2 ☐ B3 ☐ B4 ☐ B5 ☐ C

Total: 348 publicações(s) - Artigos: 56 - Eventos: 292

	Autor	Título	Qualis
<input type="checkbox"/>	Álvaro Freitas Moreira	Provind BDI Properties of Agent Oriented Programming Languages	B1
<input type="checkbox"/>	Álvaro Freitas Moreira	Temporal and Versioning Model for Schema Evolution in Object-Oriented Databases	A2
<input type="checkbox"/>	Álvaro Freitas Moreira	Proving the Asymmetry Thesis Principles for a BDI Agent-Oriented Programming Language	B1
<input type="checkbox"/>	Álvaro Freitas Moreira	MAS-SOC: a Social Simulation Platform Based on Agent-Oriented Programming	B2
<input type="checkbox"/>	Álvaro Freitas Moreira	Type-safe Versioned Object Query Language	Não possui

Figura 3.6: Pesquisa por Linha de Pesquisa.

Fonte: Imagem criada pelo autor.

- **Grupo** - Ao selecionar a opção Grupo, similar ao que acontece com a opção Linha, uma nova opção de filtro irá aparecer para o usuário, logo abaixo do filtro ‘Pesquisar por’. Esta nova opção contém um campo o qual lista todos os Grupos de Pesquisa cadastrados no sistema os quais deverão ser os Grupos de Pesquisa que atuam no Instituto de Informática. Após selecionar o Grupo de Pesquisa desejado a pesquisa mostrará todas as publicações dos pesquisadores que fazem parte do Grupo de Pesquisa selecionado pelo usuário, anteriormente. A Figura 3.7 demonstra uma busca feita utilizando o campo Grupo.

Palavra Chave

Pesquisar por
☐ Pesquisador ☐ Linha ☒ Grupo

Selecione o Grupo:

Ordenar por
☐ Ano ☐ Trieno ☐ Evolução

Agrupar por
☐ Completo/Resumo ☐ Periódico/Evento

Contar por Qualis:
☐ Todos que contém ☐ Todos que não contém ☐ Todos detalhadamente ☐ A1 ☐ A2 ☐ B1 ☐ B2 ☐ B3 ☐ B4 ☐ B5 ☐ C

Total: 399 publicações(s) - Artigos: 62 - Eventos: 337

	Autor	Título	Qualis
<input type="checkbox"/>	Fernanda Gusmão de Lima Kastensmidt	Synthesis of an 8051-like Micro-Controller Tolerant to Transient Faults	B2
<input type="checkbox"/>	Fernanda Gusmão de Lima Kastensmidt	A Multiple Bit Upset Tolerant SRAM Memory	B2
<input type="checkbox"/>	Fernanda Gusmão de Lima Kastensmidt	Designing Fault-tolerant Techniques for SRAM-based FPGAs	A2
<input type="checkbox"/>	Fernanda Gusmão de Lima Kastensmidt	An Automatic Technique for Optimization of Reed-Solomon Codes to Improve Fault-tolerance in Memories	A2
<input type="checkbox"/>	Fernanda Gusmão de Lima Kastensmidt	Desenvolvimento de Técnicas de Tolerância a Falhas para Componentes Programáveis por SRAM	Não possui

Figura 3.7 Pesquisa por Grupo de Pesquisa.

Fonte: Imagem criada pelo autor.

3.2.2.3 Ordenar por

A opção de ‘Ordenar por’ não deve, necessariamente, ser definida como acontece com a opção de ‘Pesquisar por’. Quando não escolhida as publicações retornam na ordem que a aplicação encontrar dentro dos currículos Lattes dos pesquisadores filtrados pela pesquisa. Porém, quando selecionado uma das opções a aplicação possui comportamentos diferentes para cada uma das três opções. As opções existentes são:

- Ano - Quando selecionada a opção Ano as publicações resultantes da pesquisa serão ordenadas em ordem crescente de acordo com o ano de publicação. A Figura 3.8 ilustra as publicações ordenadas pelo ano.
- Trieno - Quando selecionada a opção Trieno serão retornadas todas as publicações que tenham sido publicadas no último trieno, ou seja, que tenham sido publicadas nos últimos três anos. Estas publicações serão ordenadas em ordem crescente de acordo com o ano de publicação. A Figura 3.9 ilustra as publicações ordenadas pelo ano.
- Evolução – Quando selecionada a opção Evolução irão aparecer dois novos filtros, ‘Evolução por’ – com as opções Ano e Trieno – e Intervalo com as opções – três (3) e cinco (5). A combinação destes dois novos filtros será utilizada para selecionar o período de tempo que será usado como limite para mostrar os dados em um gráfico. Este gráfico tem como objetivo mostrar uma análise evolutiva em relação ao número de publicações por ano. Além do gráfico evolutivo as publicações resultantes da pesquisa retornarão ordenadas em ordem crescente de acordo com o ano de publicação do

mesmo modo que acontece na opção Ano, como acontece na opção Ano. A Figura 3.10 ilustra o gráfico evolutivo e as publicações ordenadas por ano.

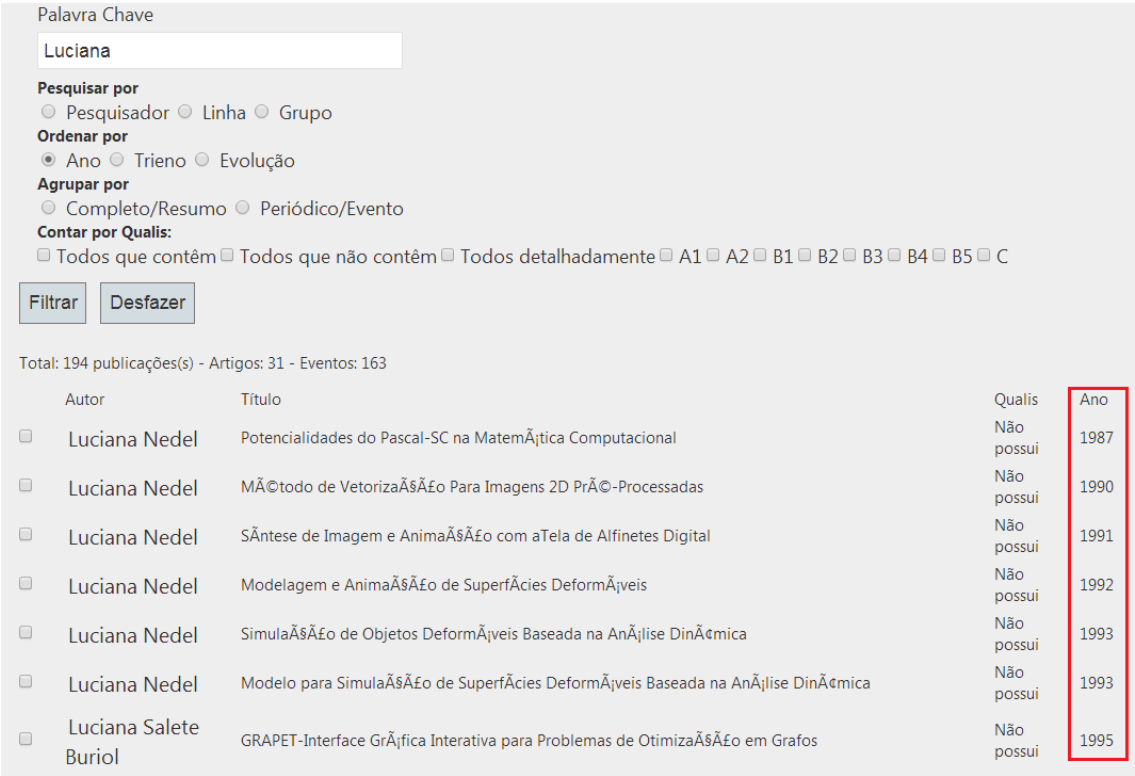


Figura 3.8: Pesquisa ordenada por ano.

Fonte: Imagem criada pelo autor.

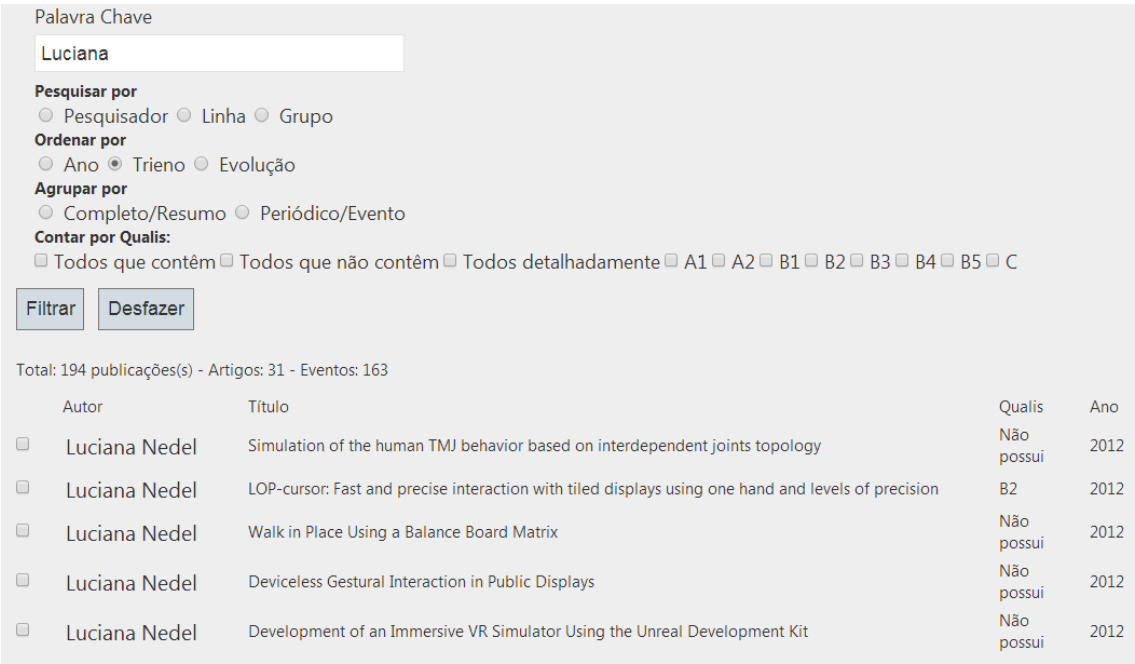


Figura 3.9: Pesquisa ordenada por trieno.

Fonte: Imagem criada pelo autor.

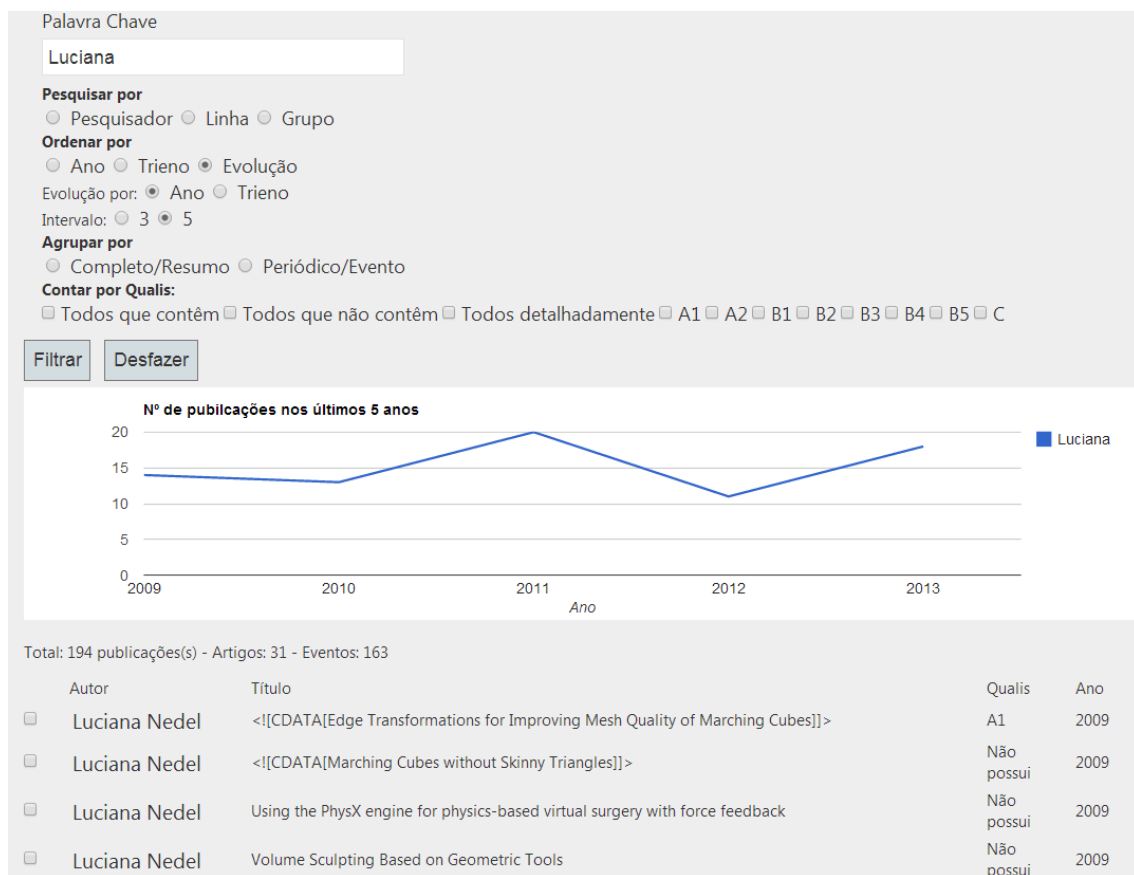


Figura 3.10: Gráfico evolutivo e pesquisa ordenanda por ano.

Fonte: Imagem criada pelo autor.

3.2.2.4Agrupar por

A opção ‘Agrupar por’ tem como objetivo agrupar as publicações, resultantes da pesquisa, em dois grupos:

- **Completo/Resumo** – Todas as publicações são classificadas pela natureza da publicação. Quando selecionada esta opção todas as publicações que possuírem a mesma natureza serão agrupadas. Existem dois tipos de publicações: artigo completo e resumo. A Figura 3.11 ilustra as publicações agrupadas pelo seu tipo.
- **Periódico/Evento** – As puvlicações também pode ser classificadas se a publivação é um artigo ou não. Sendo assim, a opção Periódico/Evento tem como objetivo agrupar as publicações por aquelas que são periódico (artigo) e as que não são. A Figura 3.12 ilustra o resultado após a pesquisa ser agrupada pela opção Periódico/Evento.

<input type="checkbox"/>	Luciana Salete Buriol	Trust-based Grouping for Cloud Datacenters: improving security in shared infrastructures	Não possui	Completo
<input type="checkbox"/>	Luciana Salete Buriol	A knowledge-based genetic algorithm to predict three-dimensional structures of polypeptides	Não possui	Completo
<input type="checkbox"/>	Luciana Salete Buriol	Finding Optimal Solutions to Sokoban Using Instance Dependent Pattern Databases	Não possui	Completo
<input type="checkbox"/>	Luciana Salete Buriol	Characterizing the Impact of Network Substrate Topologies on Virtual Network Embedding	Não possui	Completo
<input type="checkbox"/>	Luciana Salete Buriol	A biased random-key genetic algorithm for a network pricing problem	Não possui	Completo
<input type="checkbox"/>	Luciana Nedel	Método de Vetorização Para Imagens 2D Pré-Processadas	Não possui	Resumo
<input type="checkbox"/>	Luciana Nedel	Simulação de Objetos Deformáveis Baseada na Análise Dinâmica	Não possui	Resumo
<input type="checkbox"/>	Luciana Nedel	A Multimodal Visualization Framework for Medical Data	Não possui	Resumo
<input type="checkbox"/>	Luciana Nedel	Usability issues in information visualization applications	Não possui	Resumo
<input type="checkbox"/>	Luciana Nedel	On Evaluating Information Visualization Techniques	Não possui	Resumo
<input type="checkbox"/>	Luciana Nedel	Planejamento de Hepatectomias a partir de Imagens Tridimensionais (3D) do Fígado do Suíno	Não possui	Resumo

Figura 3.11: Pesquisa agrupada pelo tipo completo/resumo.

Fonte: Imagem criada pelo autor.

<input type="checkbox"/>	Luciana Salete Buriol	A knowledge-based genetic algorithm to predict three-dimensional structures of polypeptides	Não possui	Evento
<input type="checkbox"/>	Luciana Salete Buriol	Finding Optimal Solutions to Sokoban Using Instance Dependent Pattern Databases	Não possui	Evento
<input type="checkbox"/>	Luciana Salete Buriol	Characterizing the Impact of Network Substrate Topologies on Virtual Network Embedding	Não possui	Evento
<input type="checkbox"/>	Luciana Salete Buriol	A biased random-key genetic algorithm for a network pricing problem	Não possui	Evento
<input type="checkbox"/>	Luciana Nedel	Simulação de Objetos Deformáveis Baseada na Análise Dinâmica	Não possui	Artigo
<input type="checkbox"/>	Luciana Nedel	Anatomical modeling of deformable human bodies	B1	Artigo
<input type="checkbox"/>	Luciana Nedel	Automato finito não-determinístico com pesos aplicado a animações baseadas em automatos finitos	Não possui	Artigo
<input type="checkbox"/>	Luciana Nedel	Beyond user experimentation: notational-based systematic evaluation of interaction techniques in virtual reality environments	B2	Artigo
<input type="checkbox"/>	Luciana Nedel	A Proposal of a multi-level architecture for animation of synthetic actors with cognitive reasoning	C	Artigo
<input type="checkbox"/>	Luciana Nedel	Personal Computer Software Evaluation in Interactive Generation of Pig Liver Three-Dimensional Anatomical Images	B2	Artigo
<input type="checkbox"/>	Luciana Nedel	Towards an integrated system for planning and assisting maxillofacial orthognathic surgery	A2	Artigo

Figura 3.12: Pesquisa agrupada pelo tipo periódico/evento.

Fonte: Imagem criada pelo autor.

3.2.2.5 Contar por Qualis

Todas as publicações podem ser classificadas pela qualidade de produção intelectual recebendo um Qualis. Visto que, esta é uma classificação de grande importância e como existem vários níveis ao selecionar a opção ‘Contar por Qualis’ é gerado um gráfico com a soma dos Qualis desejados. As opções para gerar o gráfico de Qualis são:

- **Todos que contém** – Com o intuito de obter uma visão geral que possibilita-se fazer melhores análises, a opção ‘Todos que contém’ gera o somatório de todos as publicações que contém qualquer classificação do Qualis.
- **Todos que não contém** – Do mesmo modo que uma visão geral das publicações que contém Qualis a opção ‘Todos que não contém’ gera o

somatório de todas as publicações que não contêm qualquer classificação do Qualis.

- **Qualis separado** - Esta opção tem como objetivo proporcionar ao usuário a análise do somatório de cada Qualis em separado
- **Todos detalhadamente** – Esta opção tem como objetivo apenas poupar o usuário de selecionar todas as opções de qualis. O gráfico irá mostrar o somatório de todas as classificações de Qualis (A1, A2, B1, B2, B3, B4, B5 e C).

Para fazer a verificação se uma publicação possui um Qualis, ou ainda, qual a classificação deste Qualis, a aplicação, com o título da publicação, verifica no banco de dados se para aquela publicação há alguma classificação do Qualis. A manutenção desta tabela deve ser feita via banco de dados, não há uma tela na aplicação que faça esta manutenção. A Figura 3.13 ilustra a contagem dos Qualis de uma pesquisa.

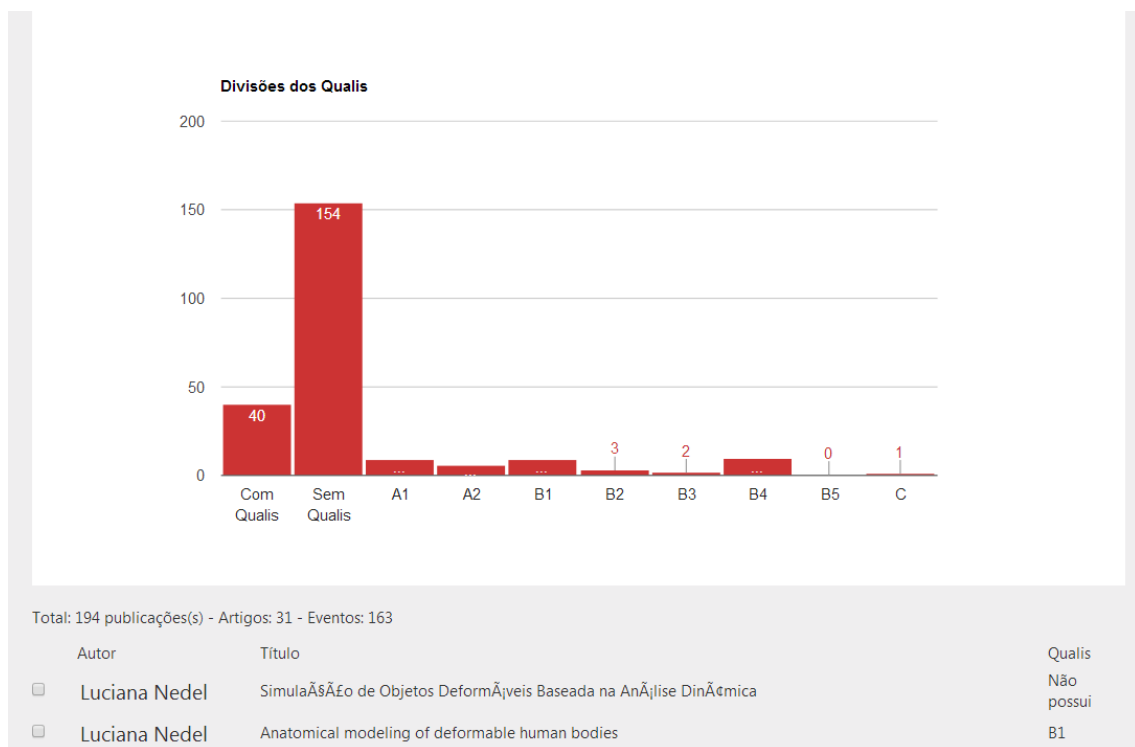


Figura 3.13: Gráfico de somatório dos Qualis.

Fonte: Imagem criada pelo autor.

3.2.3 Telas de Pesquisadores

A seção de pesquisadores tem como finalidade possibilitar a inserção e edição de pesquisadores no sistema. Ela é dividida em duas telas:

- **Listagem** – Esta tela lista todos os pesquisadores cadastrados no sistema. Ao lado do nome de cada pesquisador há um botão para redirecionar para a tela de edição de pesquisador. Ainda há um botão para adicionar um novo pesquisador ao sistema. E por último há um botão ‘Ver publicações’ o qual

redireciona para a tela de pesquisa e exibe todas as publicações referentes ao pesquisador desejado. A Figura 3.14 mostra a listagem de pesquisadores cadastrados.

Novo Pesquisador			
Nome	Trabalhando	Ações	
Luciana Nedel	Sim	Editar	Ver publicações
Renato Perez Ribas	Sim	Editar	Ver publicações
Valter Roesler	Sim	Editar	Ver publicações
Luís da Cunha Lamb	Sim	Editar	Ver publicações
Ingrid Oliveira de Nunes	Sim	Editar	Ver publicações
Érika Fernandes Cota	Sim	Editar	Ver publicações
Cláudio Rosito Jung	Sim	Editar	Ver publicações
Karin Becker	Sim	Editar	Ver publicações
Daltro José Nunes	Sim	Editar	Ver publicações
Jacob Scharcanski	Sim	Editar	Ver publicações
Nicolas Maillard	Sim	Editar	Ver publicações
Marcelo Soares Pimenta	Sim	Editar	Ver publicações

Figura 3.14: Listagem de pesquisadores do sistema.

Fonte: Imagem criada pelo autor.

- **Inserção/Edição** – As telas de inserção e edição de pesquisadores são muito semelhantes em relação ao formato que elas são exibidas para o usuário. A diferença entre elas é devido ao fato de a tela de edição vir com os campos preenchidos com os dados do pesquisador que deverá ser editado. Além dos campos preenchidos a aplicação ainda informa a data da última atualização dos dados do pesquisador que está sendo editado. Neste formulário de inserção e edição há os campos de nome do pesquisador, um campo para informar se o pesquisador ainda está ativo junto ao campo docente do Instituto de Informática e o campo para fazer upload do currículo Lattes do professor no formato XML. Após a inserção ou edição do pesquisador o usuário é redirecionado para a tela de listagem de pesquisadores. A Figura 3.15 ilustra a diferença entre as duas telas.

Nome:

Ativo: ☒

Curriculo Lattes: Nenhum...onado

INSERÇÃO

Nome:

Atualizado em: 30/04/2014 00:00:00

Ativo: ☒

Curriculo Lattes: Nenhum...onado

EDIÇÃO

Figura 3.15: Diferenças entre telas de inserção e edição de pesquisadores.

Fonte: Imagem criada pelo autor.

3.2.4 Telas de Grupos

A seção de Grupos tem como finalidade possibilitar a inserção e edição de grupos de pesquisa no sistema. Ela é dividida em duas telas:

- **Listagem** – Esta tela lista todos os grupos de pesquisa que estão cadastrados no sistema. Ao lado do nome de cada grupo de pesquisa há um botão para redirecionar para a tela de edição do grupo. Há também um botão para adicionar um novo grupo de pesquisa ao sistema. A Figura 3.16 mostra a listagem de pesquisadores cadastrados.

<input type="button" value="Novo Grupo"/>	
Bioinformática	Editar
Concepção de Circuitos Integrados	Editar
Computação Gráfica, Processamento de Imagens e Interação	Editar
Ferramentas Computacionais para Projeto de Circuitos e Sistemas Integrados	Editar
Inteligência Artificial	Editar

Figura 3.16: Listagem de grupos de pesquisa.

Fonte: Imagem criada pelo autor.

- **Inserção/Edição** - As telas de inserção e edição de grupos de pesquisa são exibidas da mesma forma ao usuário. A diferença entre elas é devido ao fato de a tela de edição vir com os campos preenchidos com os dados do grupo de pesquisa que deverá ser editado. Esta tela de inserção e edição contém os campos de nome do grupo de pesquisa, a uma lista dos pesquisadores, ativos

no sistema, onde é possível relacionar todos aqueles que fazem parte do grupo de pesquisa.. A Figura 3.17 ilustra a tela de inserção e edição de grupos de pesquisa.

Nome do grupo:

<input checked="" type="checkbox"/> Álvaro Freitas Moreira	<input checked="" type="checkbox"/> Carlos Eduardo Pereira	<input checked="" type="checkbox"/> Ingrid Oliveira de Nunes
<input type="checkbox"/> Alberto Egon Schaeffer Filho	<input type="checkbox"/> Aline Villavicencio	<input type="checkbox"/> Altamiro Amadeu Susin
<input type="checkbox"/> Anderson Maciel	<input type="checkbox"/> Andre Inacio Reis	<input type="checkbox"/> Antonio Carlos Schneider Beck Filho
<input type="checkbox"/> Carla Maria Dal Sasso Freitas	<input type="checkbox"/> Carlos Alberto Heuser	<input type="checkbox"/> Claudio Fernando Resin Geyer
<input type="checkbox"/> Daltro José Nunes	<input type="checkbox"/> Dante Augusto Couto Barone	<input type="checkbox"/> Edson Prestes e Silva Júnior
<input type="checkbox"/> Fernanda Gusmão de Lima Kastensmidt	<input type="checkbox"/> Flávio Rech Wagner	<input type="checkbox"/> Jacob Scharcanski
<input type="checkbox"/> Jose Palazzo Moreira de Oliveira	<input type="checkbox"/> Juergen Rochol	<input type="checkbox"/> Karin Becker
<input type="checkbox"/> Leila Ribeiro	<input type="checkbox"/> Lisandro Zambenedetti Granville	<input type="checkbox"/> Lucas Mello Schnorr
<input type="checkbox"/> Luciana Salete Buriol	<input type="checkbox"/> Luciano Paschoal Gaspar	<input type="checkbox"/> Lucinéia Heloisa Thom
<input type="checkbox"/> Luís da Cunha Lamb	<input type="checkbox"/> Manuel Menezes de Oliveira Neto	<input type="checkbox"/> Mara Abel
<input type="checkbox"/> Marcelo Walter	<input type="checkbox"/> Márcio Dorn	<input type="checkbox"/> Marcus Rolf Peter Ritt
<input type="checkbox"/> Nicolas Maillard	<input type="checkbox"/> Paolo Rech	<input type="checkbox"/> Paulo Martins Engel
<input type="checkbox"/> Renata Galante	<input type="checkbox"/> Renato Perez Ribas	<input type="checkbox"/> Ricardo Augusto da Luz Reis
<input type="checkbox"/> Sergio Bampi	<input type="checkbox"/> Valter Roesler	<input type="checkbox"/> Viviane Pereira Moreira

Figura 3.17: Tela de inserção e edição de grupos de pesquisa.

Fonte: Imagem criada pelo autor.

3.2.5 Telas Áreas de Pesquisa

A seção de áreas tem como objetivo possibilitar a inserção e edição de áreas de pesquisa no sistema. Ela é dividida em duas telas:

- **Listagem** – Esta tela lista todos as áreas de pesquisa que estão cadastrados no sistema. Ao lado do nome de cada área de pesquisa há um botão para redirecionar para a tela de edição da área. Há também um botão para adicionar uma nova área de pesquisa ao sistema. A Figura 3.18 mostra a listagem de pesquisadores cadastrados.

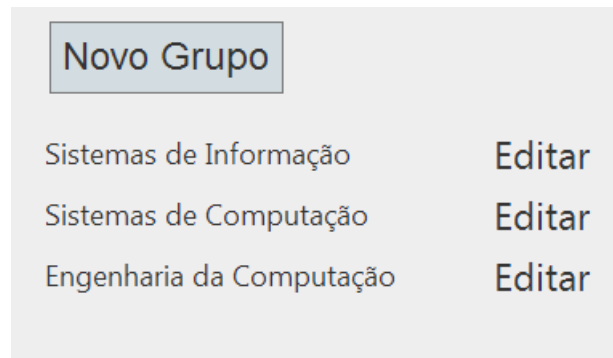


Figura 3.18: Tela de listagem de área de pesquisa.

Foten: Imagem criada pelo autor.

- **Inserção/Edição** - As telas de inserção e edição de áreas de pesquisa são exibidas da mesma forma ao usuário. A diferença entre elas é devido ao fato de a tela de edição vir com o campo nome, que é o único campo da tela, preenchido com o nome da área de pesquisa que deverá ser editado. A Figura 3.19 ilustra a tela de inserção e edição de áreas de pesquisa.

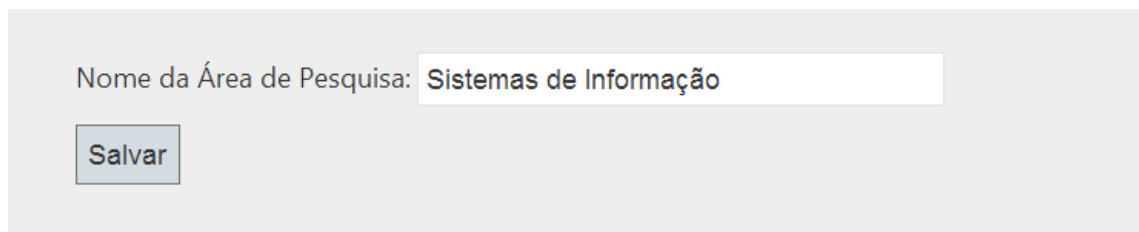


Figura 3.19: Tela de inserção e edição de área de pesquisa.

Foten: Imagem criada pelo autor.

3.2.6 Telas de Linhas de Pesquisa

A seção de linhas de pesquisa tem como finalidade possibilitar a inserção e edição de linhas de pesquisa no sistema. Ela é dividida em duas telas:

- **Listagem** – Esta tela lista todos as linhas de pesquisa que estão cadastrados no sistema. As linhas de pesquisa estão separadas por área de pesquisa. Ao lado do nome de cada linha de pesquisa há um botão para redirecionar para a tela de edição da linha de pesquisa. Há também um botão para redirecionar para a tela de inserção uma linha de pesquisa. A Figura 3.20 mostra a listagem de pesquisadores cadastrados.

Novo Grupo	
Sistemas de Informação	
Engenharia de Software	Editar
Verificação e Teste de Software	Editar
Modelagem Conceitual e Banco de Dados	Editar
Sistemas de Computação	
Computação Gráfica, Processamento de Imagens e Interação	Editar
Processamento Paralelo e Distribuído	Editar
Redes de Computadores	Editar
Engenharia da Computação	
Arquitetura e Projeto de Sistemas Computacionais	Editar
Microeletrônica	Editar
Sistemas Embarcados	Editar
Teste e Confiabilidade de Sistemas Integrados de Hardware e Software	Editar

Figura 3.20: Tela de listagem das linhas de pesquisa.

Fonte: Imagem criada pelo autor.

- **Inserção/Edição** - As telas de inserção e edição de linhas de pesquisa são exibidas da mesma forma ao usuário. A diferença entre elas é devido ao fato de a tela de edição vir com os campos preenchidos com os dados da linha de pesquisa que deverá ser editada. Esta tela de inserção e edição contém os campos de nome da linha de pesquisa, um campo onde é possível relacionar a linha de pesquisa a uma área de pesquisa. Por fim há também uma lista dos pesquisadores, ativos no sistema, onde é possível relacionar a linha de pesquisa.. A Figura 3.21 ilustra a tela de inserção e edição de linhas de pesquisa.

Nome da Linha de Pesquisa: **Engenharia de Software**

Selecione o Área: **Sistemas de Informação**

Selecione os pesquisadores que fazem parte da linha de pesquisa:

<input checked="" type="checkbox"/> Álvaro Freitas Moreira	<input checked="" type="checkbox"/> Altamiro Amadeu Susin	<input checked="" type="checkbox"/> Luciana Nedel	<input checked="" type="checkbox"/> Renato Perez Ribas
<input type="checkbox"/> Alberto Egon Schaeffer Filho	<input type="checkbox"/> Aline Villavicencio	<input type="checkbox"/> Ana Lucia Cetertich Bazzan	<input type="checkbox"/> Anderson Maciel
<input type="checkbox"/> Andre Inacio Reis	<input type="checkbox"/> Antonio Carlos Schneider Beck Filho	<input type="checkbox"/> Antonio Marinho Pilla Barcellos	<input type="checkbox"/> Carla Maria Dal Sasso Freitas
<input type="checkbox"/> Carlos Alberto Heuser	<input type="checkbox"/> Carlos Eduardo Pereira	<input type="checkbox"/> Claudio Fernando Resin Geyer	<input type="checkbox"/> Cláudio Rosito Jung
<input type="checkbox"/> Daltro José Nunes	<input type="checkbox"/> Dante Augusto Couto Barone	<input type="checkbox"/> Edson Prestes e Silva Júnior	<input type="checkbox"/> Érika Fernandes Cota
<input type="checkbox"/> Fernanda Gusmão de Lima Kastensmidt	<input type="checkbox"/> Flávio Rech Wagner	<input type="checkbox"/> Ingrid Oliveira de Nunes	<input type="checkbox"/> Jacob Scharcanski
<input type="checkbox"/> Joao Luiz Dihl Comba	<input type="checkbox"/> Jose Palazzo Moreira de Oliveira	<input type="checkbox"/> Juergen Rochol	<input type="checkbox"/> Karin Becker
<input type="checkbox"/> Leandro Krug Wives	<input type="checkbox"/> Leila Ribeiro	<input type="checkbox"/> Lisandro Zambenedetti Granville	<input type="checkbox"/> Lucas Mello Schnorr
<input type="checkbox"/> Luciana Salete Buriol	<input type="checkbox"/> Luciano Paschoal Gaspary	<input type="checkbox"/> Lucinéia Heloisa Thom	<input type="checkbox"/> Luigi Carro
<input type="checkbox"/> Luís da Cunha Lamb	<input type="checkbox"/> Manuel Menezes de Oliveira Neto	<input type="checkbox"/> Mara Abel	<input type="checkbox"/> Marcelo Soares Pimenta
<input type="checkbox"/> Marcelo Walter	<input type="checkbox"/> Márcio Dorn	<input type="checkbox"/> Marcus Rolf Peter Ritt	<input type="checkbox"/> Mariana Luderitz Kolberg
<input type="checkbox"/> Nicolas Maillard	<input type="checkbox"/> Paolo Rech	<input type="checkbox"/> Paulo Martins Engel	<input type="checkbox"/> Philippe Olivier Alexandre Navaux
<input type="checkbox"/> Renata Galante	<input type="checkbox"/> Ricardo Augusto da Luz Reis	<input type="checkbox"/> Rosa Maria Vicari	<input type="checkbox"/> Sergio Bampi
<input type="checkbox"/> Valter Roesler	<input type="checkbox"/> Viviane Pereira Moreira		

Figura 3.21: Tela de inserção e edição de linha de pesquisa.

Fonte: Imagem criada pelo autor.

4 CONCLUSÃO

Este trabalho uma solução para o problema de análise dos dados acadêmicos junto aos pesquisadores do Instituto de Informática. Este trabalho apresentou um novo sistema o qual possibilita a realização de análises referentes as publicações dos pesquisadores do Instituto de Informática. Esta análise pode ser realizada baseado em buscas que tem como escopo as publicações de um pesquisador específico, as publicações dos pesquisadores pertencentes a uma linha de pesquisa e as publicações de um grupo de pesquisa. A análise é, também, capaz de quantificar, detalhadamente, a quantidade de Qualis a pesquisa contém. Ela possibilita, ainda, o ordenamento dos resultados por ano e possibilita a visualização de um gráfico evolutivo, em relação a quantidade de publicações ao longo de um certo intervalo de tempo.

Além das análises das publicações, o sistema possibilita, ainda, a inserção e edição de áreas de pesquisa, linhas de pesquisa, grupos de pesquisa e pesquisadores. Este último podendo atualizar o currículo Lattes do pesquisador.

Durante o desenvolvimento foram realizadas análises referentes as necessidades dos usuários a serem atendidas com diversas reuniões com o professor orientador deste trabalho. Foram realizadas, também análises referentes as tecnologias que poderiam ser utilizadas no desenvolvimento desta aplicação. Ao decorrer do desenvolvimento com a mudança de algumas prioridades algumas tecnologias deixaram de fazer sentido e deixaram de ser usadas tais como, banco de dados para XML e banco de dados NoSQL para grafos.

Como sugestões para trabalhos futuros é possível salientar que, na tela de pesquisa, a pesquisa por palavra chave deve ser feita com a grafia, exatamente, correta do nome ou sobrenome do pesquisador. Poderia ser adicionado uma funcionalidade para que resultasse os pesquisadores cujo nome fosse diferente por uma ou duas letras, mas que fosse muito próximo do real nome. Poderia ser feita uma análise baseado em Cliques, grupos altamente coesos. E por fim, poderia ser gerado um gráfico de autoria e co-autoria mostrando as relações entre os pesquisadores.

REFERÊNCIAS

AGILE ALLIANCE, **The Twelve Principles of Agile Software**. Disponível em <<http://www.agilealliance.org/the-alliance/the-agile-manifesto/the-twelve-principles-of-agile-software/>>. Acesso entre Setembro de 2013 e Agosto de 2014.

BURBECK, Steve. **Application Programming in Smaltalk-(80TM): How to use Model-View-Controller (MVC)**. Disponível em <<http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html>>. Acesso entre Setembro de 2013 e Agosto de 2014.

CODE PROJECT. **LINQ Tutorial for Beginners**. Disponível em <<http://www.codeproject.com/Tips/590978/LINQ-Tutorial-for-Beginners>>. Acesso entre Setembro de 2013 e Agosto de 2014.

DEVELOPER NETWORK, **Overview of .NET Framework**. Disponível em <<http://msdn.microsoft.com/en-us/library/zw4w595w%28v=vs.110%29.aspx>>. Acesso entre Setembro de 2013 e Agosto de 2014.

DEVELOPER NETWORK. **LINQ: .NET Language-Integrated Query**. Disponível em <<http://msdn.microsoft.com/en-us/library/bb308959.aspx>>. Acesso entre Setembro de 2013 e Agosto de 2014.

GOOGLE DEVELOPERS, **Chart Galery**. Disponível em <<https://developers.google.com/chart/interactive/docs/gallery?hl=pt-br>>. Acesso entre Setembro de 2013 e Agosto de 2014.

MSDN MAGAZINE, **The Evolution Of Linq And Its Impact On The Design Of C#**. Dispon[ível em <<http://msdn.microsoft.com/en-us/magazine/cc163400.aspx>>. Acesso entre Setembro de 2013 e Agosto de 2014.

NUNES, Renato M. B. **Validação do Framework FPOA através da implementação de um repositório de objetos de aprendizagem no modelo UMBRELO**. 2013. Projeto de Diplomação (Bacharelado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre

PRESSMAN, R.; LOWE, D. Web Engineering e A Web Engineering Process. In: PRESSMAN, Roger S. **Web Engineering**. 1 ed. Sidney: David Lowe, 2009. P. 12-45.

SOMMERVILLE, Ian. **Software Engineering**. 9 ed. Boston: Addison Wesley, 2010.