

**SIGLASEARCH - ARCABOUÇO DE CONSULTA
DE DADOS FLEXIBILIZADA PARA
WORKFLOWS DE MÚLTIPLOS LABORATÓRIOS
DE PESQUISA**

CÁSSIA DO CARMO VIEIRA

**SIGLASEARCH - ARCABOUÇO DE CONSULTA
DE DADOS FLEXIBILIZADA PARA
WORKFLOWS DE MÚLTIPLOS LABORATÓRIOS
DE PESQUISA**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR: SÉRGIO VALE AGUIAR CAMPOS
COORIENTADORA: ALESSANDRA CONCEIÇÃO FARIA AGUIAR
CAMPOS

Belo Horizonte

Setembro de 2015

© 2015, Cássia do Carmo Vieira.
Todos os direitos reservados.

V658s Vieira, Cássia do Carmo
SIGLASEARCH - Arcabouço de consulta de dados
flexibilizada para workflows de múltiplos laboratórios
de pesquisa / Cássia do Carmo Vieira — Belo
Horizonte, 2015.
xxvii, 133 f. : il. ; 29cm.

Dissertação (Mestrado) — Universidade Federal de
Minas Gerais
Departamento de Ciência da Computação.

Orientador: Sérgio Vale Aguiar Campos
Coorientadora: Alessandra Conceição Faria Aguiar
Campos.

1. Computação - Teses. 2. Banco de dados - Teses.
3. Banco de dados Relacionais - Teses.
4. Bioinformática - Teses. Fluxo de trabalho - Teses.
I. Orientador. II. Coorientadora. III. Título.

519.6*72(043)




UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO


Siglasearch - Arcabouço de consulta de dados flexibilizada para workflows de múltiplos laboratórios de pesquisa


CÁSSIA DO CARMO VIEIRA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:


PROF. SÉRGIO VALE AGUIAR CAMPOS - Orientador
Departamento de Ciência da Computação - UFMG


DRA. ALESSANDRA CONCEIÇÃO FÁRIA AGUIAR CAMPOS
Instituto de Ciências Biológicas - UFMG


DR. FRANCISCO PEREIRA LOBO
Embrapa Informática Agropecuária


PROF. RAQUEL OLIVEIRA PRATES
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 22 de setembro de 2015.

Aos meus pais, José Vieira e Aparecida Maia, e à minha irmã Kessia Vieira.

Agradecimentos

Antes de tudo, quero agradecer a Deus, pelos desafios que recebi e pela força nos momentos mais difíceis.

Aos meus pais, que me guiaram, que souberam dar amor e atenção na medida certa e me treinaram para a vida. A todos os meus familiares e amigos, que vibraram comigo nas conquistas, me apoiaram nos momentos difíceis e fizeram a caminhada ser mais amena.

Aos colegas que conheci ao fazer as disciplinas do Mestrado, pelos desafios que passamos juntos, pela divisão de tarefas, troca de conhecimento e colaboração em cada disciplina.

Aos professores que souberam estimular nossos talentos e se esforçaram para transmitir o melhor do conhecimento.

Aos colegas de trabalho, que muito colaboraram nas minhas experiências profissionais e que me motivaram e me entusiasmaram para concluir esse Mestrado. Quero agradecer por todos os analistas, por abrir a minha mente e por compartilharem seus conhecimentos com satisfação a cada oportunidade.

Aos meus amigos do Laboratório de Universalização de Acesso (LUAR) por compartilharem seus conhecimentos, pelas dicas em cada fase desse trabalho, por me mostrarem na prática o que é a pesquisa científica, por contagiarem com sua paixão por criar soluções melhores sobre os problemas que existem hoje pendentes para a Bioinformática. Além dos momentos de descontração que tivemos e que fizeram o trabalho ser mais leve e mais divertido. Amigos do LUAR, jamais vou esquecer vocês e os momentos que crescemos juntos.

E, por último, mas não menos importante, agradeço ao Sérgio Campos e à Alessandra Campos, pessoas que tive a sorte de ter como orientadores, que me deram o direcionamento e a inspiração para concluir esse projeto. Obrigada pela oportunidade de aprender tanto com o profissionalismo e a paixão de vocês pelo trabalho de pesquisa científica e a criação de soluções para os problemas reais que poderão contribuir para a melhoria da nossa vida, enquanto sociedade.

“Seja a mudança que você quer ver no mundo.”
(Mahatma Gandhi)

Resumo

A pesquisa biomédica sofreu um grande crescimento de dados gerado pela ampliação da legislação sobre os processos de laboratórios, avanços na instrumentação automática, melhoria dos processos de controle de qualidade e aumento da demanda por análises estatísticas. Para gerenciar esse volume de informação, alguns laboratórios de pesquisa utilizam Sistemas de Gerenciamento de Informações de Laboratório (*Laboratory Information Management System* - LIMS), que propõem gerenciamento de gravação e acesso aos dados de forma segura e satisfatória. Devido aos laboratórios de pesquisa possuírem características e processos de trabalho bastante peculiares e à grande variedade de laboratórios, os LIMS são criados com a estrutura de dados customizada para cada um deles. Essa necessidade de adaptação do *software* provoca aumento do tempo de desenvolvimento e do custo final da aplicação, o que torna a implantação de um LIMS inviável para pequenos laboratórios ou para laboratórios de pesquisa. Muitas dessas pesquisas são aplicadas em problemas da bioinformática. Para flexibilizar a disponibilização do LIMS para diferentes contextos, foi criado o *Sistema Integrado de Gerência de Laboratórios*(SIGLa), que permite ao laboratório adaptar o fluxo de dados do sistema de acordo com as suas próprias necessidades de trabalho sem que o código fonte do sistema seja alterado. Para alterar os processos de trabalho de um laboratório informatizado, basta que o *workflow* (fluxo de trabalho) seja editado através de um editor de *workflows* como o *Together Workflow Editor*(TWE). O SIGLa atua como gerenciador de *workflows*, fazendo o controle de transação, consistência e gravação dos dados. O SIGLa já é utilizado na *Empresa Brasileira de Pesquisa Agropecuária*(Embrapa) e no *Instituto Nacional de Metrologia, Qualidade e Tecnologia*(Inmetro). Embora a gravação de dados seja flexível e não requeira customizações por mudanças dos fluxos de trabalho, a consulta aos dados não é satisfatória. A consulta existente exhibe os dados gravados, mas não exhibe tais dados de forma relacionada. Tal problema não tem solução trivial, pois a modelagem do sistema foi feita em Banco de Dados Relacional em função dos conceitos do gerenciador de *workflow*, como atividades, atributos, registros e instâncias. O antigo sistema de consulta do

SIGLa fornece resultados com pouca significância para análise das informações. As tabelas estão relacionadas pelos seus metadados. Portanto, não é possível, por exemplo, extrair os relacionamentos entre as atividades do *workflow* e relacionamentos entre os atributos dessas atividades. As ferramentas existentes hoje para a consulta automática de Bancos de Dados Relacional não atendem ao problema de consulta no SIGLa, pois a extração das restrições do esquema relacional não é suficiente para a geração dos resultados que o usuário necessita em suas análises. Esse projeto propõe a modelagem e a implementação do *SiglaSearch*, uma engenharia de consulta para dados de *workflow* com interface customizada para os diferentes usuários do sistema. A consulta deve ser flexibilizada para múltiplos fluxos de trabalho e deve exibir as informações relacionadas entre si, sem a necessidade de conversão do modelo de dados orientado a *workflow* para um modelo de banco de dados convencional. O novo sistema de consulta desenvolvido para o SIGLa traz como melhoria a pesquisa em dados de *workflow*, porque os sistemas de consulta existentes para *workflow* só oferecem busca por palavras-chave. O *SiglaSearch* propõe a pesquisa através de subconjunto de informações de *workflow*, com utilização dos dados estruturados para pesquisas mais avançadas. O *SiglaSearch* permite a busca de subconjuntos de dados, filtragem e ordenação de qualquer uma das informações do *workflow* e sumarização de resultados. Os relacionamentos são extraídos de forma automática sem especificações adicionais, com exibição das informações em formato tabular, para que os usuários possam focar em subconjuntos de informações que facilitem a avaliação das informações para os mais diversos fins. O *SiglaSearch* também fornece uma interface de consulta para os usuários de forma que eles possam buscar seus resultados em poucos passos de forma intuitiva e possam salvar essas consultas sob a forma de relatórios que podem ser executados por outros usuários que trabalhem em atividades mais isoladas do laboratório de pesquisa.

Palavras-chave: SIGLa, LIMS, Fluxo de Trabalho, Workflow, Engenharia de Consulta, Banco de Dados Relacional, Bioinformática.

Abstract

Biomedical research has undergone strong growth of data generated by the expansion of the law governing laboratory procedures, advances in automated instrumentation, improved quality control processes and increased demand for statistical analysis. To control this volume of information, some research labs use Laboratory Information Management Systems (LIMS), which propose recording management and access to data safely and satisfactorily. Research labs have quite peculiar features and work processes. Therefore, the LIMS are created with custom data structure for each of them. This need for software adaptation increases development time and cost of the final application, which makes the implementation of a LIMS impractical for small laboratories or research laboratories. Many of these researches are applied to bioinformatics problems. In order to increase availability of LIMS for the laboratory, the Integrated Laboratory Management System (*Sistema Integrado de Gerência de Laboratórios - SIGLa*) was created, which enables the laboratory to adapt the data flow of the system according to their business needs without changes in the source code of the system. To change the work processes of a lab, just the workflow is edited through a workflow editor such as the Together Workflow Editor (TWE). SIGLa acts as a workflow manager, making the transaction control, data consistency and data recording. SIGLa is already used in the Brazilian Agricultural Research Corporation (*Empresa Brasileira de Pesquisa Agropecuária - Embrapa*) and the National Institute of Metrology, Quality and Technology (*Instituto Nacional de Metrologia, Qualidade e Tecnologia - Inmetro*). Although data recording is flexible and does not require customizations for change of workflows, the query data is not satisfactory. The existing query displays the recorded data, but displays no such related form of data. This problem has a nontrivial solution since the system modeling has been made in Relational Database according to the concepts of workflow manager, such as activities, attributes, records and instances. This implies low value of the results of the old query system for information analysis. The tables are related by their metadata, so the user can not extract the relationships between the activities of the workflow, relationships between the attributes of these activities

and relationships between records that make some types of workflow attributes and other attributes. Tools existing today for automatic query Relational Databases, do not solve the query problem in SIGLa, because the extraction of the constraints of relational schema is not enough to generate the results that the user needs in his analysis. This project proposes the modeling and implementation of SiglaSearch, a system to query workflow data with customized user interface to different users, more flexible for different workflows, to view the information related with each other without the need for conversion of data model-driven workflow to a model generated in terms of the relationships between the data. The new query system developed for SIGLa brings as improvement research in workflow data, because the existing reference systems for such data only offer keyword search while SiglaSearch proposes to search through a subset of workflow information. SiglaSearch allows the search of data subsets, filtering and sorting of any workflow information and summarization of results. Relationships are extracted automatically without additional specifications, with display of information in tabular format, so users can focus on subsets of information to facilitate the evaluation of information for various purposes. SiglaSearch also provides a query interface for users so that they can get their results in a few steps intuitively and can save these consultations in the form of reports that can be run by other users working in more specific activities of the laboratory search.

Keywords: SIGLa, LIMS, Workflow, Search Engine, Relational Database, Bioinformatics.

Lista de Figuras

1.1	Exemplo ilustrativo de pesquisa para identificação de sintomas relacionados a uma doença estudada	5
1.2	Resultados gerados pelo antigo módulo de consulta do sistema SIGLa	6
1.3	Funcionamento do antigo sistema de consulta do SIGLa	7
3.1	Tela de entrada de dados de instância de <i>workflow</i> do SIGLa	34
3.2	Tela principal do editor de consulta do sistema SiglaSearch com as principais instruções	36
3.3	Tela principal do editor de consultas do sistema SiglaSearch com exibição dos resultados	37
3.4	Tela para abrir consulta salva no sistema SiglaSearch	37
3.5	Tela para salvar consulta no sistema SiglaSearch	38
3.6	Fluxo de uso do assistente de consulta	39
3.7	Telas de execução do assistente de consulta do sistema SiglaSearch para os primeiros passos da edição de uma consulta	40
3.8	Fluxo de passos para adição de novo campo à consulta do sistema SiglaSearch	42
3.9	Passos para seleção de campo de uma consulta do sistema SiglaSearch	43
3.10	Escolha de outra coluna do registro do sistema SiglaSearch	44
3.11	Passos para adição de campo de sumarização de dados do sistema SiglaSearch	45
3.12	Para modificar a ordem das colunas de resultado do sistema SiglaSearch, basta arrastar as colunas com o <i>mouse</i>	46
3.13	Escolha da opção de ordenação do campo como crescente ou decrescente no sistema SiglaSearch	46
3.14	Fluxo de passos para adição de filtro à consulta do sistema SiglaSearch	48
3.15	Telas dos passos específicos para adição de filtro no sistema SiglaSearch	49
3.16	Tela de visualização dos detalhes de um campo selecionado	50
3.17	Tela de visualização dos detalhes de um filtro entre dois campos	50
3.18	Tela de visualização dos detalhes de um filtro entre um campo e uma constante	51

3.19	Visualização de critério de ordenação no sistema SiglaSearch	51
3.20	Menu de contexto para colunas selecionadas no sistema SiglaSearch	52
3.21	Menu de relatórios disponíveis do sistema SiglaSearch	53
3.22	Tela inicial de Relatório do sistema SiglaSearch	53
3.23	Tela de relatório com exibição dos resultados do sistema SiglaSearch	54
3.24	Tela de administração das consultas e relatórios salvos.	55
4.1	Esquema das tabelas básicas para gravação das informações dos laboratórios do sistema SIGLa.	61
4.2	Tabelas externas para gravação de dados de atividades do <i>workflow</i> de Gestão de Projetos.	63
4.3	Registro do <i>workflow</i> de Gestão de Projetos gravado em tabela externa.	63
4.4	Representação de uma instância do <i>workflow</i> Gestão de Projetos, com atividades rotuladas com seus identificadores.	66
4.5	Tabelas usadas para a extração dos relacionamentos entre as atividades.	67
4.6	Extração dos quatro primeiros nós dos caminhos do nó raiz aos nós folha.	69
4.7	Extração dos oito primeiros nós dos caminhos do nó raiz aos nós folha.	69
4.8	Geração dos registros de correspondência entre grupos de atividades e atividades.	72
4.9	Plano de execução para a busca das metas.	73
4.10	Plano de execução para a busca das atividades.	74
4.11	Plano de execução para a busca dos projetos.	74
4.12	Plano de execução da consulta de metas e atividades de um projeto.	75
4.13	Modelo para persistência de dados das consultas do banco de dados do sistema SiglaSearch	79
5.1	<i>Workflow</i> de Gestão de Projetos	82
5.2	Consulta de andamento do Projeto	83
5.3	Consulta de prazos críticos do Projeto	84
5.4	<i>Workflow</i> de pesquisa de Meningite	84
5.5	Consulta de casos de meningite por escolaridade e por estado da federação	85
5.6	Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos gerados	86
5.7	Análise da relação de HIV com as sequelas	87
5.8	Relatório da relação entre tabagismo prévio com o tempo de evolução da doença e com a quantidade de casos	89
5.9	Análise de vômitos em pacientes	89

5.10	Relação de diagnósticos por gênero	90
A.1	Tela de entrada de dados de instância de <i>workflow</i> do SIGLa	104
A.2	Tela principal do editor de consulta do sistema SiglaSearch com as principais instruções	105
A.3	Tela principal do editor de consultas do sistema SiglaSearch com exibição dos resultados	106
A.4	Tela para abrir consulta salva no sistema SiglaSearch	106
A.5	Tela para salvar consulta no sistema SiglaSearch	107
A.6	Fluxo de uso do assistente de consulta	108
A.7	Telas de execução do assistente de consulta do sistema SiglaSearch para os primeiros passos da edição de uma consulta	109
A.8	Fluxo de passos para adição de novo campo à consulta do sistema SiglaSearch	111
A.9	Passos para seleção de campo de uma consulta do sistema SiglaSearch . . .	112
A.10	Escolha de outra coluna do registro do sistema SiglaSearch	113
A.11	Passos para adição de campo de sumarização de dados do sistema SiglaSearch	114
A.12	Para modificar a ordem das colunas de resultado do sistema SiglaSearch, basta arrastar as colunas com o <i>mouse</i>	115
A.13	Escolha da opção de ordenação do campo como crescente ou decrescente no sistema SiglaSearch	115
A.14	Fluxo de passos para adição de filtro à consulta do sistema SiglaSearch . .	117
A.15	Telas dos passos específicos para adição de filtro no sistema SiglaSearch . .	118
A.16	Tela de visualização dos detalhes de um campo selecionado	119
A.17	Tela de visualização dos detalhes de um filtro entre dois campos	119
A.18	Tela de visualização dos detalhes de um filtro entre um campo e uma constante	120
A.19	Visualização de critério de ordenação no sistema SiglaSearch	120
A.20	Menu de contexto para colunas selecionadas no sistema SiglaSearch	121
A.21	Menu de relatórios disponíveis do sistema SiglaSearch	122
A.22	Tela inicial de Relatório do sistema SiglaSearch	122
A.23	Tela de relatório com exibição dos resultados do sistema SiglaSearch	123
A.24	Tela de administração das consultas e relatórios salvos.	124

Lista de Tabelas

2.1	Limitações identificadas em alguns dos sistemas de apoio à pesquisa analisados	21
3.1	Tipos de atributos do SIGLa	41
3.2	Funções de sumarização e tipos compatíveis do sistema SiglaSearch	44
3.3	Especificação dos filtros de consulta do sistema SiglaSearch	47
4.1	Descrições das tabelas de gravação dos dados dos laboratórios no SIGLa	60
4.2	Descrições das tabelas de persistência da consulta do sistema SiglaSearch	78
A.1	Tipos de atributos do SIGLa	110
A.2	Funções de sumarização e tipos compatíveis do sistema SiglaSearch	113
A.3	Especificação dos filtros de consulta do sistema SiglaSearch	116
B.1	Consulta de andamento do Projeto	126
B.2	Consulta de prazos críticos do Projeto	127
B.3	Consulta de casos de meningite por escolaridade e por estado da federação	127
B.4	Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos (Parte I)	128
B.5	Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos (Parte II)	129
B.6	Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos (Parte III)	130
B.7	Análise da relação de HIV com as sequelas (Parte I)	131
B.8	Análise da relação de HIV com as sequelas (Parte II)	132
B.9	Relatório da relação entre tabagismo prévio com o tempo de evolução da doença e com a quantidade de casos	132
B.10	Análise de vômitos em pacientes com PCM	133
B.11	Relação do diagnóstico por gênero	133

Lista de Abreviaturas e Siglas

BPL Boas Práticas de Laboratório

BPMN Business Process Modeling Notation

CEBS Chemical Effects in Biological Systems

CSV Comma-Separated Values

DNA Ácido Desoxirribonucleico

EAV Entity-Attribute-Value

ER Entidade-Relacionamento

EST Expressed Sequence Tag

GLP Good Laboratory Practices

GMO Genetically Modified Organism

GO Gene Ontology

JSF Java Server Faces

LIMS Laboratory Information Management System

MVC Modelo, Visão e Controle

OLAP On-Line Analytical Processing

OWL Web Ontology Language

RDBMS Relational Database Management System

RDF Resource Description Framework

RNA Ribonucleic Acid

SGBD Sistema Gerenciador de Banco de Dados

SIGLa Sistema Integrado de Gerência de Laboratórios

SNP Single Nucleotide Polymorphism

SQL Structured Query Language

SUS Sistema Único de Saúde

TWE Together Workflow Editor

XML eXtensible Markup Language

XPDL XML Process Definition Language

W3C World Wide Web Consortium

WfMC Workflow Management Coalition

Sumário

Agradecimentos	ix
Resumo	xiii
Abstract	xv
Lista de Figuras	xvii
Lista de Tabelas	xxi
Lista de Abreviaturas e Siglas	xxiii
1 Introdução	1
1.1 Motivação	2
1.2 Definição do Problema	5
1.2.1 Sistema de consulta antigo do SIGLa	5
1.3 Contribuições	7
1.4 Estrutura da Dissertação	8
2 Fundamentação teórica	9
2.1 Fundamentos	9
2.1.1 Bioinformática	9
2.1.2 <i>Workflow</i>	9
2.1.3 <i>Laboratory Information Management Systems</i> (LIMS)	10
2.1.4 <i>Good Laboratory Practices</i> (GLP)	11
2.1.5 Armazenamento e recuperação de informação em Banco de Dados	11
2.1.6 Banco de Dados Relacional (<i>Relational Database Management</i> <i>System - RDBMS</i>)	14
2.1.7 Ontologia	15
2.1.8 Usabilidade de sistemas	18

2.2	Trabalhos relacionados	19
2.2.1	Sistemas de computação para auxílio às pesquisas	20
2.2.2	<i>Laboratory Information Management Systems (LIMS)</i>	22
2.2.3	<i>Framework SIGLa</i>	24
2.2.4	Ferramentas de Consulta	26
3	Interface Gráfica do Sistema de Consulta	33
3.1	Entrada de Dados no SIGLa	33
3.2	Perfis de usuário	34
3.2.1	Administrador do sistema	35
3.2.2	Usuário especialista	35
3.3	Edição de Consulta	35
3.3.1	Tela principal	36
3.3.2	Criar nova consulta	38
3.3.3	Campo de Dados da Consulta	39
3.3.4	Adição de campo selecionado	41
3.3.5	Adicionar critério de ordenação	46
3.3.6	Adicionar filtro	46
3.3.7	Alterar e excluir itens da consulta	49
3.3.8	Caminhos alternativos de funcionalidades	52
3.4	Consulta de Relatórios	52
3.5	Administração de Consultas	54
3.6	Discussão	54
4	Implementação da Engenharia de Consulta	59
4.1	Modelo de dados do SIGLa	60
4.2	Definição de relacionamento dos dados de workflow	64
4.3	Execução da Consulta	71
4.4	Modelo de dados para persistência da consulta	78
5	Estudos de Caso: Modelagem de Consultas	81
5.1	Gestão de Projetos	81
5.2	Meningite	83
5.3	Paracoccidiodomicose (PCM)	87
6	Conclusão	91
6.1	Trabalhos Futuros	92

Referências Bibliográficas	93
Anexo A Manual do usuário	103
A.1 Entrada de Dados no SIGLa	103
A.2 Edição de Consulta	104
A.2.1 Tela principal	105
A.2.2 Criar nova consulta	107
A.2.3 Campo de Dados da Consulta	108
A.2.4 Adição de campo selecionado	110
A.2.5 Adicionar critério de ordenação	115
A.2.6 Adicionar filtro	115
A.2.7 Alterar e excluir itens da consulta	118
A.2.8 Caminhos alternativos de funcionalidades	121
A.3 Consulta de Relatórios	121
A.4 Administração de Consultas	123
Anexo B Resultados dos Relatórios dos Casos de Uso	125
B.1 Gestão de Projetos	125
B.2 Meningite	125
B.3 Paracoccidiodomicose (PCM)	125

Capítulo 1

Introdução

O uso da tecnologia da informação contribuiu para um avanço científico e tecnológico sem precedentes em diversas áreas de conhecimento, tanto nas universidades quanto na indústria e na prestação de serviços. E o potencial para a geração de soluções através da tecnologia da informação não foi esgotado. A interdisciplinaridade explorada entre a tecnologia da informação e as demais áreas de conhecimento produz aumento da produtividade na geração de resultados, maior precisão nas soluções, qualidade e segurança no aproveitamento da informação gerada. A automatização, outra característica forte da computação, reduz custos de capital e de tempo, valoriza a mão-de-obra dos pesquisadores, que podem usar a maior parte do tempo na tomada de decisões, após liberados do trabalho de execução de atividades repetitivas. Os sistemas de informação são peças-chave no gerenciamento de dados e serviços em diversos tipos de trabalho, inclusive nos laboratórios de biomedicina e de bioquímica.

No contexto da informatização dos laboratórios de pesquisa, surgem os Sistemas de Gerenciamento de Informações de Laboratório (*Laboratory Information Management System* - LIMS), sistemas de informação utilizados para gerenciar informações dos laboratórios de pesquisa. Os LIMS se propõem a oferecer o gerenciamento de gravação e acesso aos dados de forma segura e satisfatória. Hoje existem diversos LIMS disponíveis no mercado, mas muitos deles são de código proprietário e com alto custo, o que se torna um impedimento para pequenos laboratórios. Outros LIMS são de código aberto, mas apresentam a maior parte dos seus dados de forma não estruturada, como dados gravados sob a forma de arquivos anexos, anotações ou descrição de experimentos, além de um grande volume de experimentos. Outro grupo de LIMS ainda são de código aberto, mas são modelados para laboratórios específicos, sendo de difícil adaptação em outros laboratórios, ainda que semelhantes, sem customização de *software*, pois os laboratórios possuem características e processos de trabalho bas-

tante peculiares. Com a enorme variedade de laboratórios, é necessário que se crie um LIMS com a estrutura de dados customizada para cada um deles. Essa necessidade de adaptação do *software* provoca aumento do tempo de desenvolvimento e do custo final da aplicação. Para flexibilizar a disponibilização do LIMS para os laboratórios, foi criado o *Sistema Integrado de Gerência de Laboratórios*(SIGLa), que permite ao laboratório adaptar o fluxo de dados do sistema de acordo com as suas próprias necessidades de trabalho sem que o código-fonte do sistema seja alterado [Simões *et al.*, 2010]. O SIGLa já é utilizado em diversos laboratórios de pesquisa, como a *Empresa Brasileira de Pesquisa Agropecuária*(EMBRAPA), o *Instituto Nacional de Metrologia, Qualidade e Tecnologia*(INMETRO) e o *Hospital das Clínicas da Universidade Federal de Minas Gerais*(HC/UFGM), promovendo ganho de produtividade e rastreamento das informações. No entanto, o primeiro módulo de consultas criado para o SIGLa não é satisfatório.

Este projeto propõe a modelagem e a implementação do SiglaSearch, uma engenharia de consulta para dados de *workflow* com interface customizada para os diferentes usuários do sistema. O SiglaSearch é flexibilizado para múltiplos fluxos de trabalho e exibe as informações relacionadas entre si, sem a necessidade de conversão do modelo de dados orientado a *workflow* para um modelo de dados convencional.

1.1 Motivação

A pesquisa biomédica sofreu um grande crescimento de dados gerado pela ampliação da legislação sobre os processos de laboratórios, avanço na instrumentação automática, melhoria dos processos de controle de qualidade e aumento da demanda por análise estatística [Hinton, 1995]. Para controlar esse volume de informação nos laboratórios de pesquisa não informatizados, toda geração e gravação de dados é feita manualmente através de sistemas de informação em papel, como cadernos de anotações de experimento e fichas, que devem ser cuidadosamente etiquetados, ordenados e guardados para segurança no armazenamento de dados e facilitar a busca da informação. Mas esse controle de dados manual tem o custo de busca da informação elevado quando ocorre aumento de informações. A informação, quando é guardada apenas em meio físico (através de papel), exige muito espaço para armazenamento e cuidado na ordenação dos materiais. Este modelo de armazenamento também dificulta o gerenciamento de concorrência de acesso e atualização dos dados, quando dois ou mais pesquisadores desejam acessar o mesmo documento [Li *et al.*, 2006]. Outro problema desses sistemas é o acesso lento à informação, já que o arquivo deve ser procurado pelo usuário do

laboratório. A gravação de qualquer dado também é lenta, pois o arquivo é acessado manualmente pelo usuário, para guardar a nova informação e retornar o arquivo ao seu devido lugar para que ele possa ser acessado posteriormente. Dessa forma, vemos que o sistema informatizado para gerência das informações de laboratório se torna de grande valia, mesmo para pequenos laboratórios de pesquisa.

Para controlar esse volume de informação, alguns laboratórios de pesquisa utilizam Sistemas de Gerenciamento de Informações de Laboratório (*Laboratory Information Management System* - LIMS), que propõem gerenciamento de gravação e acesso aos dados de forma segura e satisfatória. LIMS são sistemas de informação utilizados para gerenciar informações dos laboratórios de pesquisa. Um LIMS é capaz de gerenciar a explosão da informação, garantir a qualidade, reduzir erros na entrada de dados e prover eficiência e eficácia das atividades do laboratório [Hinton, 1995]. Devido aos laboratórios de pesquisa possuírem características e processos de trabalho bastante peculiares e à grande variedade de laboratórios, os LIMS são criados com a estrutura de dados customizada para cada um deles. Essa necessidade de adaptação do *software* provoca aumento do tempo de desenvolvimento e do custo final da aplicação, o que torna a implantação de um LIMS inviável para pequenos laboratórios.

Para flexibilizar a disponibilização do LIMS para os laboratórios, foi criado o *Sistema Integrado de Gerência de Laboratórios*(SIGLa), que permite ao laboratório adaptar o fluxo de dados do sistema de acordo com as suas próprias necessidades de trabalho sem que o código-fonte do sistema seja alterado [Simões *et al.*, 2010]. O SIGLa é um LIMS que oferece adaptação do sistema para diferentes laboratórios através de gerenciamento de *workflows* ou fluxos de trabalho. Os *workflows* são editados através da ferramenta de código aberto, como o *Together Workflow Editor*(TWE) [Editor, 2015]. O fluxo de trabalho editado é gravado em um arquivo *eXtensible Markup Language*(XML), com uso do padrão *XML Process Definition Language*(XPDL).

Dentre as funções que um LIMS deve oferecer, as relacionadas à área de consulta são: gerir a qualidade dos dados, consultar dados de auditoria e fazer cruzamento de informações entre diferentes entidades de dados com a finalidade de descobrir padrões. O SIGLa atende as duas primeiras funções citadas, mas não existe possibilidade de cruzamento de dados entre diferentes entidades do contexto de um laboratório. Embora o SIGLa cumpra com eficiência a função de persistência de dados de diferentes laboratórios, o acesso aos dados através de consulta não atende à perspectiva dos laboratórios de pesquisa.

A recuperação da informação é tão importante quanto o seu armazenamento. Através dela é possível extrair dos dados pistas importantes para o aperfeiçoamento dos experimentos em laboratório. É necessário que os resultados das consultas sejam

eficazes na descoberta de padrões de acontecimentos e fatores favoráveis à geração de experimentos de sucesso. Os resultados devem ser apresentados de forma clara para que os usuários da informação possam analisá-la com rapidez e sem excesso de informações não úteis que ofusquem o evento pesquisado.

O *Laboratório de Universalização do Acesso* (LUAR), laboratório de pesquisas do DCC/UFMG e desenvolvedor do SIGLa, colabora com diversas entidades de pesquisa, através da disponibilização do sistema SIGLa. Essa cooperação tem auxiliado na geração de resultados bastante satisfatórios nas pesquisas dessas instituições. No entanto, essas organizações têm solicitado melhorias do módulo de consultas, para facilitar o rastreamento de informações. Torna-se necessário, portanto, o desenvolvimento de um módulo avançado de consultas para suprir essas necessidades no sistema SIGLa.

Uma das aplicações requisitadas para o novo módulo de consulta é o rastreamento de dados do sistema que possibilite identificar fatores relacionados com os melhores experimentos. Em um experimento de laboratório vários procedimentos são feitos. Determinados subconjuntos de procedimentos podem ser fundamentais para o sucesso de um experimento. Pode-se citar também os pesquisadores que trabalham no diagnóstico de doenças. A partir de um conjunto de registros de pacientes em um determinado estudo, pode-se identificar um subconjunto de sintomas suficientes para sugerir o diagnóstico de uma doença. Na Figura 1.1, temos o esquema que ilustra o conjunto de registros de pacientes e a presença ou ausência de cada sintoma estudado. Na primeira coluna são listados os pacientes identificados por um número sequencial. Em seguida temos sete colunas que indicam presença ou ausência de sintomas. Considera-se que todos os pacientes da tabela foram diagnosticados com a doença estudada. Observamos que os sintomas *IV* e *VI* estão presentes em todos os pacientes listados. Podemos inferir, portanto, que a presença do conjunto dos sintomas *IV* e *VI* é um forte indício da doença.

O objetivo desse trabalho é o desenvolvimento do *framework* SiglaSearch, um novo módulo de consulta para o SIGLa capaz de oferecer aos usuários do LIMS qualquer consulta que tenha os filtros flexíveis para trazer os resultados que auxiliem a tomada de decisão sobre os resultados da pesquisa, próximos experimentos e estudos. O sistema deve realizar o cruzamento de dados das entidades utilizadas pelos laboratórios, não explícitas no modelo de dados do SIGLa, entre todos os passos de execução de um *workflow*. Os dados devem ser apresentados em uma única visualização consolidada, em formato tabular. Deve inclusive possibilitar a sumarização dos dados. O novo sistema auxilia a descoberta de padrões de eventos pesquisados, através de consulta aos fatores que estão relacionados à geração de eventos ou estudos esperados. Os procedimentos de busca de informações devem ser executados de forma genérica, para

Paciente	Sintoma I	Sintoma II	Sintoma III	Sintoma IV	Sintoma V	Sintoma VI	Sintoma VII
101	✓		✓	✓		✓	✓
102		✓		✓	✓	✓	✓
103	✓	✓	✓	✓		✓	✓
104	✓			✓		✓	
105				✓	✓	✓	✓
106		✓	✓	✓		✓	
107	✓	✓		✓		✓	✓
108		✓	✓	✓		✓	
109				✓	✓	✓	
110	✓			✓	✓	✓	

↓ ↓

Forte indicio da doença

Figura 1.1. Exemplo ilustrativo de pesquisa para identificação de sintomas relacionados a uma doença estudada

que o sistema de consulta interprete qualquer fluxo de dados de um laboratório, sem customização de *software*.

1.2 Definição do Problema

O sistema SIGLa foi modelado utilizando um Banco de Dados Relacional (*Relational Database Management System - RDBMS*). Em RDBMS, a recuperação de dados é baseada na álgebra relacional, utilizando as chaves estrangeiras para mapeamento dos relacionamentos entre as tabelas [Elmasri & Navathe, 2011]. Apesar de ser modelado usando RDBMS, o modelo de dados do SIGLa foi construído em função dos conceitos de *workflow*, como *atributo*, *atividade* e *instância*, para garantir a alta abstração do modelo de dados. Os conceitos de fluxo de trabalho são mapeados como entidades no modelo de dados. Conseqüentemente, a busca de dados das entidades conhecidas pelo usuário de um laboratório específico não pode ser feita através das técnicas existentes para busca de dados em RDBMS.

1.2.1 Sistema de consulta antigo do SIGLa

O sistema SIGLa, antes da criação do novo sistema de consulta, permitia realizar consultas através da filtragem de qualquer conjunto de atributos de um *workflow*. Como resultado, o sistema retornava todos os dados das entidades cujos atributos

filtrados satisfaziam a todos os critérios de pesquisa. Porém o produto final da consulta agrega pouco valor de análise, pois os resultados são apresentados separadamente por atividade, sem apresentar os relacionamentos entre esses dados. O sistema não oferece junção dos registros de diferentes atividades das instâncias do *workflow*. Entretanto muitas dessas informações só fazem sentido quando estão relacionadas com outros campos de informação. Dessa forma, a análise sobre os resultados é complicada e inconclusiva para a maioria das consultas. A Figura 1.2 é um exemplo de geração dos resultados do sistema não relacionados, para o fluxo de trabalho de Controle de Qualidade de um laboratório.

Recebimento da Substância Teste

Norma de Origem	Responsável pela entrega
Norma de armazenamento de frascos	Fernando
Norma de armazenamento de ácidos	
Norma de armazenamento de bases	Fernando

Dados da Substância Teste

id Original da Substância Teste	Tipo de solução
2344	Soro
234554	Tripsina
	Meio de cultura

Testes Solicitados

Data da Solicitação	Teste de Micoplasma - PCR
23/12/2014	Sim
23/12/2014	
23/12/2014	

Teste de Autenticidade - Análise crítica

Autenticidade - Identificação	Autenticidade - Quantidade
Satisfatória	Satisfatória
Satisfatória	Insatisfatória

Figura 1.2. Resultados gerados pelo antigo módulo de consulta do sistema SIGLa

O sistema de consulta também não oferecia opção de sumarização de informações em função dos demais campos buscadas, como a extração da média, dos valores

máximos e mínimos, soma de valores e contagem de registros em função das demais informações. Essa consulta não era otimizada, pois acessava o banco centenas de vezes para geração de dezenas de resultados. A pesquisa era feita em três fases. A primeira para identificação e busca das instâncias do *workflow* que satisfaziam aos critérios de consulta. A segunda fase, para a busca de cada atividade selecionada, para cada instância, através de iterações nas instâncias. E a terceira, para cada registro de atividade, a busca dos atributos selecionados. Essa abordagem impede a aplicação de otimizações de consultas feitas durante a criação dos *scripts* de consulta e de otimizações feitas internamente no próprio Sistema Gerenciador de Banco de Dados (SGBD). O esquema da Figura 1.3 mostra o funcionamento do sistema antigo de consulta do SIGLa.

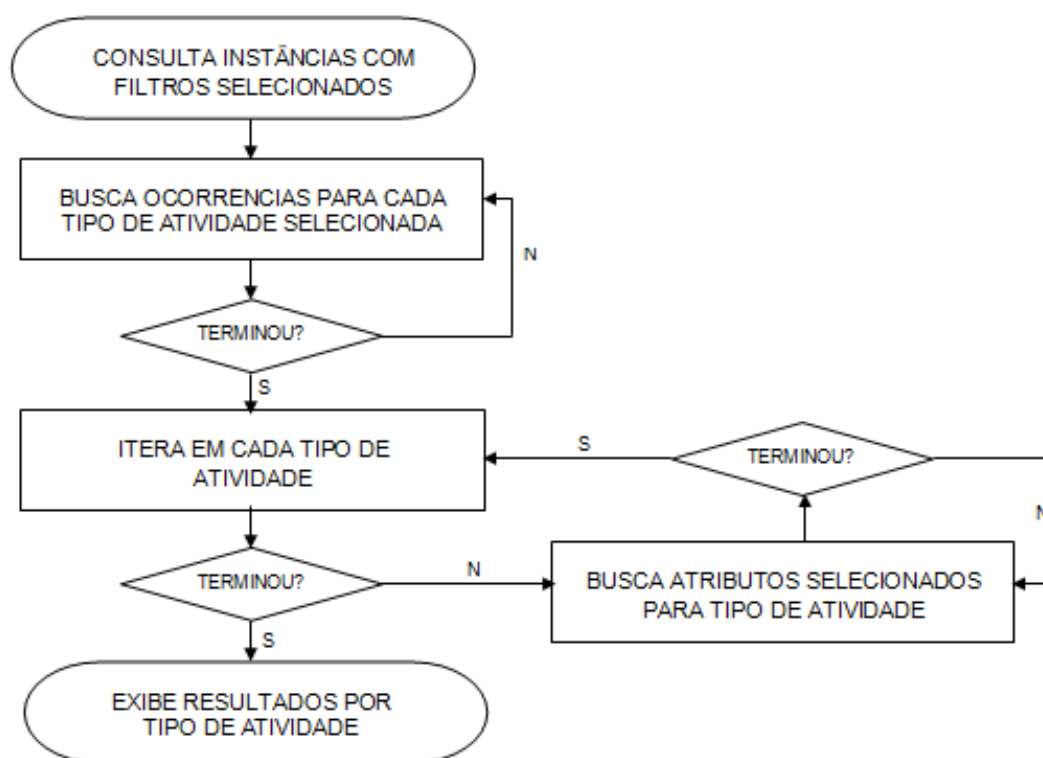


Figura 1.3. Funcionamento do antigo sistema de consulta do SIGLa

1.3 Contribuições

Esse projeto propõe a modelagem e a implementação do SiglaSearch, uma engenharia de consulta para dados de *workflow* com interface customizada para os diferentes usuários do sistema, flexibilizada para múltiplos fluxos de trabalho, que exiba as informações relacionadas entre si, sem a necessidade de conversão do modelo de

dados orientado a *workflow* para um modelo de dados convencional. O novo sistema de consulta desenvolvido para o SIGLa traz como melhoria a pesquisa em dados de *workflow*, porque os sistemas de consulta existentes para esses dados só oferecem busca por palavras-chave [Cohen-Boulakia & Leser, 2011] enquanto o SiglaSearch propõe a pesquisa através de subconjunto de informações de *workflow*. O SiglaSearch permite a busca de subconjuntos de dados, filtragem e ordenação de qualquer uma das informações do *workflow* e sumarização de resultados. Os relacionamentos são extraídos de forma automática sem especificações adicionais do usuário, com exibição das informações em formato tabular, para que os usuários possam focar em subconjuntos de informações que facilitem avaliações específicas. Os usuários podem, por exemplo, buscar eventos elite, aqueles que representam os melhores resultados para a pesquisa. A engenharia de consulta desenvolvida representa tanto uma melhoria para o SIGLa enquanto LIMS quanto uma proposta para pesquisa em *workflows* para a comunidade científica. O SiglaSearch também fornece uma interface de consulta para os usuários, para busca de resultados em poucos passos de forma intuitiva e persistência das consultas sob a forma de relatórios que podem ser executados por outros usuários que trabalhem em atividades mais específicas do laboratório de pesquisa.

1.4 Estrutura da Dissertação

Essa dissertação está organizada da seguinte forma: Este Capítulo (1) contém uma introdução sobre o projeto. O Capítulo 2 apresenta os fundamentos relacionados a esta dissertação e os trabalhos relacionados. O Capítulo 3 mostra a interface do sistema de consulta, as funções que podem ser utilizadas pelos usuários e as premissas utilizadas para construir o sistema com boa usabilidade. O Capítulo 4 apresenta o modelo de dados do SIGLa, as opções de armazenamento de dados nesse sistema, a arquitetura e a implementação da engenharia de consulta criada para o SiglaSearch e as tecnologias utilizadas para desenvolvimento do trabalho. O Capítulo 5 apresenta três casos de uso do SiglaSearch para três fluxos de trabalhos diferentes: um para Gestão de Projetos, um para estudos da doença Meningite e outro para estudos da doença Paracoccidioidomicose (PCM). Por fim, as conclusões e os trabalhos futuros são descritos no Capítulo 6.

Capítulo 2

Fundamentação teórica

2.1 Fundamentos

2.1.1 Bioinformática

Bioinformática é o campo da ciência em que a biologia, a ciência da computação e tecnologia da informação se fundem em uma única disciplina. Existem vários ramos dentro da bioinformática como o desenvolvimento de novos algoritmos e estatísticas para avaliação das relações entre membros de grandes conjuntos de dados; a análise e a interpretação de vários tipos de dados, incluindo sequências de nucleotídeos e de aminoácidos, proteínas, domínios e estruturas de proteínas e o desenvolvimento e a implementação de ferramentas que possibilitam o acesso eficiente à gestão de diferentes tipos de informação [NCBI, 2015]. O projeto que será desenvolvido tem como foco a geração de ferramenta que amplifique a gerência e a análise de experimentos para facilitar a rastreabilidade das informações.

2.1.2 *Workflow*

Conforme definição da *Workflow Management Coalition*(WfMC) [WfMC, 2015], um *workflow*, também chamado de fluxo de trabalho, representa a automação do processo de negócio, através da compilação de documentos, informações ou tarefas, que são passadas de um participante a outro na execução de uma ação. Um fluxo de trabalho contém processos compostos por atividades, que podem estar relacionadas e causar efeitos entre si, através de regras de negócio pré-definidas. A ordem de execução das atividades é registrada através da criação de transições. As transições, as variáveis das atividades e demais regras são definidas no momento de criação do *work-*

flow. O *workflow* é armazenado em um arquivo no padrão *XML Process Definition Language*(XPDL). O *XML Process Definition Language*(XPDL) é um padrão definido pela WfMC para interfaceamento de fluxos de trabalho entre diferentes *softwares* ou ferramentas que trabalhem separadamente [WfMC, 2015].

Segundo Maguire e colaboradores [2013], ao projetar *workflows*, modelamos uma sequência de passos ou conjuntos de passos que executam em paralelo. Um fluxo de trabalho normalmente descreve uma seqüência de etapas para a realização de um conjunto de tarefas de um determinado trabalho. Quando os *workflows* seguem a Notação de Modelagem de Processos de Negócio (*Business Process Modeling Notation* - BPMN), são capazes de transmitir fluxo de trabalho, fluxo de dados e interações entre processos de sistemas complexos, sendo adequados para descrever experiências biológicas. Um *workflow* pode ser entendido como um grafo direcionado. No caso do LIMS SIGLa [Simões *et al.*, 2010], o *workflow* é um grafo direcionado onde os nós representam as atividades e as arestas representam as transições entre essas atividades. Uma instância de *workflow* é uma instanciação específica de um *workflow* para um problema particular, com definição dos dados de entrada [Deelman *et al.*, 2009].

A aplicação de *workflows* em pesquisa científica busca ampliar a capacidade dos usuários para definição dos seus fluxos de trabalho, reutilização e integração facilitada entre fluxos de trabalho personalizados e diminuição do tempo para desenvolvê-los [Cohen-Boulakia & Leser, 2011]. Os sistemas desenvolvidos em função de *workflows* visam principalmente apoiar cientistas do domínio de uma área específica no desenvolvimento, execução e acompanhamento dos programas de análise de dados codificados como fluxos de trabalho, com apresentação dos processos em um nível maior de abstração [Deelman *et al.*, 2009].

2.1.3 **Laboratory Information Management Systems (LIMS)**

Laboratory Information Management Systems (LIMS) são sistemas de informação utilizados para gerenciar informações dos laboratórios de pesquisa. As principais funções de um LIMS são gerenciamento da explosão da informação, garantia da qualidade, redução de erros na entrada de dados e maior eficiência e eficácia na execução das atividades do laboratório [Hinton, 1995]. Wendl e colaboradores [2007] afirmam que os LIMS devem oferecer robustez, capacidade adequada para representação de dados complexos, flexibilidade para evolução que acompanhe mudanças de processos, controle de processos, integridade e verificação de dados.

2.1.4 *Good Laboratory Practices (GLP)*

Segundo Estevam dos Santos [2010], *Good Laboratory Practices (GLP)* ou *Boas Práticas de Laboratório (BPL)* são um conjunto de regras aplicadas aos processos da organização, onde os estudos do laboratório são planejados, executados, monitorados, registrados e relatados. A implantação das Boas Práticas de Laboratório em laboratórios de ensaio é regulada pela norma ABNT NBR ISO/IEC 17025. As BPLs promovem a avaliação e a certificação da qualidade dos serviços do laboratório, o aprimoramento contínuo e a garantia da qualidade dos ensaios realizados. A qualidade dos ensaios de laboratório tem grande importância pois aumenta a credibilidade e a confiabilidade dos dados, evita erros e repetições de trabalho, facilita a rastreabilidade dos dados e melhora a organização das informações [Estevam dos Santos, 2010].

As BPLs devem ser usadas em todos os laboratórios de testes de espécies biológicas para diagnósticos, cuidados com pacientes, controle de doenças e em pesquisas. As BPLs contemplam a infraestrutura do laboratório, a formação e desenvolvimento de pessoal, os equipamentos, os reagentes, os produtos químicos, a coleta de espécimes, formulários de requisição, registros de amostras, os relatórios de resultados de testes, a gestão de dados, os procedimentos operacionais padrão, a segurança do laboratório, a biosegurança laboratorial, a garantia da qualidade e a auditoria interna [Selvakumar, 2010].

2.1.5 **Armazenamento e recuperação de informação em Banco de Dados**

Para armazenar dados, são utilizados comumente arquivos físicos, como cadernos de anotações e fichários. Outros utilizam documentos de texto ou planilhas eletrônicas. Mas nesses casos a gravação, o controle de acesso de múltiplos usuários e a busca da informação são lentos ou precários. Para automatizar e melhorar o armazenamento e a recuperação de bancos de dados, foram criados os Sistemas Gerenciadores de Banco de Dados (SGBDs). Um SGBD é uma coleção de programas que permite aos usuários criar e manter um banco de dados e facilita os processos de definição, construção, manipulação e compartilhamento de bancos de dados entre vários usuários e aplicações [Elmasri & Navathe, 2011]. Os SGBDs podem usar diversos paradigmas, como os sistemas hierárquicos, os sistemas orientados a grafo e os sistemas de bancos de dados relacionais.

O Sistema de Gerenciamento de Banco de Dados Relacional (*Relational Database Management System - RDBMS*) foi criado na década de 1970 como um sistema

para permitir a interação de dados em várias tabelas através de linguagem algébrica formal, com uso da Linguagem de Consulta Estruturada (*Structured Query Language* - SQL) para consulta de dados [Jin *et al.*, 2014]. Conforme Batra & Tyagi [2012], essa tecnologia domina a indústria desde os anos 80, para armazenamento e recuperação da informação. Os sistemas de banco de dados relacionais são mais maduros e estáveis e existem empresas que provêem uma boa estrutura de suporte como Oracle [Loney, 2008] e MySQL [MySQL, 2015]. Os RDBMS possuem a linguagem de consulta padrão SQL, que não se diferencia muito entre as implementações, e oferecem extenso suporte para aplicações multiusuário [Batra & Tyagi, 2012].

Os RDBMS apresentam como deficiências a dependência de um esquema rígido que dificulta a adição de novos relacionamentos entre objetos e o custo de processamento de consultas quando elas envolvem grandes volumes de dados em várias tabelas, devido ao custo de junção entre essas tabelas [Batra & Tyagi, 2012]. Os RDBMS podem ser menos adequados para gerenciar esquemas *ad hoc* que evoluem ao longo do tempo, podendo ser necessária a reestruturação de todo o esquema. Contudo, os custos de junção em várias tabelas do banco de dados podem ser amenizados com a criação de índices em colunas estratégicas do banco de dados [Elmasri & Navathe, 2011].

Conforme Batra & Tyagi [2012], o armazenamento em grafo é o meio mais natural para aplicações onde a essência dos dados está na ligação entre objetos em uma rede, como as redes sociais, por exemplo. Nessa tecnologia, novos relacionamentos podem ser inseridos sem a necessidade de reestruturar o esquema. Os dados são representados por nós (objetos), arestas (relacionamentos) e propriedades. Como exemplos de bancos de dados em grafo pode-se citar Infogrid, HypergraphDB, Jena, DEx, FlockdB e Neo4j. Os bancos de dados em grafo apresentam algumas fraquezas. Eles não são maduros e não oferecem alta estabilidade. Também não possuem mecanismos para gerenciar restrições de segurança e aplicações multiusuário. Suas linguagens de consulta possuem implementações muito diferentes, o que dificulta a sua utilização. Como exemplos de linguagens de consulta em bancos de dados em grafo pode-se citar a SPARQL, a Gremlin e a Cypher Query. Embora nos últimos anos os RDBMS estejam perdendo algum espaço para os bancos de dados em grafos, aqueles são mais seguros que estes [Angles & Gutierrez, 2008].

Bancos de dados NoSQL são bancos de dados não relacionais, incluindo os bancos de dados hierárquicos, orientados a grafo e orientados a objeto [Leavitt, 2010]. Hoje os bancos de dados NoSQL ganham espaço no mercado, como alternativa ao RDBMS, pois apresentam alta performance em consultas de grandes volumes de dados. Não possuem esquema, de forma que os dados armazenados são semi-estruturados. Embora a consulta seja mais rápida, bancos NoSQL não garantem consistência ampla de

dados, pois suas transações não implementam as propriedades ACID (Atomicidade, Consistência, Isolamento e Durabilidade) [Han *et al.*, 2011].

Lee e colaboradores [2013] propõem como alternativas de RDBMS o uso de bancos de dados NoSQL e XML. Como dados clínicos são dinâmicos e esporádicos, a performance de consulta pode ser prejudicada devido ao fato de as tabelas armazenarem muitos campos nulos. E a grande quantidade de atributos para descrever eventos científicos pode inviabilizar a criação de tabelas, pois existe um limite máximo para a quantidade de colunas em uma tabela de RDBMS. Esse artigo afirma que bancos de dados clínicos não precisam ter restrições de integridade, devido aos dados serem volumosos e sua definição de valores ser complexa. Isso gera a necessidade de tratamento de dados posterior e geração das ontologias para consulta aos dados. A ausência de restrições de integridade pode gerar complicações em processos de consulta e integração de bases de dados.

Para os LIMS, o armazenamento em banco de dados com extrema rapidez não é necessário, pois normalmente a carga de dados é feita de forma gradual. O caso de uso mais natural de um LIMS é a entrada de dados quando estes vão sendo gerados pelo usuário a partir de entrevistas, geração de experimentos e análise de dados. Mesmo a carga de grandes volumes de dados em um LIMS não justifica a eliminação da fase de análise da qualidade e das restrições de integridade oferecidas pelo banco de dados relacional. Essas verificações previnem problemas posteriores de melhoria da qualidade dos dados de entrada ou dificuldades na integração de diferentes bases de dados. Queiroz [2009], por exemplo, cita os altos custos e a perda de um grande volume de informação na consolidação de diferentes bases de dados do Sistema Único de Saúde (SUS) devido à baixa consistência das bases de dados. A modelagem feita pelo SIGLa aproveita as características e as funcionalidades de RDBMS de relação de integridade. Ao mesmo tempo, contorna a necessidade da geração do modelo de dados como pré-requisito para desenvolvimento de um LIMS, devido a esse sistema ter uma estrutura modelada em função dos conceitos de *workflow*. E no SIGLa não há limitação de atributos por entidade, já que os atributos são gravados em uma tabela separada que representa um único atributo em cada registro de tabela.

A tecnologia RDBMS foi utilizada para implementação do banco de dados do *framework* SIGLa, devido aos seus níveis de maturidade, suporte, padronização e alto nível de consistência. A rigidez dos RDBMS, foi contornada pelo SIGLa através de um esquema relacional especial cujos relacionamentos são orientados aos metadados do *workflow*. Mas essa solução de contorno gerou, como efeito colateral, uma complexidade não tratada para a geração de consulta aos dados. Tal complexidade é tratada através do SiglaSearch, a engenharia de consulta deste projeto e será explicada no Capítulo 4.

2.1.6 Banco de Dados Relacional (*Relational Database Management System* - RDBMS)

Para entender o funcionamento do SiglaSearch, a engenharia de consulta criada para o SIGLa, é necessário entender conceitos fundamentais dos Bancos de Dados Relacionais (RDBMS), conforme descritos por Elmasri & Navathe [2011]. Um banco de dados modelado com a tecnologia RDBMS é descrito através de um modelo Entidade-Relacionamento (ER) e é representado como uma coleção de dados relacionados. Entidades representam objetos do mundo real que são de interesse para alguma aplicação, com uma existência independente, como pessoa e carro. Cada entidade é composta por um conjunto de atributos, que são propriedades que descrevem as entidades. Os atributos podem ser classificados como simples ou compostos, monovalorados ou multivalorados, armazenados ou derivados. Eles são compostos se podem ser divididos em subpartes com significados diferentes ou simples se forem atômicos. São monovalorados se possuem um único valor para a entidade ou multivalorados se possui quantidades de valores diferentes para cada entidade. São armazenados se estão gravados no banco de dados ou derivados se são obtidos a partir de outras informações gravadas. Todo atributo possui um domínio, que define o conjunto de valores que podem ser atribuídos ao atributo. A idade da pessoa, por exemplo, tem de ser um valor inteiro. Um tipo de entidade define uma coleção de entidades que possuem os mesmos atributos, sendo descrita pelo seu nome e seus atributos. O tipo de entidade *Livro*, com seus atributos *autor*, *título*, *editora* e *volume*, por exemplo, descreve todas as entidades que possuem as mesmas características descritas nessa entidade. Uma coleção de entidades do mesmo tipo são gravadas no banco de dados sob a forma de tabela. A chave primária é um conjunto de atributos que identificam de forma única uma entidade dentre todas as entidades do mesmo tipo.

O modelo relacional representa o banco de dados como uma coleção de relações. Cada relação é uma coleção de entidades do mesmo tipo e é gravada no banco de dados sob a forma de uma tabela de dados. Uma linha da tabela é denominada *tupla*, um cabeçalho de coluna é conhecido como *atributo*, e a tabela é chamada *relação*. O tipo de dado que descreve os tipos de valores que podem aparecer em cada coluna é representado pelo *domínio* de valores possíveis. Um esquema de relação R , indicado por $R(A_1, A_2, \dots, A_n)$, é composto de um nome de relação R e de uma lista de atributos A_1, A_2, \dots, A_n . Cada atributo A_i é o nome de um papel desempenhado por algum domínio D no esquema de relação R . D é chamado de domínio de A_i e é indicado por $dom(A_i)$. Uma relação r de R , indicado por $r(R)$, é um conjunto de n -tuplas $r = t_1, t_2, \dots, t_m$. Cada n -tupla t é uma lista ordenada de n valores $t = (v_1, v_2, \dots, v_n)$ em

que cada valor v_i , $1 \leq i \leq n$, é um elemento do $dom(A_i)$ ou valor nulo.

Para garantir a consistência de dados em um banco relacional, são definidas as restrições de integridade de entidade e de integridade referencial. As restrições de integridade de entidade estabelecem que nenhum valor de chave primária pode ser nulo. E as restrições de integridade referencial garantem a consistência do relacionamento entre tuplas de duas relações. Esta declara que uma tupla de uma relação que faz referência a outra relação deve-se referir a uma tupla existente nesta relação. As relações entre duas tuplas são definidas por uma chave estrangeira. Outras restrições devem ser aplicadas ao banco de dados, como gatilhos e checagens, para restrições específicas que dependem de cada contexto de banco de dados.

O conjunto básico de operações para o modelo relacional é a *álgebra relacional*. Através dela o usuário pode especificar as operações básicas para recuperação de informação. O *cálculo relacional* disponibiliza uma notação declarativa para a especificação de consultas relacionais. A operação de seleção é usada para selecionar um subconjunto de tuplas de uma relação que satisfaça uma condição de seleção, sendo indicada por $\sigma_{\langle \text{condicao de selecao} \rangle}(R)$. A operação de projeção seleciona um subconjunto de colunas da relação e é dada por $\pi_{\langle \text{lista de atributos} \rangle}(R)$. Há três operações de conjunto permitidas, a união (\cup), a interseção (\cap) e a subtração ($-$). A operação de produto cartesiado, indicada pelo operador \times , é usada para combinar as tuplas de duas relações de forma combinatória. A operação de junção, indicada por \bowtie , combina as tuplas relacionadas em duas relações dentro de uma tupla única. Existem outros operadores, mas eles não serão tratados aqui por não serem utilizados no escopo do trabalho. A recuperação de informação em RDBMS é feita através da linguagem SQL, que implementa esses operadores citados.

2.1.7 Ontologia

Uma ontologia é definida como uma representação formal de termos, relações, conceitos e entidades dentro de um determinado domínio [Wild *et al.*, 2012]. Essa dissertação está relacionada com a ontologia aplicada à Web Semântica, para aplicação de significados a conteúdos, definida através da linguagem *Web Ontology Language* (OWL). A Web Semântica é definida com uma coleção de dados ligados por um caminho para que possam ser processados [Jain & Singh, 2013]. Como exemplos de geradores de ontologias pode-se citar OntoMinD [Al-Jadir *et al.*, 2010] e Oracle Semantic [Murray, 2008].

Os dados de uma ontologia são armazenados através de triplas no formato *Resource Description Framework* (RDF), uma recomendação do *World Wide Web Con-*

sortium (W3C) para padronização de metadados de recursos da Web. As triplas RDF são expressões que contêm três partes: sujeito, propriedade e objeto. Os sujeitos, também conhecidos como recursos, são tudo que pode ser descrito por uma expressão RDF. Como exemplos de sujeito, pode-se citar *carro*, *João* e *Paris*. A propriedade é qualquer aspecto, característica, atributo ou relação para descrever um recurso. São exemplos de propriedade *fazer*, *Cidadania* e *proprietário*. Um objeto atribui valor à propriedade. Estes podem ser *Ford* e *Reino Unido*. As triplas RDFs representam os dados como um grafo direcionado e são usados para extração dos dados e seus relacionamentos nas consultas [Britton *et al.*, 2013].

O armazenamento de tuplas RDF é classificado em duas categorias: armazenamento nativo, que usa customizações binárias para representação de dados RDF, e armazenamento relacional, onde os dados RDF são armazenados em tabelas apropriadas de RDBMS. O armazenamento nativo é feito em arquivos RDF com sintaxe XML (eXtensible Markup Language). A consulta em arquivos RDF é feita normalmente através da linguagem SPARQL [Bornea *et al.*, 2013]. O armazenamento relacional apresenta vantagens em relação ao armazenamento nativo, como escalabilidade, suporte a transações, compressão e segurança [Bornea *et al.*, 2013]. O armazenamento relacional também permite que a consulta de informações semânticas possam ser traduzida para consulta em banco de dados relacionais [Ouyang *et al.*, 2010].

Há uma série de pesquisas em desenvolvimento para estudar a conversão de bancos de dados relacionais em ontologias para buscas na Web. Segundo Jain & Singh [2013], essa é uma importante área de pesquisa atualmente. Há numerosas técnicas e ferramentas para converter bancos de dados relacionais em ontologias. Alguns exemplos dessas abordagens são RDBOnto, Data Semantic Preservation, DB2OWL, R2O, D2RQ, Semantic Bridge e DartGrid Semantic. Britton e colaboradores [2013] sugerem a conversão de dados de bancos de dados relacional em triplas armazenadas no próprio banco de dados relacional, em um novo esquema de tabelas específicas para armazenamento dos dados da ontologia. Nesse modelo, a busca das triplas RDF é feita através de consultas em SQL. As triplas RDFs são indexadas com a fonte de dados de origem, o que mantém os dados sempre atualizados, mas não elimina a redundância de dados. He-ping e colaboradores [2008] apresentam um gerador de ontologia chamado OWLFROMDB, que pode converter um banco de dados relacional para uma ontologia OWL também armazenada em RDBMS de forma automática. Isso reduz o volume de trabalho para construção de ontologia e economiza tempo e aumenta a eficiência do processo. A conversão automática da ontologia é possível pela correspondência dos conceitos de tabelas, campos e registros do banco de dados relacional com os conceitos de classes, propriedades e instâncias da ontologia OWL. Sequeda & Miranker [2013]

apresentam uma solução para a geração de ontologia em bancos de dados relacionais, através de visualizações de banco de dados não materializadas, solução que não gera redundância de dados, diferente de outros sistemas de conversão. Nesse sistema, os relacionamentos entre as entidades são extraídos através das chaves estrangeiras. Bornea e colaboradores [2013] apresentam os desafios de armazenamentos em RDBMS, devido aos dados esparsos e a variedade e dinamicidade do conhecimento da ontologia. Os dados esparsos nas tabelas de dados diminuem a performance de processamento das consultas e a rigidez dos bancos de dados relacionais são conhecidas por obrigar mudanças de esquema frequentes em caso de dados dinâmicos. Esse trabalho sugere, portanto, a gravação de ontologias em arquivos RDF, para flexibilizar o esquema e garantir a escalabilidade das consultas. Mas no caso de armazenamento em arquivos RDF, a consulta deve ser feita através de SPARQL, uma linguagem que não é facilmente compreendida e pode gerar um trabalho árduo para usuários com pouco conhecimento em linguagens de programação [Jain & Singh, 2013].

O desenvolvimento de ontologias aplicadas à Web Semântica representa vários benefícios para a ciência, como a pesquisa e a descoberta de drogas no estudo da relação entre drogas e efeitos colaterais, pois possibilita a integração entre diferentes bancos de dados e posterior mineração das informações consolidadas [Wild *et al.*, 2012]. No entanto, a conversão de dados para ontologia gera redundância de armazenamento de dados, gasta tempo de processamento e gasto de espaço em disco. Para bases dinâmicas, onde os dados sempre são alterados, é necessário manter as duas fontes de dados ativas, uma para gravação de dados com segurança e controle de transação (a base de dados de origem), e outra para busca otimizada de dados (a base de dados convertida em ontologia). A geração da ontologia pode ser um gargalo para o desenvolvimento de uma pesquisa em dados que são frequentemente alterados, pois exige que profissionais de tecnologia da informação façam o serviço e, neste caso, a disponibilização dos dados atualizados para pesquisa não é imediata. O trabalho de novas conversões da base de dados em ontologia pode ser eliminado quando a geração da ontologia é automática e os dados da ontologia estão indexados com os dados de origem, mas essas soluções só existem para as bases de dados modeladas para um problema específico, onde as entidades estão relacionadas em função dos dados e as ontologias podem extrair os relacionamentos com base nas chaves estrangeiras. As ontologias são interessantes para análises em bases de dados estáticas, como as 446 ontologias disponibilizadas no site BioPortal [BioPortal, 2015] à época de escrita dessa dissertação. Ontologias em banco de dados são utilizadas como solução alternativa para consulta em *workflow* em alguns trabalhos de pesquisa [Buil-Aranda *et al.*, 2013].

2.1.8 Usabilidade de sistemas

Segundo Benyon [2011], sistemas de alta usabilidade são eficientes, eficazes, intuitivos, seguros e com boa utilidade. São eficientes quando permitem a execução das tarefas mediante quantidade de esforço razoável. A eficácia é indicada pela presença de funções e conteúdos adequados e organizados. São intuitivos quando são assimilados e incorporados ao conhecimento do usuário e fixados durante algum tempo. Devem ser seguros em diferentes contextos. A utilidade diz respeito a oferecer as funcionalidades que o usuário deseja. Três princípios são importantes para a usabilidade: a apreensibilidade, a efetividade e a adaptabilidade. A apreensibilidade está relacionada com o aproveitamento da experiência anterior do usuário para aprender o novo sistema. A efetividade é capacidade de mudar decisões anteriores com flexibilidade e e completar as tarefas com facilidade, eficácia e segurança. E a adaptabilidade está relacionada com a multiplicidade de caminhos para executar as funções e com a agradabilidade do usuário com a interface. Um sistema deve ter a interface projetada para atender a todas as pessoas afetadas pelo sistema ou que influenciam na definição dos requisitos.

Outra abordagem para a melhoria da usabilidade de sistemas consiste na Engenharia Semiótica. A Engenharia Semiótica reúne conceitos importantes para entendimento da comunicação entre os *designers* e os usuários, através do computador [De Souza, 2005]. Segundo Prates & Barbosa [2007], Semiótica é uma disciplina que estuda os fenômenos de significação e comunicação. Seus principais conceitos são *signo*, *sistemas de significação* e *comunicação*. *Signo* é tudo que emite significado a alguém. *Significação* é o processo através do qual expressão e conteúdo de signos são estabelecidos, com base no contexto social e cultural dos usuários, produzindo e interpretando signos. A codificação entre expressão e conteúdo é denominada *sistemas de comunicação*. *Comunicação* é o processo onde pessoas produzem mensagens formadas por signos usando sistemas de significação para expressar determinados conteúdos. Existem três tipos de signos: metalinguísticos, estáticos e dinâmicos. Os *signos metalinguísticos* são aqueles presentes nos manuais de usuário, sistema de ajuda, dicas, mensagens de erro e instruções. Os *signos estáticos* expressam o estado do sistema, através de observação da interface, como estado dos botões, elemento de interação utilizado e opções selecionadas. E os *signos dinâmicos* expressam o comportamento do sistema, através da interação do usuário com ele. São exemplos de signos dinâmicos ação disparada por um botão e efeito de alteração de valor de atributo.

Preece e colaboradores [2015] exploram o *design* de interação em sistemas. O sistema desenvolvido deve ser fácil, eficaz e agradável de usar, a partir da perspectiva do usuário. Para desenvolver um sistema com boa usabilidade, é importante ter uma

melhor compreensão dos usuários. Diferentes usuários têm necessidades diferentes e os sistemas devem ser desenvolvidos de forma a atender todas expectativas. Há seis princípios de *design* que devem ser observados na construção de uma interface de sistema: visibilidade, *feedback*, restrições, mapeamento, consistência e *affordance*. A visibilidade infere que as funções mais visíveis para os usuários são mais facilmente encontradas e utilizadas, enquanto as funções mais ocultas não são encontradas de forma trivial e são compreendidas com menor facilidade. O *feedback* está relacionado com a capacidade de o sistema enviar mensagens ou gerar eventos como resposta às interações do usuário. O princípio das restrições diz respeito à capacidade de restringir o acesso do usuário com o sistema, para que ele acesse apenas as funções possíveis para a fase atual de uso, para evitar o acesso às opções incorretas e para prevenir erros. As restrições podem ser feitas desativando os botões não aplicados à situação de uso, por exemplo. O mapeamento se refere à relação entre os controles e os seus efeitos. A consistência se refere à criação de interface onde operações e elementos semelhantes produzem tarefas similares. E o princípio de *affordance* se refere aos itens de uma interface cujas funcionalidades são compreendidas pelo usuário sem a necessidade de uma explicação prévia, de forma intuitiva ou através de experiências anteriores.

O equilíbrio entre flexibilidade e usabilidade também deve ser explorado nos sistemas de consulta. Jarke & Vassiliou [1985] apresenta um critério de avaliação de linguagens de consulta em banco de dados. Esse critério trabalha com os conceitos de *funcionalidade* e *facilidade de uso*. A funcionalidade se refere à capacidade expressiva da linguagem, ou seja, a capacidade de obter a informação buscada. A facilidade de uso mede o custo de representar a necessidade de informação, estimando o esforço de formulação de consultas.

2.2 Trabalhos relacionados

A maioria dos pesquisadores científicos e clínicos de hoje dos laboratórios de pequeno porte usam planilhas eletrônicas para gerenciamento dos seus dados. Planilhas eletrônicas são fáceis de usar, prontamente disponíveis e permitem a coleta rápida de dados. No entanto, as planilhas são insuficientes para lidar com grandes conjuntos de dados, com relações complexas, dados de multimídia e grupos de investigação colaborativos [Li *et al.*, 2006]. Para melhorar os procedimentos de gerenciamento e acesso à informação, são criados sistemas de computação para auxílio a essas pesquisas. Dentre esses sistemas existem as aplicações mais robustas denominadas Sistemas de Gerenciamento de Informações de Laboratório (LIMS).

2.2.1 Sistemas de computação para auxílio às pesquisas

Há um grande número de sistemas computadorizados, não necessariamente classificados como LIMS, que exemplificam o suporte da tecnologia da informação em bioinformática. Scott e colaboradores [2013] descrevem o sistema MIMIC-II para gerenciamento de dados clínicos, que consiste na consolidação de diferentes bases de dados de pacientes em um único sistema. Esse sistema representa um trabalho interdisciplinar com resultados positivos em pesquisa de dados clínicos. O acesso aos dados clínicos deve se feito através de edição de consultas SQL. O usuário explora as tabelas e os relacionamentos através da ferramenta QueryBuilder e pode exportar os resultados em arquivos CSV (*Comma-Separated Values* - Valores Separados por Vírgula). Esse sistema requer que o usuário saiba a linguagem SQL e conheça o esquema de banco de dados do MIMIC-II. Os nomes das colunas não possuem descrições muito intuitivas e são apresentados como abreviações das descrições para nomear as colunas do banco de dados. Utiliza-se, por exemplo, os nomes *dob* e *dod* para data de nascimento e de óbito. As consultas admitem filtros simples de intervalo e de igualdade, reunidos por operações de conjunção. Como são necessários conhecimentos técnicos em banco de dados, é necessário que engenheiros de sistema atuem na extração e obtenção de dados estatísticos, o que requer mais tempo na geração de pesquisa para especificação dos critérios de consulta.

Hill e colaboradores [2013] apresentam um sistema de consulta modelado em banco de dados relacional, usando MySQL [MySQL, 2015], que pode ser manipulado por interface online ou SQL. Lee & Markley [2012] descrevem o sistema PACSY, um banco de dados relacional criado para pesquisa de proteínas, com a consolidação de três bases de dados. E possui ferramenta para o usuário selecionar tabelas e atributos, permitindo ao usuário incluir procedimentos mais avançados através de edição direta ao SQL. Hensley e colaboradores [2012] descrevem o sistema ZeBase, para armazenar informações genéticas e hábitos de diversas espécies de peixes analisadas em laboratórios. Guo e colaboradores [2013] descrevem um banco de dados específico para armazenamento de dados sobre o metabolismo da bactéria *Escherichia coli*. Robinson e colaboradores [2013] descrevem o sistema FlyAtlas, um banco de dados específico para o genoma da espécie *Drosophila melanogaster*. Esse sistema possui um conjunto de consultas pré-definidas e a inclusão de novas consultas só é possível através da escrita de novos programas de consulta. Schaab e colaboradores [2012] apresentam o banco de dados MaxQB modelado em Oracle [Loney, 2008], para armazenamento e consulta de dados de proteômica. O usuário pode navegar, consultar e obter dados de proteína através de uma interface Web. O sistema é modelado com uma única entidade, que re-

Característica	MIMIC-II	PACSY	ZeBase	FlyAtlas	MaxDB	dbSNP	CEBS	ESTIMA
Edição de consultas SQL	X	X				X		
Descrições de colunas pouco intuitivas	X					X		
Poucas opções de filtros	X	X			X			
Estrutura de dados rígida	X	X	X	X	X	X	X	X
Apenas consultas pré-compiladas			X	X				
Dependência de ontologias							X	X

Tabela 2.1. Limitações identificadas em alguns dos sistemas de apoio à pesquisa analisados

presenta a proteína. Usuários podem escolher campos para aplicar filtros de igualdade. O sistema mostra uma quantidade fixa de campos e o usuário pode ver mais detalhes da proteína clicando no respectivo registro. Saccone e colaboradores [2011] apresentam um sistema de banco de dados com informações de Polimorfismos de Nucleotídeo Único (Single Nucleotide Polymorphisms - SNPs), o dbSNP. Esse sistema disponibiliza uma consulta para que usuários acessem as informações através da edição de códigos em SQL, com auxílio de uma documentação sobre as tabelas e colunas. Waters e colaboradores [2008] apresentam o repositório CEBS (Chemical Effects in Biological Systems), um repositório público que integra dados de toxicidade com dados de microarranjos e proteômica. Esse repositório possui um conjunto de *workflows*, pré-definidos no sistema, como *Mostrar Todos os Estudos*, *Busca por Características de Estudos* e *Busca por Assuntos Característicos*. O esquema relacional do sistema possui grupos de tabelas, onde cada um desses grupos representa os dados de um *workflow*. As consultas aos dados são feitas após a geração de ontologias do banco de dados.

Kumar e colaboradores [2004] apresentam a ferramenta ESTIMA para gerenciamento e consulta de genomas. Coleções de Marcadores de Sequência Genética (Expressed Sequence Tag - EST) representam descrições parciais das partes transcritas de genomas. O sistema ESTIMA fornece um esquema de banco de dados para a gestão de ESTs e é associado a um conjunto de ferramentas Web personalizadas para busca de pares através do BLAST [Altschull *et al.*, 1990] e busca por classificação funcional através Gene Ontology (GO) Consortium [Consortium *et al.*, 2001], uma ontologia para representação dos conceitos de genes.

A Tabela 2.1 resume as limitações identificadas em alguns dos sistemas de apoio

à pesquisa analisados. Todos esses sistemas citados possuem uma base de dados rígida e a mudança desses dados obriga a mudança do esquema relacional e a customização de *software*. A rigidez desses sistemas impede a aplicação deles em outros cenários, ainda que parecidos, quando os conjuntos de dados não possuem a mesma estrutura. Muitos desses sistemas também não possuem interface de consulta, e o usuário deve ter conhecimentos de lógica de programação para a edição de consultas SQL para os interpretadores de código de consulta. Alguns desses sistemas possuem interface de consulta bem simplificada, com uma quantidade mínima de funções. Ainda existem aqueles sistemas que disponibilizam telas com parâmetros fixos de consulta, onde qualquer alteração só pode ser feita pelo desenvolvedor do sistema, através de customização.

2.2.2 *Laboratory Information Management Systems (LIMS)*

Atualmente existem muitos LIMS para atender as pesquisas de bioinformática. Viksna e colaboradores [2007] apresentam o sistema PASSIM, LIMS específico para estudos biomédicos envolvendo o corpo humano, desenvolvido em RDBMS. Tolopko e colaboradores [2010] mostram o sistema Screensaver, LIMS específico para pesquisa de genoma na forma de Ácido Ribonucleico (Ribonucleic Acid - RNA). Nieuwland e colaboradores [2012] descrevem o sistema Phytotracker, um LIMS de gerenciamento de laboratório projetado especificamente para organizar plasmídeos de sementes e plantas em crescimento. O Phytotracker foi desenvolvido com RDBMS e modelado através de fluxo de trabalho, mas os conceitos foram embutidos no sistema durante o seu desenvolvimento, o que exige manutenção de *software* para mudança do sistema. Outros exemplos de LIMS específicos e comerciais são o LabVantage Solutions [2015], usado em laboratórios farmacêuticos, alimentícios, químicos e petroquímicos; o Computing Solutions [2015], que possui o escopo de atendimento do LabVantage Solutions acrescido do atendimento aos laboratórios da indústria e da agricultura, e o Solutions [2015].

Embora os LIMS sejam eficazes para as indústrias farmacêuticas e de biotecnologia, a maioria deles são caros e inviáveis para pequenos laboratórios de pesquisa [Li *et al.*, 2006]. Como cada laboratório tem características e tipos de dados diferentes, é necessário que os LIMS sejam adaptados para cada tipo de laboratório, o que exige custos altos de customização para especificação do armazenamento e acesso aos dados. Torna-se necessária a alteração de código-fonte para esses LIMS. Esses sistemas exigem customização de *software* a cada nova implantação em um novo laboratório. Conforme Murray & Crocker [2011], os sistemas de gerenciamento de dados científicos frequentemente não estão disponíveis, adequados e nem acessíveis para projetos com poucos recursos. E as ferramentas comerciais são incapazes de atender as demandas

dos projetos ou muito rígidas para se adaptarem às necessidades. Quando o esquema de dados é frequentemente alterado, a aplicação fica obsoleta até que as novas adaptações sejam aplicadas ao sistema [Goodman *et al.*, 1998]. Trabalhos de pesquisa estão sendo desenvolvidos para aumentar o nível de abstração dos dados dos LIMS, com objetivo de facilitar sua adaptação em diferentes contextos.

Wendl e colaboradores [2007] apresentam um LIMS capaz de generalização de procedimentos. Ele generaliza alguns conceitos, como tipos de aparelho, tipos de DNA (fragmentos brutos de DNA, ligações, subclones, produtos de sequenciamento de DNA, entre outros), tipos de recipiente, bandejas, caixas, prateleiras, mas não possui um alto nível de abstração para apresentação de qualquer conceito no momento de definição de fluxo de trabalho. Esse LIMS possui um conjunto de módulos para representação dos seus conceitos, como módulo de código de barras, de recipientes e de integração com robótica. Para cada conceito, é criado um sub-esquema de tabelas para representação dos seus metadados, como tabela dos subtipos, das instâncias desses tipos, dos seus relacionamentos, entre outros. Dessa forma, o nível de abstração é limitado, visto que a cada novo conceito, o sistema precisa receber customização de *software*.

Stocker e colaboradores [2009] descrevem o iLAP, um LIMS orientado a *workflow*. Os procedimentos do laboratório são organizados em forma hierárquica, o que possibilita a execução de atividades em paralelo, similar ao *SIGLa*. Não há definição das regras do *workflow*, e os conceitos podem ser utilizados de acordo com a necessidade do usuário durante os experimentos. Pode-se criar nova instância de projeto, experimento, anotação, revisão, anexar arquivos, entre outros. Mas como não há uma definição dos procedimentos, o usuário não é guiado com a definição das atividades que precisam ser executadas, nem as informações que devem ser colhidas a cada atividade. A maior parte dos dados não é estruturada e se encontra gravada em textos, sob a forma de descrição de experimentos, anotações e arquivos anexos. Isso não permite consultas avançadas com aplicação de filtros em campos de dados específicos, pois os dados não são estruturados. A pesquisa é feita apenas através de palavras-chave para a busca de experimentos e projetos que contêm os padrões digitados. A implementação desse LIMS trouxe como benefícios a conexão dos arquivos de experimentos diretamente ao processo relacionado; a execução paralela de tarefas, devido ao modelo hierárquico de apresentação dos procedimentos, enquanto as anotações em caderno só permitem a execução sequencial e busca rápida de experimentos e protocolos através de termos-chave. O iLAP apresenta benefícios, mas como não há definição das regras do *workflow*, os usuários podem ficar perdidos. Se os usuários não têm conhecimento sólido sobre o fluxo de trabalho da pesquisa, podem ficar indecisos, pois não são guiados pelo sistema sobre os procedimentos pendentes. Além disso, as informações são pouco estruturadas

o que impede rastreamentos mais avançados de informação como a pesquisa de faixa de valores.

Nelson e colaboradores [2011] apresentam o LabKey Server, um LIMS para integração de dados, análise e pesquisa colaborativa, desenvolvido em RDBMS. É formado por um conjunto de módulos independentes, construídos para realidades específicas, que podem ser integrados. Esse sistema oferece uma flexibilidade de modelo de dados moderada. O laboratório deve usar apenas as entidades pré-definidas no sistema como espécimes, frascos e participantes no módulo de gestão de espécimes. O usuário pode definir quais atributos devem estar presentes em cada entidade em formulários que o sistema disponibiliza para cada entidade. No LabKey Server, o modelo de dados é definido até o nível de entidades e o usuário define quaisquer quantidades e tipos de atributos. Com a necessidade de novas entidades ou alteração de modelagem acima do nível de entidades, é necessária uma manutenção de *software*. O usuário pode aplicar filtros, ordenações e escolhas de campos nas consultas, mas a consulta tem como centro uma tabela específica com seu conjunto de colunas e as colunas das tabelas relacionadas. Não abrange os dados do experimento como um todo em uma mesma consulta. Os resultados de consulta podem ser exportados em formatos de planilha ou texto. Em caso de pesquisas mais complexas, o usuário deve editar o código da consulta.

Muitas tentativas de flexibilização de LIMS são limitadas, como o iLAP e o LabKey Server. Esses sistemas não estão livres de customização de *software* em certas demandas de modificação do ambiente de trabalho de pesquisa. Para aumentar o nível de flexibilidade, existe outro grupo de LIMS, orientados a *workflow*. São exemplos de LIMS orientados a *workflow* o Taverna [Hull *et al.*, 2006], o Kepler [Altintas *et al.*, 2004], o Galaxy [Goecks *et al.*, 2010] e o QTreds [Palla *et al.*, 2012] e o SIGLa. O SIGLa foi escolhido dentre os sistemas orientados a *workflow* para desenvolvimento do sistema de consulta por ser o único genérico.

2.2.3 *Framework* SIGLa

O *framework* SIGLa foi criado em 2010, adaptável para diferentes laboratórios, sem a necessidade de adaptação de código, modelado em RDBMS [Simões *et al.*, 2010]. Esse sistema possui alto grau de usabilidade e possibilita o gerenciamento de dados de diferentes laboratórios de forma flexível. Isso é possível porque todos os procedimentos e atividades do laboratório são modelados em um fluxo de trabalho (*workflow*), externamente ao sistema, editado por uma ferramenta de edição e salvo no formato XPDL. Após a criação do XPDL, basta carregá-lo no SIGLa para que o LIMS específico do laboratório esteja pronto para uso. O arquivo XPDL pode ser criado através de um

editor de *workflows* como o *Together Workflow Editor* (TWE), que edita e gerencia arquivos de definição de processos baseados no WfMC XPDL [Editor, 2015]. Através do *framework* SIGLa foram criados vários LIMS para os mais diversos contextos de pesquisa, como o FluxTransgenics, o NanoTrack, o NMO-DBr, a pesquisa de Proteômica e a pesquisa de Microarranjos de DNA.

Hanke e colaboradores [2014] apresentam o FluxTransgenics, um LIMS criado a partir do *framework* SIGLa para laboratórios de pesquisa de Organismos Geneticamente Modificados (*Genetically Modified Organisms* - GMOs). O FluxTransgenics foi aplicado na Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA) para pesquisa de transformação de milho e sorgo pelo método de *Agrobacterium tumefaciens*. O *workflow* criado representa todas as fases de geração de plantas de milho e sorgo transgênicas. Através do carregamento do *workflow* no sistema SIGLa, o FluxTransgenics foi criado como LIMS completo para o gerenciamento de dados de plantas modificadas. A aplicação desse LIMS garante a qualidade das atividades e produtos no processo de produção de organismos transgênicos e aplica as Boas Práticas de Laboratório (BPL). Os laboratórios de produção transgênica tem poucos LIMS disponíveis e estes frequentemente não se adaptam ao seu fluxo de trabalho, por existirem processos diferentes de transformação genética para cada espécie. Embora existam LIMS que oferecem funções genéricas para transformação, a maioria deles não trata a questão da transformação genética de milho e sorgo. Portanto, O FluxTransgenics surge como um sistema para atendimento eficiente e eficaz sem necessidade de customização adicional de *software*.

Silva e colaboradores [2013] apresentam o NanoTrack, uma ferramenta para apoio a laboratórios de nanociência e nanotecnologia baseada no conceito de LIMS, também criado através de carregamento do *workflow* no SIGLa. Lana-Peixoto e colaboradores [2011] apresentam o Brazilian Neuromyelitis Optica Database System (NMO-DBr), um sistema de banco de dados que coleta, arquiva, recupera e analisa informações de pacientes com neuromielite óptica (NMO). O NMO-DBr contém dados relativos a identificação, sintomas, condições associadas, eventos, índices, recorrências, história familiar, avaliação visual e da medula, incapacidade, exames do líquido e de sangue, IRM, tomografia de coerência óptica (OCT), diagnóstico e tratamento.

Simões e colaboradores [2010] citam mais duas aplicações de uso do SIGLa, uma para pesquisa de Proteômica e outra para pesquisa de microarranjos de DNA. O fluxo de trabalho da pesquisa de proteômica foi desenvolvido com apoio de especialistas de proteômica para aplicação de todos os requisitos do laboratório. A técnica de Microarranjos de DNA é usada para estudar a expressão de genes em diferentes espécies de seres vivos. Essa técnica gera um grande número de informação de dados laboratoriais, imagens e arquivos. O processo de microarranjo tem várias etapas, exigindo o armaze-

namento de informação para cada uma delas. A aplicação do SIGLa nesses laboratórios foi fundamental na melhoria dos seus processos de pesquisa, através da troca de informações armazenadas em cadernos de laboratório, que são de rastreamento difícil, por um gerenciamento de dados em um LIMS completo.

A qualidade dos dados armazenados é importante para evitar consumo de recursos e de tempo em procedimentos de melhoramento das bases de dados. Dados gravados sem a preocupação com a consistência podem gerar necessidades posteriores de melhoria da qualidade de dados ou consolidação de diferentes bases de dados. O pareamento de registros nas bases de saúde do estado de Minas Gerais para eliminação de registros duplicados de pacientes, por exemplo, gerou a necessidade de se fazer uma etapa de limpeza dos dados com eliminação de termos especiais e informações equivocadas, o que gerou perda de informação e custo [Queiroz, 2009]. Chen & Coppé [2012] também exemplificam o custo de limpeza e identificação de dados para consolidação de bases não padronizadas com dados de proteínas e peptídeos.

O SIGLa foi criado e é operado em conformidade com as Boas Práticas de Laboratório (BPL) e implementa aplicações no sistema que validam os campos de entrada, evitando que o usuário insira tipos de dados incorretos, com formato errado, ou esqueça o preenchimento de um campo obrigatório. Dados derivados, como campos que são alimentados por resultados de fórmulas, são automaticamente calculados. O sistema também permite a criação de listagem para campos que têm um domínio bem limitado de valores, como gênero que pode ser *masculino* ou *feminino*.

Embora o SIGLa possua alta flexibilidade no armazenamento de dados, a busca desses dados através de consulta não atende a todos os objetivos dos laboratórios. Os usuários do sistema SIGLa precisam de uma ferramenta mais avançada de pesquisa que consulte qualquer subconjunto de dados do *workflow* e faça o relacionamento de dados entre as entidades modeladas para cada laboratório. E as consultas devem ser feitas *ad hoc*, modeladas pelo próprio usuário a partir das suas necessidades, em vez de consultas pré-modeladas embutidas em código procedimental, para que a flexibilidade do módulo de consultas seja coerente com a flexibilidade do modelo de dados e do armazenamento de informações do SIGLa.

2.2.4 Ferramentas de Consulta

Atualmente existem diversas ferramentas de construção de consultas *ad hoc* para sistemas modelados em RDBMS. Elas são construídas através da análise das restrições de integridade, onde são extraídos os relacionamentos entre as entidades do banco de dados. Mendonça [1999] propõe uma interface gráfica para especificação de consultas

em RDBMS. Os usuários especificam os critérios de busca de dados e o sistema adiciona as junções de forma automática. Yang e colaboradores [2011] apresentam sistema para representação do relacionamento entre um subconjunto de tabelas de um RDBMS, através da criação de caminhos de junção entre as tabelas, mostrando o menor caminho entre elas, pela análise das chaves estrangeiras. Zeng e colaboradores [2012] mostram o sistema *iSearch ranking* para pesquisa de palavras-chave. O sistema transforma o banco de dados relacional em um grafo, onde as tuplas representam os nós e os relacionamentos representam as arestas do grafo. As redes de tuplas são formadas e a ordenação dos resultados é feita a partir da análise dos campos onde as palavras-chave são encontradas. Schulte & Khosravi [2012] tratam a interpretação do RDBMS como aprendizado de máquina para prever ligações entre entidades. As junções são extraídas das dependências relacionais. Buda e colaboradores [2013] usam as restrições de integridade para geração das dependências em RDBMS para garantia de consistência nos sistemas.

As técnicas de uso das restrições de integridade em RDBMS para criação de sistemas de formulação de consultas *ad hoc* são bem conhecidas. Contudo, não são compatíveis com a implementação do SIGLa. Essas técnicas são aplicadas em bancos de dados pré-modelados para um sistema de informação específico. Nesse caso, as entidades criadas no banco de dados correspondem às entidades que representam os conceitos trabalhados na situação em que estão aplicados. E as restrições de integridade do banco de dados representam exatamente os relacionamentos entre as entidades utilizadas por esse contexto. RDBMS são conhecidos pela rigidez do seu modelo de dados [Batra & Tyagi, 2012] e a implementação do SIGLa têm como uma das principais inovações a alta abstração e a flexibilidade para adaptação nos contextos mais diversos, ainda que o mesmo seja implementado em RDBMS. Para obter essa modelagem de dados abstrata, o sistema foi criado em função dos conceitos de *workflow*, como atividades, atributos, registros e instâncias. As entidades do modelo representam, portanto, os conceitos de *workflow* e os relacionamentos do modelo de dados são os relacionamentos entre esses conceitos de *workflow*. A extração das restrições de integridade nesse banco tem com resultados, por exemplo, as junções entre as instâncias do *workflow* e suas atividades e as junções entre as atividades e seus atributos. Como todos os atributos das diferentes atividades são gravados em uma mesma tabela (*attribute*), bem como todas as atividades do *workflow*, na tabela *activity*, o resultado dessas junções são centenas de registros para representação de pouca informação, pois cada linha do resultado apresenta um atributo diferente. Por isso, o resultado é obscuro para qualquer análise de dados, pois não é possível associar, de forma natural, os atributos que pertencem a uma mesma atividade. O resultado adequado para o

usuário é a apresentação dos conceitos finais aplicados ao seu laboratório específico, como *sintoma*, *exame*, *identificação*, *responsável*, *meta* e *projeto* e a apresentação dos atributos que descrevem cada um desses conceitos de forma horizontal, como deve ser apresentado em qualquer outra consulta a sistema de RDBMS com modelo de dados específico.

Os metadados de um banco de dados são informações adicionais que, embora não sejam estritamente parte dos dados registrados, definem a forma como os dados foram gravados [Allan *et al.*, 2012]. Podemos concluir, portanto, que o modelo de dados do SIGLa é orientado aos metadados, conceitos de *workflow* importantes para o funcionamento do SIGLa, e não orientado aos dados, entidades importantes e conhecidas pelo usuário de um laboratório específico onde o SIGLa está implantado. Para criação de um sistema de consulta adequado ao SIGLa, é necessária a aplicação de um mecanismo para pesquisa de dados em *workflows*. Contudo, atualmente foram desenvolvidas consultas de *workflow* apenas por palavras-chave [Cohen-Boulakia & Leser, 2011]. Os usuários pesquisam dados por um conjunto de palavras-chave que são comparadas com os dados do *workflow*, compostos por documentação, metadados, tipos de dados, nomes de tarefas, entre outros. Esse é o único tipo de pesquisa apoiada em sistemas orientados a *workflow* atuais [Oinn *et al.*, 2006]. Wilson [2003] apresenta um sistema de consulta em estruturas de dados hierárquica, similares a estruturas de instâncias de *workflows*, com busca de nós ou de trechos da estrutura a partir de palavras-chave digitadas pelo usuário. Liu e colaboradores [2010] apresentam um sistema de consulta para busca da menor sub-árvore possível do *workflow* que contenha todos os termos pesquisados. Shao e colaboradores [2009] apresentam o sistema WISE para consultas dos nós que possuem as palavras-chave, de forma relacionada, ainda que não estejam conectados de forma direta.

A pesquisa de dados em *workflows* através de palavras-chave é muito limitada, pois não é possível a adição de critérios avançados e consulta à informação estruturada, como a pesquisa de intervalos em valores. Como solução alternativa, muitos sistemas orientados a *workflow* utilizam ontologias de banco de dados para ampliar a capacidade de consulta [Buil-Aranda *et al.*, 2013]. Santoso e colaboradores [2011], por exemplo, utilizam ontologias para extração de todas as relações hierárquicas de um RDBMS, estruturas similares às instâncias de *workflow*. Mas é necessário um pré-processamento para geração da ontologia para posterior consulta aos relacionamentos hierárquicos. As ontologias, conforme descrito anteriormente, podem gerar duplicação da base de dados, necessidade de tempo e mão-de-obra especializada para geração das ontologias, busca em dados obsoleta (consulta à base de dados da ontologia gerada até a última carga de dados) e dificuldade de acesso pelo usuário (as linguagens para consulta de ontologias

são de difícil interpretação pelo usuário).

Los e colaboradores [2005] descrevem o sistema OpenSDE, sistema para a gravação de dados médicos, como queixas, exames físicos e progressos de pacientes em um sistema eletrônico. Os dados são gravados em uma estrutura de árvore e com os campos definidos para que os dados sejam gravados de forma estruturada, com o objetivo de facilitar a busca de dados clínicos, tanto com objetivos de pesquisa quanto com objetivo de cuidados com o paciente. Os formulários para entrada de dados são gerados através de árvores com os conceitos médicos, similar ao SIGLa. Dados clínicos são heterogêneos e a necessidade de mudança dos dados de entrada são frequentes. Para armazenamento desses dados de forma razoavelmente flexível, o sistema expande a metodologia de modelagem de linha tradicional para que usuários possam definir as colunas de cada entidade, para possibilitar a representação estruturada de narrativas médicas. Para flexibilizar a definição dos atributos, o sistema OpenSDE utiliza modelo Entidade-Atributo-Valor (*Entity-Attribute-Value* - EAV), mais indicada para gravação de dados heterogêneos em bancos de dados [Nadkarni, 1998]. O modelo EAV grava um atributo por linha de tabela de atributos, o que permite a gravação flexível de qualquer conjunto de atributos de uma entidade. Esse modelo é o mesmo usado no LabKey Server [Nelson *et al.*, 2011] para flexibilizar a definição de atributos das entidades. Esse modelo tem semelhanças com o modelo de dados do SIGLa, onde cada atributo é gravado em uma linha da tabela de atributos. Mas o nível de abstração do modelo de dados do SIGLa é superior pois permite abstração de toda a definição de estrutura de dados de um laboratório e não apenas a definição dos atributos das entidades. Los e colaboradores [2005] esboçam as possibilidades para extração de dados do OpenSDE e defendem a conversão do modelo de dados EAV para o modelo relacional convencional, o que gera duplicação das bases de dados.

A modelagem feita pelo SIGLa aproveita as características e funcionalidades de RDBMS para restrições de integridade para garantir a qualidade de dados, controle de concorrência, maturidade e suporte a aplicações multiusuário. Ao mesmo tempo, contorna a necessidade da geração do modelo de dados como pré-requisito para desenvolvimento de um LIMS. O SiglaSearch, desenvolvido no projeto desta dissertação, complementa o SIGLa através da disponibilização de um método de consulta para consulta aos dados de *workflow* sem a necessidade de geração de ontologias ou conversão do modelo de dados para bancos de dados convencionais. O SiglaSearch faz as transformações de dados necessárias em tempo de execução das consultas, com tempo de execução razoável e permite a construção *ad hoc* das consultas.

O desenvolvimento do sistema de consulta SiglaSearch não se limitou a formalização dos critérios de consulta e desenvolvimento de uma engenharia de consulta. Foi

desenvolvida também uma interface que procura fornecer boa usabilidade e facilidade de acesso à informação pelo usuário não especializado com uma curva de aprendizado rápida e sem a necessidade de um longo treinamento. Isso estreita a distância entre a tecnologia da informação e os profissionais de outras áreas do conhecimento, como a medicina, a biologia e a física, que já usam o LIMS SIGLa em suas pesquisas.

Brice e colaboradores [2013] apresentam um sistema de consulta em RDBMS e alertam sobre as dificuldades de usuários operarem um sistema dessa categoria. Diferentes usuários têm diferentes necessidades de uso dos dados em um banco de dados. Cada grupo de usuários de um sistema pode ter interesse em diferentes campos de dados, filtros e parâmetros de relatórios. Exibir todas as opções pode representar um grande volume de informação que pode confundir o usuário no momento de geração de suas consultas. Este artigo propõe, portanto, a criação de perfis de usuários que restrinjam a consulta para cada perfil. O SiglaSearch, desenvolvido nessa dissertação, atende a essa necessidade através de definições de relatórios específicos para cada grupo de usuários. Apenas os administradores do sistema SIGLa acessam todas as funcionalidades do SiglaSearch. Demais usuários só precisam executar os relatórios, informando os parâmetros necessários.

Fischer e colaboradores [2011] apresentam uma interface de consulta gráfica em bancos de dados genômicos. A proposta é facilitar o acesso de todos usuários a consultas avançadas. Essa interface implementa as operações de conjunto união, interseção e diferença. As consultas criadas podem ser reaproveitadas por outros usuários. Usuários rodam uma consulta simples e em seguida realizam refinamentos acrescentando ou modificando critérios. Os resultados da consulta são atualizados automaticamente após a inclusão ou alteração de algum passo. Usuários podem adicionar, remover, ordenar e reposicionar colunas de resultados. O universo da consulta desse sistema é restrito à consulta de genes.

Vijayprasath & Rajan [2015] apresenta uma interface para usuários que não são especialistas em tecnologia da informação, para acesso e manipulação de dados em banco de dados relacional Oracle [Loney, 2008]. Esse sistema permite criar tabela, inserir, excluir e alterar registros. A consulta pode ser feita em uma tabela por vez, aplicando os filtros *maior que*, *menor que*, *maior ou igual a*, *menor ou igual a*, *diferente de*, *igual a*, *intervalo* e filtro em lista de valores. Esse sistema de consulta é muito restrito, pois não faz relacionamento entre tabelas, uma das funções essenciais de consultas em RDBMS.

Conforme trabalhos citados neste referencial teórico, a maioria das interfaces de consulta para sistemas de dados biológicos, inclusive na bioinformática, apresenta baixa usabilidade, exigindo que os usuários editem as próprias consultas. Outros sistemas

possuem consultas pré-compiladas, que não admitem mudanças sem customização de *software*. Há ainda outras interfaces muito limitadas, restringindo a consulta a uma única entidade de dados ou com os filtros muito limitados. E existem as interfaces que admitem apenas pesquisas por palavras-chave. Um dos objetivos do SiglaSearch é a construção de interface para consulta em *workflows*, com pesquisa em dados estruturados, aplicação das metodologias de usabilidade e equilíbrio entre expressividade da consulta e facilidade de uso. Os detalhes da interface e as técnicas utilizadas estão descritos no Capítulo 3.

A geração de um novo sistema de consulta que atenda as necessidades do usuário, através da criação de uma engenharia de consulta e de uma interface adequada, torna o LIMs SIGLA um LIMs completo, segundo os pré-requisitos definidos pela literatura.

Capítulo 3

Interface Gráfica do Sistema de Consulta

A interface gráfica do SiglaSearch foi criada para atender as necessidades dos diferentes usuários do SIGLa. Esse sistema possui três partes importantes: a edição de consultas, a geração de relatórios e a administração das consultas. A edição de consultas, que pode ser acessada apenas pelos administradores do sistema, permite que o usuário acesse qualquer conjunto de informações de um *workflow*, com a execução de poucos passos. O sistema disponibiliza um assistente de consulta para obtenção dos primeiros resultados. Em seguida, pode-se usar as outras funções do editor, para refinar a consulta e adicionar, excluir ou alterar itens selecionados, filtros ou critérios de ordenação. As consultas podem ser salvas e abertas posteriormente, para novas alterações. O usuário pode salvar a consulta como um relatório, com os parâmetros adequados, para que esse relatório seja disponibilizado em um menu de relatórios. O menu de relatórios pode ser acessado por usuários especialistas em determinados ramos da pesquisa gerenciada pelo *workflow*. Quando o usuário escolhe um relatório do menu, o sistema exibe a tela de relatório para o usuário inserir os parâmetros, se necessários, e gerar os relatórios. Além da edição de consultas e da execução de relatórios, o SiglaSearch possui o módulo de administração, para gerenciar as consultas e os relatórios disponíveis no SIGLa. A interface do SiglaSearch foi criada observando as teorias da Engenharia Semiótica e demais estudos apresentados no referencial teórico.

3.1 Entrada de Dados no SIGLa

Após os *workflows* serem carregados no SIGLa, eles ficam prontos para serem utilizados pelo usuário. Para cada *workflow* carregado, o sistema exibe as opções

para criar nova instância do *workflow* ou alterar instâncias criadas. Uma instância executada de um *workflow* possui uma estrutura em árvore, onde cada nó da árvore representa uma atividade executada. A Figura 3.1 ilustra um exemplo de tela para entrada de dados de *workflow* no SIGLa. O usuário pode navegar pelas atividades selecionando o respectivo nó da árvore de atividades exibida no lado esquerdo da tela. Para cada atividade o sistema provê um formulário, à direita, com todos os atributos da atividade, para preenchimento e atualização dos dados ou para consulta aos dados gravados, conforme figura.

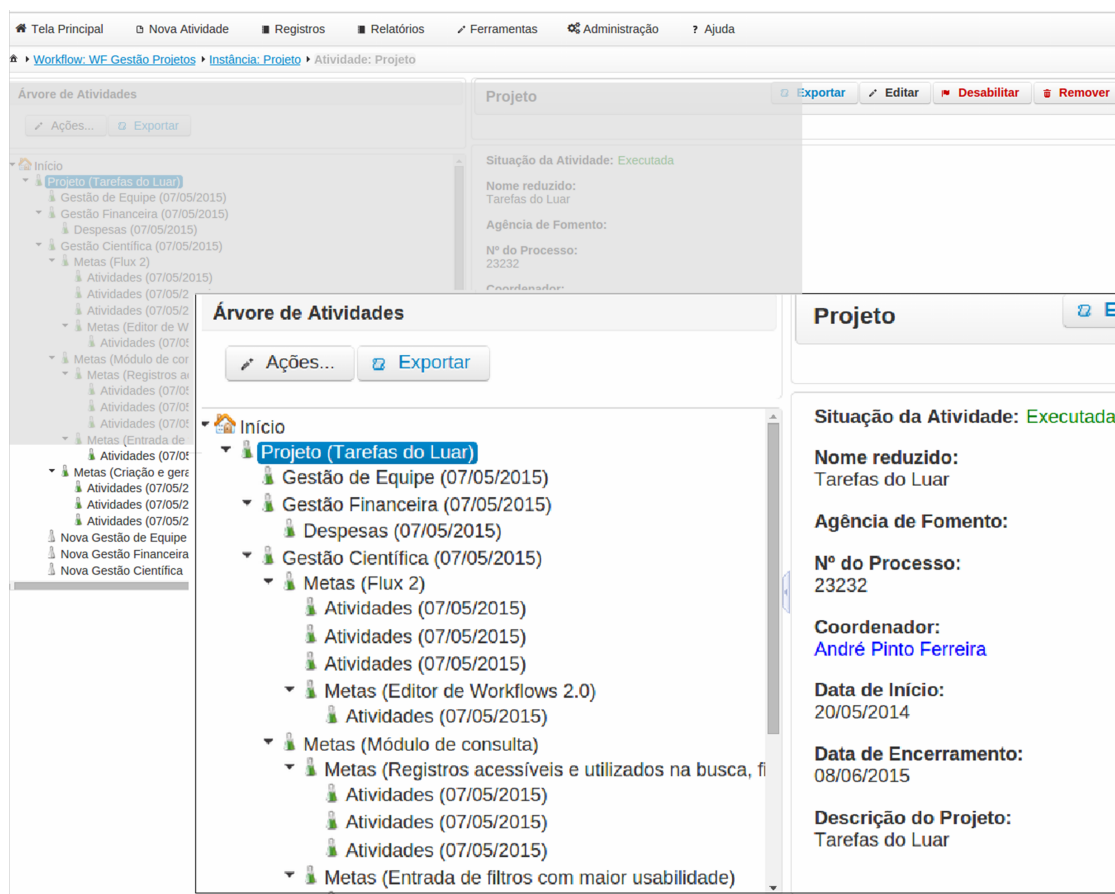


Figura 3.1. Tela de entrada de dados de instância de *workflow* do SIGLa

3.2 Perfis de usuário

Os usuários de informação do SIGLa têm diferentes necessidades de uso do sistema e de trabalho sobre as informações. Para especializar os usuários e diminuir a complexidade do sistema, tais usuários foram classificados em dois perfis: de *administrador do sistema* e de *usuário especializado*. Esses perfis de usuário foram criados para

atribuição de diferentes níveis de visibilidade aos três módulos do SiglaSearch, que são a *edição de consulta*, a *geração de relatórios* e a *administração de consultas*.

3.2.1 Administrador do sistema

O administrador é responsável por manter o correto funcionamento do sistema, carregamento de módulos e controle de acesso aos usuários no SIGLa. No SiglaSearch, ele é responsável pela edição de consultas e pela disponibilização de relatórios, que funcionam como consultas pré-definidas para os usuários especializados. O administrador do sistema não precisa ter conhecimentos aprofundados em computação. Ele pode ser, por exemplo, um chefe de laboratório que recebeu treinamento na ferramenta e trabalha como um facilitador para agilizar a rastreabilidade das informações para demais usuários, sem envolvimento de um especialista em tecnologia da informação.

3.2.2 Usuário especialista

Em uma instituição de pesquisa onde o SIGLa esteja instalado, podem ser realizadas diversas linhas de pesquisa simultâneas. Para cada uma dessas linhas de pesquisa existe um grupo de usuários especialistas. Esses usuários só devem estar focados em sua própria área de atuação. É desejável, portanto, que eles só tenham acesso às ferramentas de relatório do seu interesse. Dessa forma, eles podem recorrer ao administrador do sistema sempre que necessitarem de uma nova ferramenta de relatório. Tal ferramenta pode ser criada em poucos minutos pelo administrador do sistema, através do uso do editor de consultas.

3.3 Edição de Consulta

O editor de consulta é acessado por um menu restrito aos administradores do sistema. Através desse módulo é possível criar, editar e salvar consultas. O sistema permite que o usuário escolha os campos a serem exibidos, que podem ser atributos de atividades ou campos de registros. O usuário também pode fazer sumarização dos dados de algum(s) dos campos selecionados em função dos demais campos. O sistema fornece suporte ao usuário através de documentação, de instruções de introdução ao módulo de consulta, na tela inicial da aplicação, e durante o acesso às funcionalidades, através da exibição de dicas e de alertas quanto a procedimentos incorretos. Esse conjunto de informações facilita o aprendizado rápido sobre o funcionamento do SiglaSearch, além da obtenção ágil dos dados procurados.

3.3.1 Tela principal

Quando se acessa o editor de consulta do SiglaSearch, o sistema exibe as principais instruções para familiarização com a interface em pouco tempo. A Figura 3.2 ilustra a tela principal com as primeiras instruções. Para cada instrução inicial, o usuário pode acessar figuras ilustrativas ao passar o mouse sobre os botões de informação, para que o aprendizado seja mais intuitivo. Para criar nova consulta, o assistente de consulta deve ser executado obrigatoriamente, para obtenção dos primeiros resultados. As instruções iniciais são exibidas até que o assistente de nova consulta seja executado ou até que uma consulta salva seja aberta. Para executar o assistente, basta clicar no botão *Iniciar*.

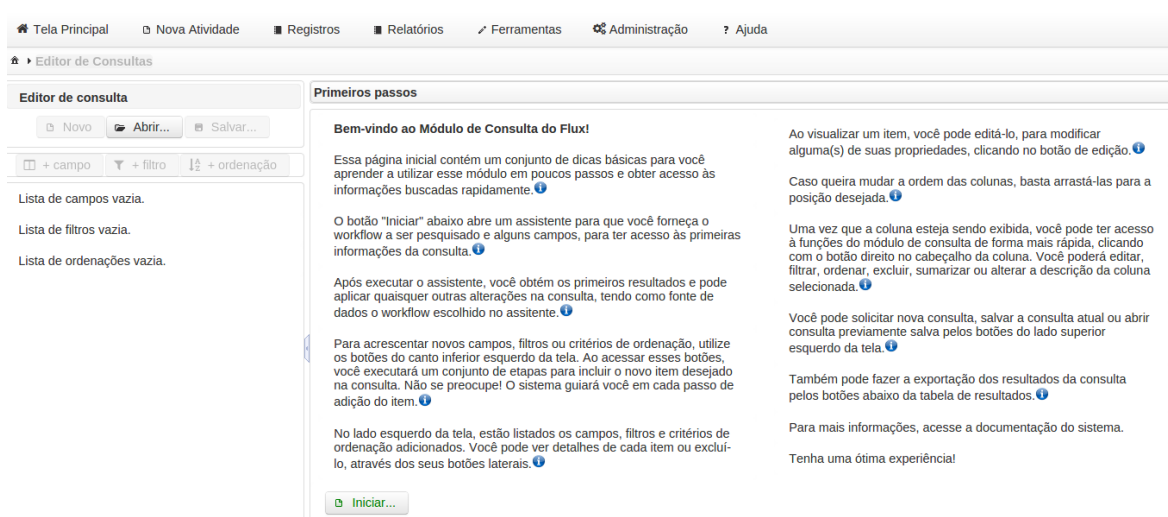


Figura 3.2. Tela principal do editor de consulta do sistema SiglaSearch com as principais instruções

Após a execução do assistente, com os primeiros passos da consulta, ou após abrir uma consulta salva, o sistema passa a exibir a tela com os resultados e habilita os botões com as funções de consulta (Figura 3.3).

A tela principal habilita todos os recursos que estiverem disponíveis para cada cenário de uso. Na região superior esquerda há três botões. O primeiro botão, de rótulo *Novo*, permite criar uma nova consulta, voltando o sistema para situação inicial de execução dos primeiros passos. Quando se pede uma nova consulta com outra consulta em desenvolvimento não salva, o sistema confirma a operação, para prevenir perda de dados. O botão *Abrir* exibe a tela para abrir uma das consultas salvas (Figura 3.4). E o botão *Salvar* abre a tela para salvar a consulta que estiver sendo editada (Figura 3.5). Para disponibilizar a consulta em formato de relatório para todos os usuários do

Consulta: Consulta Andamento Projeto

Meta	Responsável Meta	Perc.	Atividade	Data Conclusão
Criação e geração de Reports concluídas	João Costa Gonçalves	100	Criação do menu dinâmico para Report com os relatórios	24/06/2015
Criação e geração de Reports concluídas	João Costa Gonçalves	100	Criação do formulário dinâmico do Report	24/06/2015
Criação e geração de Reports concluídas	João Costa Gonçalves	100	Criação das funções para exportação dos resultados mod	18/06/2015
Editor de Workflows 2.0	André Pinto Ferreira	70	Parser do XPDL2 Editor	27/06/2015
Entrada de filtros com maior usabilidade	João Costa Gonçalves	100	Limitação dos filtros disponíveis pelo tipo do campo seleci	27/06/2015
Flux 2	André Pinto Ferreira	40	Pendências de registros/auditoria	23/06/2015
Flux 2	André Pinto Ferreira	40	Atributos multi-select, file	30/06/2015
Flux 2	André Pinto Ferreira	40	I18N Preview de Atributos	23/05/2015
Flux 2	André Pinto Ferreira	40	Parser do XPDL2 Editor	27/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Busca de dados de registro	26/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento dos filtros para registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento da ordenação usando registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Limitação dos filtros disponíveis pelo tipo do campo seleci	27/06/2015
Registros acessíveis e utilizados na busca,	Bianca Fernandes Barros	90	Tratamento da ordenação usando registro	25/06/2015
Registros acessíveis e utilizados na busca,	Bianca Fernandes Barros	90	Busca de dados de registro	26/06/2015
Registros acessíveis e utilizados na busca,	Bianca Fernandes Barros	90	Tratamento dos filtros para registro	25/06/2015

Página 1 de 1 16 resultado(s).

Exportar Tudo Exportar Página Atual

Figura 3.3. Tela principal do editor de consultas do sistema SiglaSearch com exibição dos resultados

sistema, basta marcar opção de relatório ao salvar a consulta. Se a consulta não é salva como relatório, fica disponível apenas para os administradores do sistema.

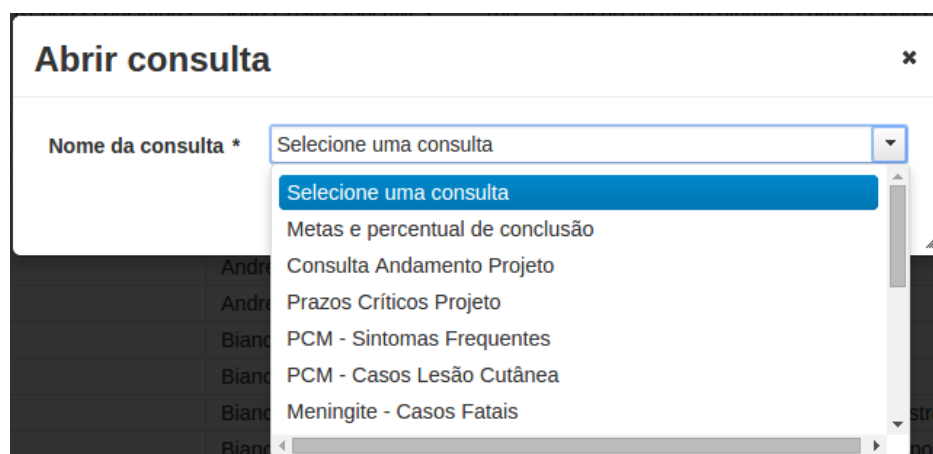
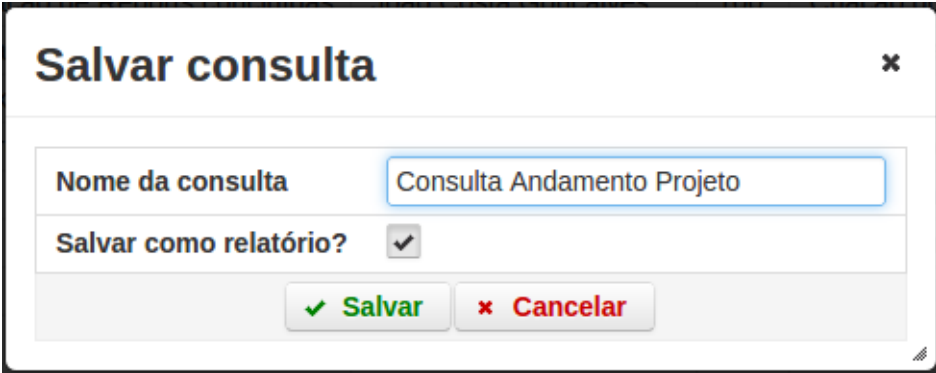


Figura 3.4. Tela para abrir consulta salva no sistema SiglaSearch

Na região superior esquerda da tela, logo abaixo dos botões *Novo*, *Abrir* e *Salvar*, há outros três botões. O primeiro, de rótulo *+campo*, permite que o usuário adicione um novo campo de dados na consulta. O segundo botão, de rótulo *+filtro*, permite que o usuário inclua novo filtro comparando um campo da consulta com um valor fixo ou com outro campo. O terceiro botão, de rótulo *+ordenação*, permite que o usuário inclua novo critério de ordenação, com opção crescente ou decrescente. Decidiu-se



The image shows a dialog box titled "Salvar consulta" with a close button (X) in the top right corner. Inside the dialog, there is a text input field labeled "Nome da consulta" containing the text "Consulta Andamento Projeto". Below this is a checkbox labeled "Salvar como relatório?" which is checked. At the bottom of the dialog, there are two buttons: "Salvar" with a green checkmark icon and "Cancelar" with a red X icon.

Figura 3.5. Tela para salvar consulta no sistema SiglaSearch

por criar a opção de ordenação para que pudessem ser criados mais de um critério de ordenação aplicados à mesma consulta e também para que tais critérios fossem automaticamente aplicados aos relatórios. No lado esquerdo do editor são exibidos todos os itens adicionados à consulta, ou seja, os campos selecionados, os filtros e os critérios de ordenação. Na região central da tela, são mostrados os resultados gerados para a consulta, atualizados a cada adição ou alteração de item. O usuário pode exportar os dados em arquivos de formato Excel, CSV, PDF ou XML, através dos botões exibidos abaixo da tabela de resultados.

3.3.2 Criar nova consulta

Quando o menu do editor de consulta é acessado, o sistema é iniciado em modo de criação de nova consulta, exibindo as instruções iniciais. Durante o uso do editor, basta clicar no botão *Novo* para iniciar nova consulta. Em uma nova consulta, o assistente deve ser executado. Ele é acessado através do botão *Iniciar*. Nesse assistente, o usuário define o *workflow* a ser consultado e os primeiros atributos que desejar consultar. A escolha de *workflow* em uma consulta é feita uma única vez, pois toda a consulta será feita em um mesmo *workflow*. Após a escolha do *workflow*, o usuário pode escolher sucessivas atividades e respectivos atributos a consultar. O fluxo de uso do assistente está esquematizado na Figura 3.6.

As telas que compõem o assistente estão ilustradas na Figura 3.7. O *workflow* é selecionado na tela da Figura 3.7(a). Essa tela é exibida apenas uma vez a cada nova consulta. A atividade é escolhida na tela da Figura 3.7(b). E a tela da Figura 3.7(c) mostra a lista de campos a serem selecionados, que pertençam à atividade selecionada pelo usuário no passo de escolha da atividade. Nessa última tela, o usuário pode marcar vários atributos da atividade escolhida. Após a escolha dos campos da atividade

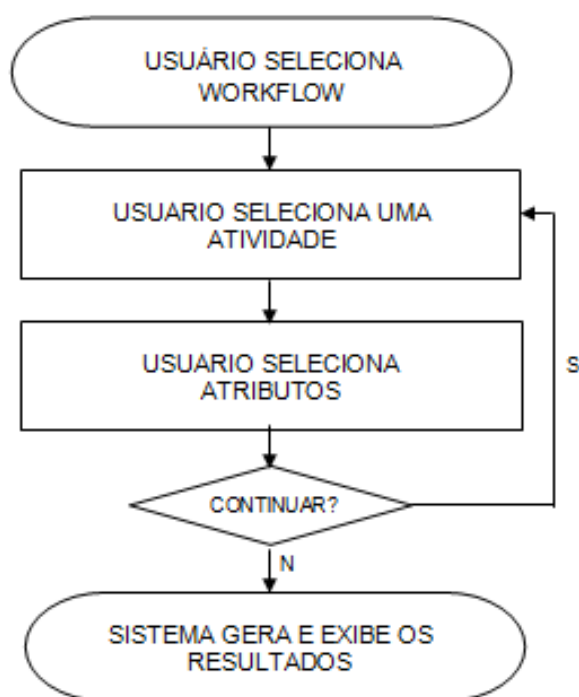
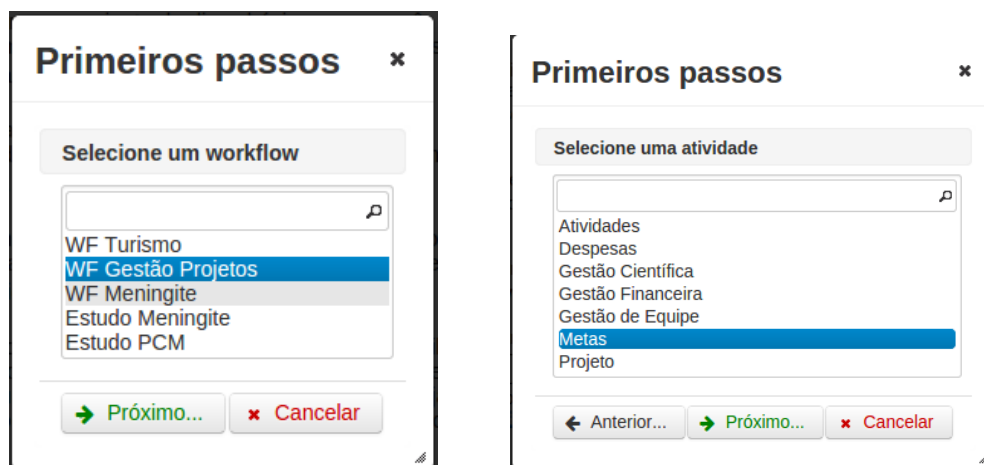


Figura 3.6. Fluxo de uso do assistente de consulta

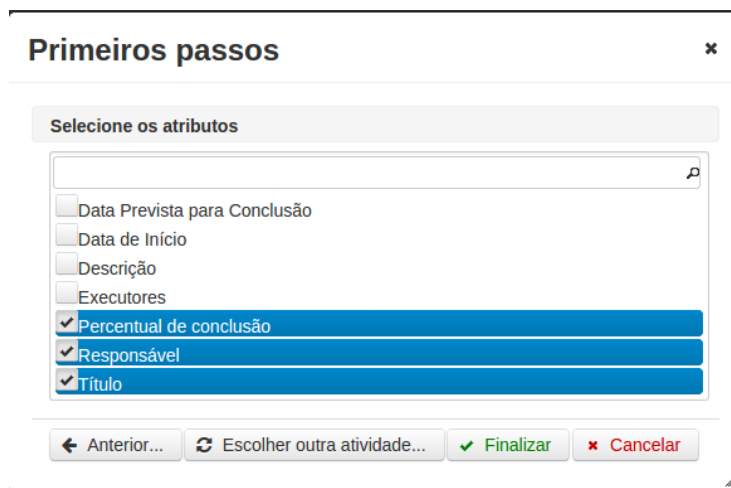
selecionada, pode-se continuar adicionando campos, clicando no botão *Escolher outra atividade* ou finalizar o assistente para visualizar os primeiros resultados, através do botão *Finalizar*. Ao finalizar o assistente, o sistema exibe os resultados da consulta na região da tela anteriormente usada para mostrar as dicas iniciais. O assistente é executado uma única vez a cada consulta. Para refinar os critérios da consulta, deve-se usar os recursos disponíveis na tela, para adicionar outros campos, filtros, critérios de ordenação, além de alterações e exclusões dos itens já incluídos.

3.3.3 Campo de Dados da Consulta

Para entender o refinamento da consulta, é necessário entender o conceito de campo no SiglaSearch. O conceito de *campo* é utilizado para descrever o componente fundamental de um item da consulta. Um *item de consulta* pode ser de três tipos: um campo selecionado, um filtro ou um critério de ordenação. E um *campo de consulta* também possui três classificações: um campo de banco de dados, um campo de sumarização e uma constante. Um *campo de banco de dados* representa todo campo concreto, extraído diretamente do banco de dados, sendo originado de um atributo de atividade ou de uma coluna de registro. Um *campo de sumarização* representa a sumarização de dados, através de uma função de agregação aplicada a um campo de banco de dados. Uma *constante* serve para armazenar qualquer valor fixo, lista de valores ou intervalo

(a) Escolha do *workflow*.

(b) Escolha da atividade.



(c) Escolha dos atributos da atividade selecionada.

Figura 3.7. Telas de execução do assistente de consulta do sistema SiglaSearch para os primeiros passos da edição de uma consulta

informado pelo usuário, como um CPF, um intervalo de datas ou uma lista de nomes. A lista completa de tipos de atributos disponibilizados pelo SIGLa é mostrada na Tabela 3.1. Um campo de sumarização possui como parâmetro um campo de banco de dados. Os itens de consulta também são parametrizados pelos campos de consulta. É possível selecionar um campo de banco de dados ou campo de sumarização. A ordenação também pode ser aplicada em um desses dois tipos de campos. O filtro possui dois campos da consulta e um operador como parâmetros. Pode-se criar filtro entre dois campos de banco de dados, entre dois campos de sumarização, entre um campo de banco de dados e um campo de sumarização ou entre um campo de banco de dados ou campo de sumarização e uma constante. Ou seja, é possível qualquer combinação de classificação de campo em um filtro, exceto a combinação de constante com constante.

Classificação	Atributos
Básicos	boolean, date, datetime, float, integer, string, time.
Declarados	barcode, button_many, button_one, CEP, checkbox, CNPJ, CPF, currency, dropdown, email, file, information, menu_many, menu_one, picture, radio, register, select, text, textbox.

Tabela 3.1. Tipos de atributos do SIGLa

A fonte de dados de um campo de consulta pode ser um atributo de uma atividade ou uma coluna de dados de um registro. Cada atributo de uma atividade possui um tipo, como *integer*, *CPF*, *register* e *select*, por exemplo. Quando o atributo é do tipo *register* ou *select*, significa que seu valor é representando por uma tupla de registro ou por um conjunto de tuplas de um registro respectivamente. Nesse caso, pode-se navegar em um novo nível de detalhe, para escolher uma coluna específica do registro ou deixar como padrão a coluna que nomeia o registro, definida pelo administrador no momento de carga da definição do registro no sistema.

O tipo de dado do campo de consulta é importante para definir compatibilidade com os filtros e com as funções de sumarização. Não podemos, por exemplo, fazer a média de nomes de pacientes, pois se trata de um campo de texto. Mas podemos fazer a média de um campo data ou o somatório de despesas. O sistema também verifica a compatibilidade dos campos da consulta com os filtros. Não podemos comparar data com valor inteiro, por exemplo, ou aplicar filtro de intervalo em campo de nome. O sistema se responsabiliza por verificar os campos disponíveis para cada filtro e para cada função de sumarização e a compatibilidade entre campos na aplicação dos filtros, para diminuir as responsabilidades do usuário e impedir erros que frustrem sua experiência com o sistema.

3.3.4 Adição de campo selecionado

Para adicionar um campo selecionado à consulta após a execução do assistente, basta clicar no botão *+ campo*. O sistema guia o usuário em um conjunto de passos, com apresentação de dicas nos passos menos intuitivos. O fluxograma de passos para adição do campo está ilustrado na Figura 3.8. O usuário deve escolher atividade e atributo. Caso o atributo seja do tipo *register* ou *select*, o sistema apresenta a escolha do campo de registro como próximo passo. O usuário pode apenas manter a opção padrão, para usar nome do registro, ou escolher outra coluna. O próximo passo é a escolha de

função de sumarização, onde o usuário pode escolher uma função de sumarização ou apenas manter a opção padrão de não aplicar função. Se o usuário optar pela aplicação de função, adicionará um campo de sumarização. Se não aplicar função, definirá um campo de banco de dados, oriundo de atributo de atividade ou de campo de registro, que é parte de atributo de tipo *register* ou *select* de uma atividade. O último passo consiste em informar uma descrição para o campo, se o usuário não quiser utilizar a opção padrão de nome do atributo.

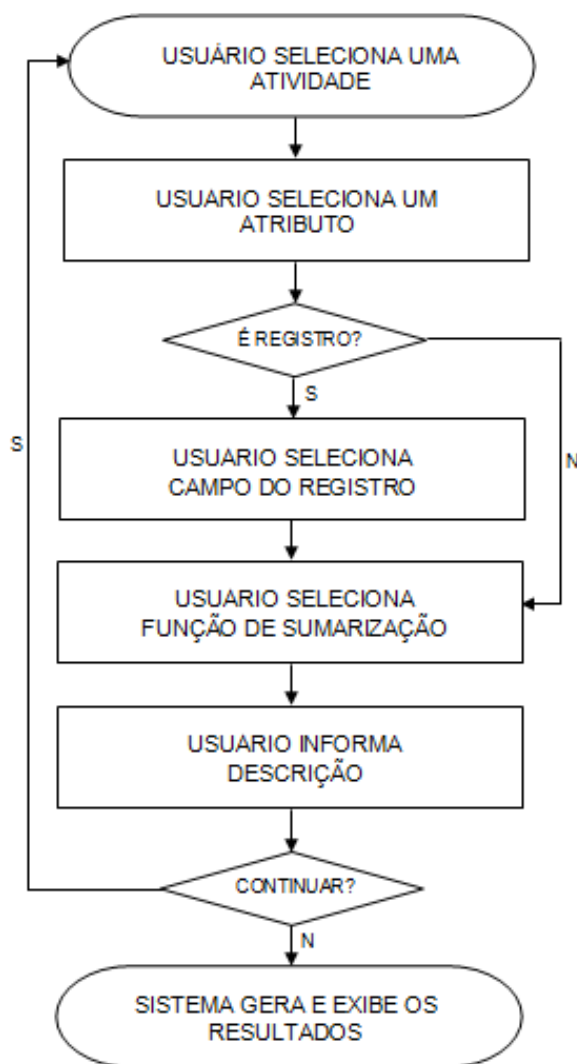
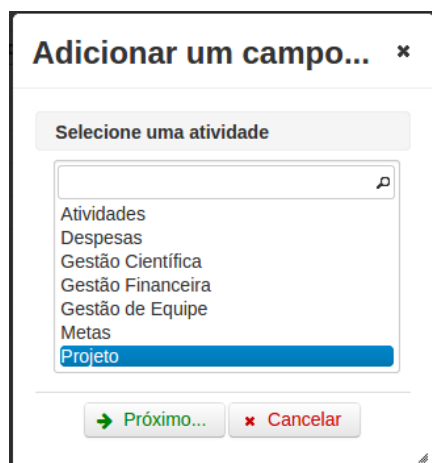


Figura 3.8. Fluxo de passos para adição de novo campo à consulta do sistema SiglaSearch

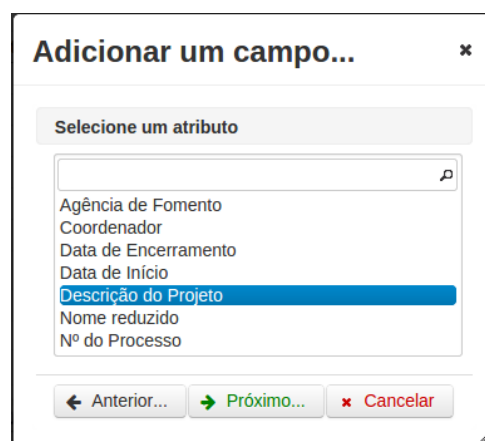
A Figura 3.9 mostra as telas para cada passo de adição de campo. O primeiro passo é a escolha da atividade (Figura 3.9(a)). O segundo passo é a escolha do atributo (Figura 3.9(b)). Caso o atributo seja do tipo *register* ou *select*, o sistema apresenta o passo da Figura 3.10. Caso contrário, mostra a tela para escolha da função de

sumarização (Figura 3.9(c)). Quando a adição de campo envolve registro, o passo de escolha de função é apresentado logo após a tela de escolha de coluna de registro. O último passo de adição de campo está ilustrado na Figura 3.9(d) e é utilizado para informar uma descrição do campo diferente do nome padrão.



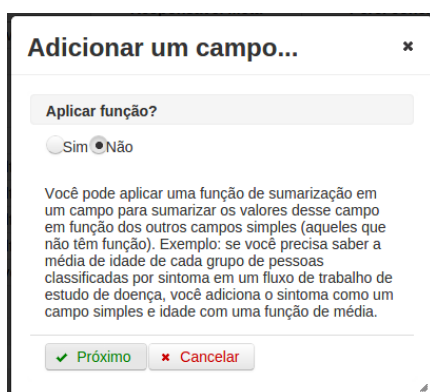
The screenshot shows a dialog box titled "Adicionar um campo...". It has a search bar and a list of activities: "Atividades", "Despesas", "Gestão Científica", "Gestão Financeira", "Gestão de Equipe", "Metas", and "Projeto". The "Projeto" option is highlighted in blue. At the bottom, there are two buttons: "Próximo..." with a green arrow and "Cancelar" with a red 'x'.

(a) Escolha da atividade.



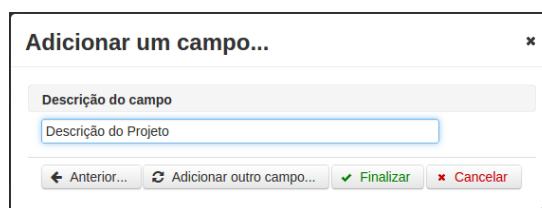
The screenshot shows a dialog box titled "Adicionar um campo...". It has a search bar and a list of attributes: "Agência de Fomento", "Coordenador", "Data de Encerramento", "Data de Início", "Descrição do Projeto", "Nome reduzido", and "Nº do Processo". The "Descrição do Projeto" option is highlighted in blue. At the bottom, there are three buttons: "Anterior..." with a left arrow, "Próximo..." with a green arrow, and "Cancelar" with a red 'x'.

(b) Escolha do atributo.



The screenshot shows a dialog box titled "Adicionar um campo...". It has a section "Aplicar função?" with two radio buttons: "Sim" and "Não". The "Não" option is selected. Below this is a paragraph of text explaining the summarization function. At the bottom, there are two buttons: "Próximo" with a green checkmark and "Cancelar" with a red 'x'.

(c) Escolha de função de sumarização.



The screenshot shows a dialog box titled "Adicionar um campo...". It has a section "Descrição do campo" with a text input field containing "Descrição do Projeto". At the bottom, there are four buttons: "Anterior..." with a left arrow, "Adicionar outro campo..." with a circular arrow, "Finalizar" with a green checkmark, and "Cancelar" with a red 'x'.

(d) Descrição do campo selecionado.

Figura 3.9. Passos para seleção de campo de uma consulta do sistema SiglaSearch

O procedimento para adicionar um campo de sumarização é similar ao de adição de um campo de dados. Basta que se escolha uma função de sumarização no passo de adição de função. Como exemplos de campos de sumarização pode-se citar a média de temperatura corporal por diagnóstico de uma doença ou a data máxima de conclusão das atividades para cada meta e para cada responsável pela execução. A função de sumarização é aplicada em todos os valores encontrados para cada item do conjunto de registros formado pelos campos de bancos de dados (não sumarizados). Se forem selecionados apenas campos de sumarização, as funções de sumarização serão aplica-

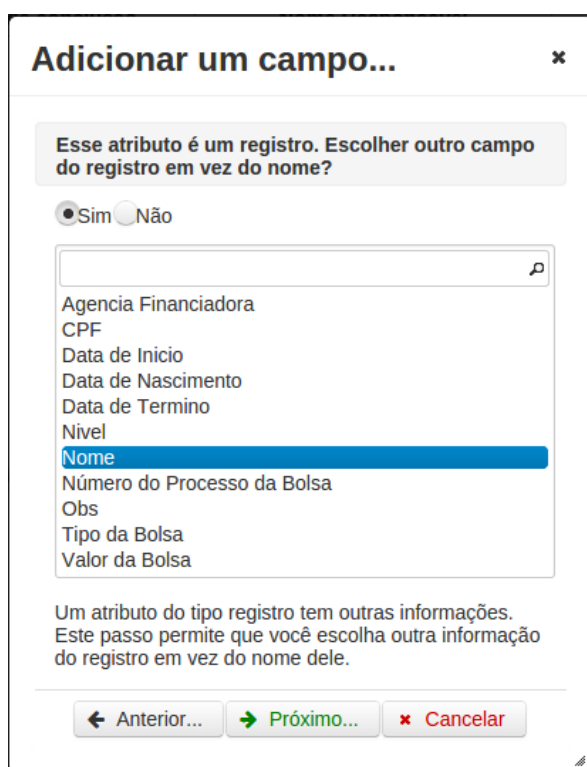


Figura 3.10. Escolha de outra coluna do registro do sistema SiglaSearch

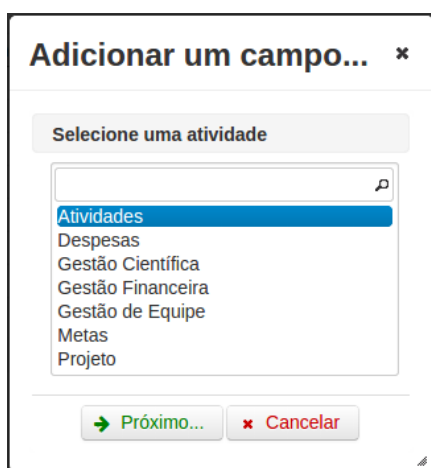
das em todo o banco de dados, retornando uma única tupla de resultados. Funções de sumarização são aplicadas principalmente em relatórios estatísticos, para análise resumida dos dados da aplicação. Os tipos de dados compatíveis para cada função de sumarização estão especificados na Tabela 3.2.

Função	Tipos Compatíveis
Contagem	Todos
Somatório	Valores numéricos
Máximo	Valores numéricos e datas
Mínimo	Valores numéricos e datas
Média	Valores numéricos e datas

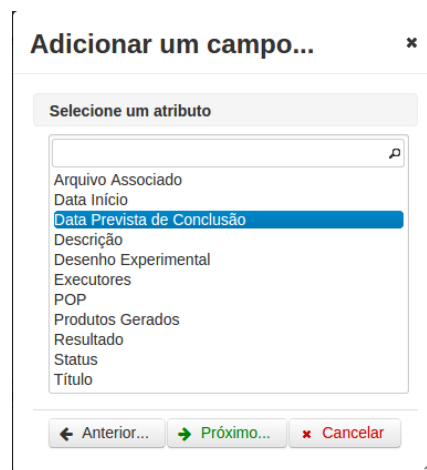
Tabela 3.2. Funções de sumarização e tipos compatíveis do sistema SiglaSearch

A Figura 3.11 exemplifica os passos para adição de um campo de sumarização. Esse é um exemplo de busca da data máxima de conclusão das atividades em função dos demais campos selecionados no *workflow* de Gestão de Projetos. Se, além desse campo de sumarização, selecionarmos o campo de responsável, o resultado da consulta será a data máxima de conclusão das atividades para cada responsável. Os resultados dessa consulta mostram quais responsáveis estão com prazos mais apertados para a

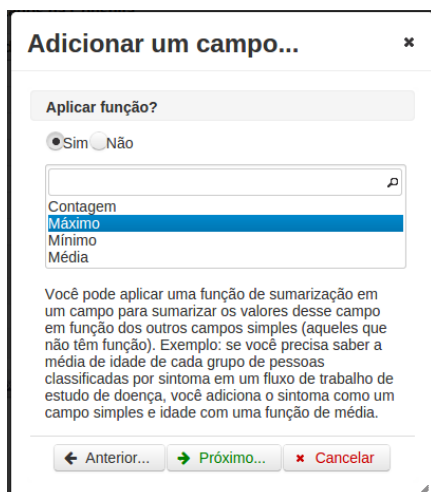
conclusão do projeto, uma função importante para o gerenciamento de projetos. A Figura 3.11(a) mostra a escolha da atividade *Atividades*, a Figura 3.11(b) mostra a escolha do atributo *Data prevista de conclusão* e a Figura 3.11(c) mostra a escolha da função *Máximo* para busca da data máxima de conclusão de atividade, o que representa a data de conclusão de todas as atividades, para cada responsável selecionado. A Figura 3.11(d) ilustra a tela de edição do rótulo do campo de sumarização.



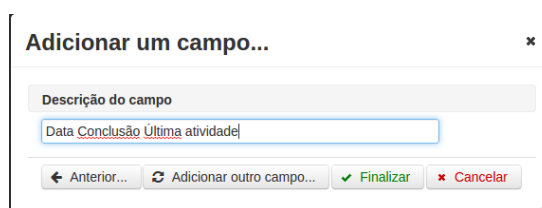
(a) Escolha da atividade.



(b) Escolha do atributo.



(c) Escolha da função de sumarização.



(d) Entrada da descrição do campo de sumarização.

Figura 3.11. Passos para adição de campo de sumarização de dados do sistema SiglaSearch

Para alterar a ordem de exibição dos campos selecionados, basta arrastar a coluna da tabela, conforme ilustrado na Figura 3.12.

Responsável Meta	Meta
Bianca Fernandes Barros	Criação e geração de Report

Figura 3.12. Para modificar a ordem das colunas de resultado do sistema SiglaSearch, basta arrastar as colunas com o *mouse*

3.3.5 Adicionar critério de ordenação

Para adicionar um critério de ordenação à consulta, basta clicar no botão *+ordenação*. O sistema guia o usuário em cada passo. O procedimento para adição de critério de ordenação é bem semelhante ao de escolha de campo. A diferença está no último passo. Não se define descrição de critério de ordenação, mas deve-se definir a opção de ordenação, como crescente ou decrescente, através da tela ilustrada pela Figura 3.13.

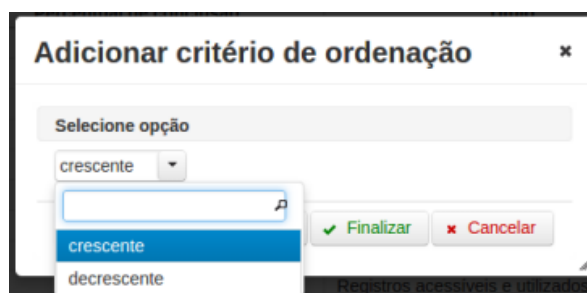


Figura 3.13. Escolha da opção de ordenação do campo como crescente ou decrescente no sistema SiglaSearch

3.3.6 Adicionar filtro

Para facilitar o entendimento dos filtros do SiglaSearch, é conveniente fazer uma simplificação dos conceitos de campo de consulta. Conforme seção anterior, um campo de consulta admite as classificações de campo de banco de dados, campo de sumarização e uma constante. Sabemos que tanto o campo de banco de dados quanto o campo de sumarização é extraído do banco de dados do SIGLa. A diferença é que o campo de banco de dados é uma informação concreta e o campo de sumarização é uma informação derivada. Nesta seção sobre filtros de consulta, os campos de banco de dados e os campos de sumarização serão denominados campos de dados do *workflow* ou apenas *campos*. E o campo de constante será denominado aqui como apenas *constante*. As nuances entre escolha de campo de banco de dados e de campo de sumarização são as

mesmas tratadas na seção sobre seleção de campo, portanto não serão tratadas nesta seção.

Para adicionar um filtro à consulta, basta clicar no botão *+filtro*. O fluxograma da Figura 3.14 esquematiza os passos para a adição de um novo filtro. Após a escolha do primeiro campo, deve-se fazer a escolha do operador do filtro. O sistema exibe apenas os operadores compatíveis com o campo escolhido. Um campo de data, por exemplo, aceita o operador *maior* ($>$), mas não aceita o operador de comparação de textos. Alguns filtros aceitam operação entre dois campos, outros admitem apenas a operação de campo com constante. O filtro de igualdade, por exemplo, aceita a comparação de campo com campo, mas o filtro de intervalo só opera entre campo e constante. A Tabela 3.3 mostra os tipos compatíveis com cada tipo de filtro e mostra quais filtros aceitam outro campo, para operação de campo com campo. Caso se escolha um filtro que aceite comparação com outro campo, o sistema solicita a escolha da opção para o segundo operador, para usar outro campo ou uma constante. Se o filtro não aceitar outro campo como opção, o sistema passa diretamente para a escolha da constante. Caso o usuário escolha outro campo como opção, o sistema exibe os passos para a escolha do segundo campo, exibindo apenas os campos compatíveis nas listas de opções. Se o usuário escolher filtrar campo com constante ou o filtro admitir apenas constante, o sistema mostra a tela para a entrada da constante com dicas customizadas para cada tipo de filtro e de campo. Em seguida, basta entrar com a descrição do filtro. O usuário pode encerrar a operação de inclusão de filtro ou solicitar outro filtro no botão *Adicionar outro filtro...* dessa mesma tela.

Filtro	Tipos de dados compatíveis	Aceita dois campos?
Igual	Todos	Sim
Diferente	Todos	Sim
Maior	Valores numéricos e datas	Sim
Menor	Valores numéricos e datas	Sim
Maior ou igual	Valores numéricos e datas	Sim
Menor ou igual	Valores numéricos e datas	Sim
Intervalo	Valores numéricos e datas	Não
Como	Cadeias de caracteres	Não
Não é como	Cadeias de caracteres	Não

Tabela 3.3. Especificação dos filtros de consulta do sistema SiglaSearch

As telas específicas para adição de filtro estão ilustradas na Figura 3.15. Os passos para seleção de campo não são mostrados aqui, pois são os mesmos aplicados na seleção de campo. A Figura 3.15(a) ilustra a escolha do filtro. A Figura 3.15(b) mostra a tela para escolha de opção para o segundo operador. A Figura 3.15(c)

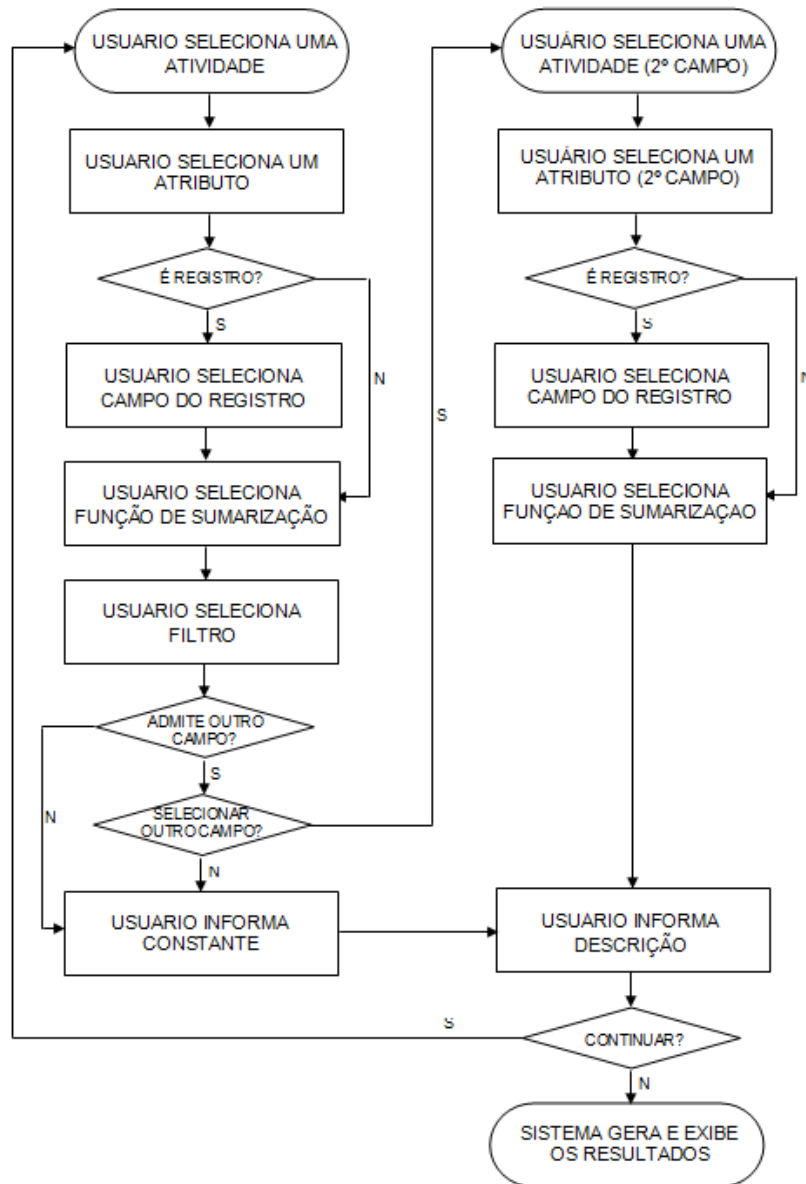


Figura 3.14. Fluxo de passos para adição de filtro à consulta do sistema Sigla-Search

mostra a tela de definição de constante para o segundo operador, que é customizada para orientar o usuário a inserir constante compatível com o primeiro operador e com o filtro escolhido. Essa tela também exibe a opção para que a constante seja modificada durante a execução do relatório. Essa opção é aplicada quando a consulta editada é salva como relatório para ser disponibilizada aos usuários de áreas especializadas da pesquisa. Por último, deve-se inserir uma descrição para o filtro na tela ilustrada pela Figura 3.15(d), para que ele seja entendido de forma clara durante a execução do relatório. O filtro editado na consulta se tornará um parâmetro para o relatório, se a constante puder ser modificada e se a consulta for salva como relatório.

(a) Escolha do filtro.

(b) Escolha da opção para o segundo operador do filtro.

(c) Definição da constante para o segundo operador.

(d) Descrição do filtro.

Figura 3.15. Telas dos passos específicos para adição de filtro no sistema Sigla-Search

3.3.7 Alterar e excluir itens da consulta

Todos os itens adicionados à consulta podem ser vistos no lado esquerdo da tela de edição. Para acessar detalhes de cada item, basta clicar no botão de visualização à

esquerda do item desejado. Um item de consulta pode ser um campo selecionado, um filtro ou um critério de ordenação. A Figura 3.16 mostra a visualização de um campo selecionado. Há duas telas de visualização de filtros, uma para visualização de filtro entre dois campos (Figura 3.17) e outra para filtro entre campo e constante (Figura 3.18). A Figura 3.19 mostra a visualização de critério de ordenação. Para editar o item, basta clicar no botão *Editar* da mesma tela. Para excluir o item da consulta, basta clicar no botão de exclusão à esquerda do respectivo item.

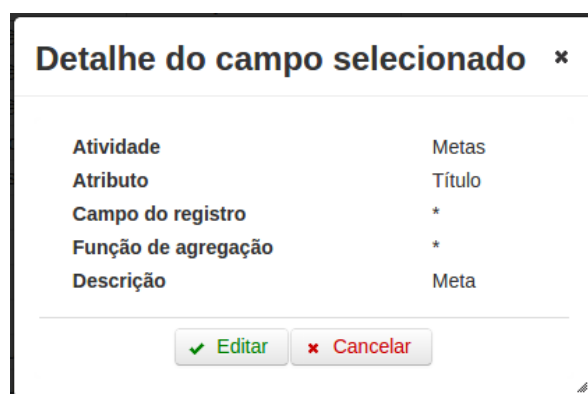


Figura 3.16. Tela de visualização dos detalhes de um campo selecionado



Figura 3.17. Tela de visualização dos detalhes de um filtro entre dois campos

Detalhe do filtro ✕

Primeiro operador	Atividade Atividades Atributo Data Prevista de Conclusão Campo do registro * Função de agregação *	
Operação	Intervalo (Intervalo de Primeiro operador: Segundo operador)	
Segundo operador	Parâmetro 01/01/2015-01/07/2015 Esse valor poderá ser modificado na geração de relatório? <input checked="" type="checkbox"/>	
Descrição	Intervalo de Data Prevista de Conclusão: ?	

✓ Editar
✕ Fechar

Figura 3.18. Tela de visualização dos detalhes de um filtro entre um campo e uma constante

Detalhe de ordenação ✕

Atividade	Metas
Atributo	Título
Campo do registro	*
Função de agregação	*
Opção	crecente

✓ Editar
✕ Cancelar

Figura 3.19. Visualização de critério de ordenação no sistema SiglaSearch

3.3.8 Caminhos alternativos de funcionalidades

Para facilitar o acesso ao sistema, foram criados caminhos alternativos para determinadas funcionalidades. Para cada coluna selecionada e mostrada na tabela de resultados é exibido um menu de contexto, ilustrado na Figura 3.20. O menu *Editar* tem a mesma função do botão de edição da tela de visualização dos detalhes dos itens. O menu *Excluir* tem a mesma função do botão de exclusão exibido no lado esquerdo da tela para o respectivo item. Os menus *Filtrar* e *Ordenar* são atalhos para adição de filtro e critério de ordenação, respectivamente, utilizando como parâmetro a coluna selecionada. O menu *Sumarizar* permite transformar um campo de dados em campo de sumarização, útil para a construção progressiva de relatórios sumarizados. E o menu *Alterar Descrição* é um atalho para edição da coluna, para alterar apenas a descrição do campo, o que é uma das ações mais frequentes de alteração de item.

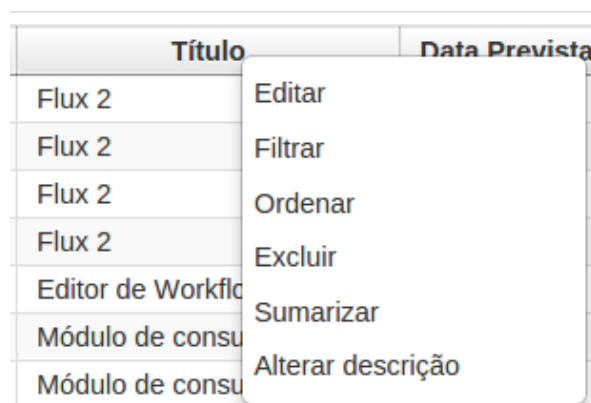


Figura 3.20. Menu de contexto para colunas selecionadas no sistema SiglaSearch

3.4 Consulta de Relatórios

Todas as consultas salvas como relatório são disponibilizadas para os usuários do sistema no menu específico de relatórios (Figura 3.21). Ao salvar a consulta como relatório, este é automaticamente exibido como opção no menu de relatórios.

Quando um relatório é acessado, o sistema exibe uma tela inicial com as principais instruções para uso do mesmo (Figura 3.22). Cada botão de informação exibe figura ilustrativa para facilitar o entendimento da instrução. Essas instruções iniciais são exibidas até que o usuário execute o relatório pela primeira vez.

Os usuários devem informar os valores para os parâmetros dos relatórios, exibidos no formulário à esquerda da tela. Também podem ser utilizados os valores padrão,



Figura 3.21. Menu de relatórios disponíveis do sistema SiglaSearch

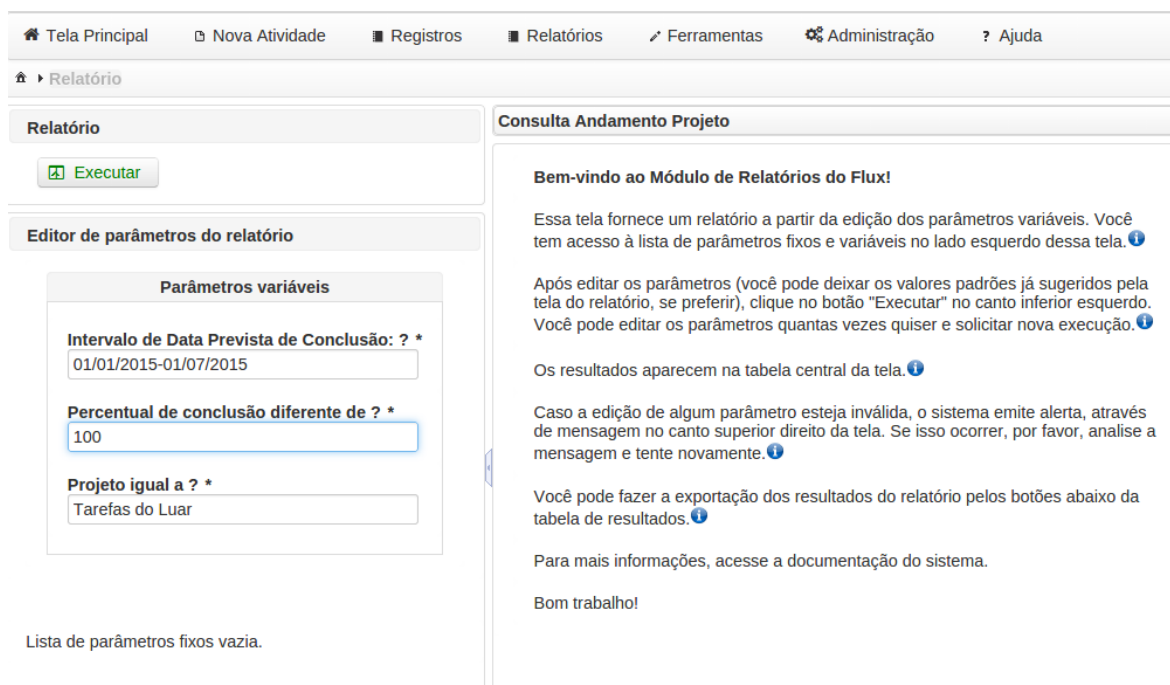


Figura 3.22. Tela inicial de Relatório do sistema SiglaSearch

definidos na edição da consulta. O usuário pode executar o relatório quantas vezes desejar, clicando no botão *Executar* para atualizar os resultados. A Figura 3.23 ilustra uma tela de relatório com exibição dos resultados. Os resultados dos relatórios também podem ser exportados em arquivos de formato Excel, CSV, PDF ou XML, através dos botões exibidos abaixo da tabela de resultados.

The screenshot shows the 'Relatório' interface. On the left, the 'Editor de parâmetros do relatório' contains the following parameters:

- Parâmetros variáveis**
- Intervalo de Data Prevista de Conclusão: ? *
01/01/2015-01/07/2015
- Percentual de conclusão diferente de ? *
100
- Projeto igual a ? *
Tarefas do Luar

The main table, 'Consulta Andamento Projeto', displays the following data:

Meta	Responsável Meta	Per	Atividade	Data Conclusão
Editor de Workflows 2.0	André Pinto Ferreira	70	Parser do XPDL2 Editor	27/06/2015
Flux 2	André Pinto Ferreira	40	Pendências de registros/auditoria	23/06/2015
Flux 2	André Pinto Ferreira	40	Atributos multi-select, file	30/06/2015
Flux 2	André Pinto Ferreira	40	I18N Preview de Atributos	23/05/2015
Flux 2	André Pinto Ferreira	40	Parser do XPDL2 Editor	27/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento dos filtros para registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento da ordenação usando registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Limitação dos filtros disponíveis pelo tipo do campo	27/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Busca de dados de registro	26/06/2015
Registros acessíveis e utilizados na busca	Bianca Fernandes Barros	90	Busca de dados de registro	26/06/2015
Registros acessíveis e utilizados na busca	Bianca Fernandes Barros	90	Tratamento dos filtros para registro	25/06/2015
Registros acessíveis e utilizados na busca	Bianca Fernandes Barros	90	Tratamento da ordenação usando registro	25/06/2015

Below the table, it indicates 'Página 1 de 1' and '12 resultado(s)'. At the bottom, there are buttons for 'Exportar Tudo' and 'Exportar Página Atual', along with icons for exporting to PDF, CSV, and XML.

Figura 3.23. Tela de relatório com exibição dos resultados do sistema SiglaSearch

3.5 Administração de Consultas

O administrador do sistema pode gerenciar as consultas e relatórios salvos através da tela de administração das consultas (Figura 3.24). Através dessa tela, é possível alterar a descrição de consultas e relatórios, disponibilizar relatório ou suspender exibição, através da mudança do indicador de relatório, e excluir consultas e relatórios. O sistema também exibe um atalho para acesso à tela de edição da consulta, através dos botões de edição, para cada consulta exibida.

3.6 Discussão

A interface do sistema foi desenvolvida com embasamento nas teorias e premissas apresentadas no referencial teórico. A eficiência foi obtida com a implementação de um sistema que procura ser intuitivo, automatização de certos procedimentos possíveis e desenvolvimento das opções padrão para que o usuário apenas confirme o passo,

Descrição	Relatório?	Editar	Excluir
Metas e percentual de conclusão	<input type="checkbox"/>		
Consulta Andamento Projeto	<input checked="" type="checkbox"/>		
Prazos Críticos Projeto	<input checked="" type="checkbox"/>		
Paracoccidiodomicose - Sintomas Frequentes	<input checked="" type="checkbox"/>		
PCM - Contagem Casos Lesão Cutânea	<input checked="" type="checkbox"/>		
Meningite - Casos Fatais	<input checked="" type="checkbox"/>		
Meningite - Contagem Casos por Região	<input checked="" type="checkbox"/>		

Figura 3.24. Tela de administração das consultas e relatórios salvos.

quando a opção padrão o atender. Considera-se que o SiglaSearch é eficaz por extrair qualquer subconjunto de informações de forma relacionada e com exibição clara. É possível aplicar critérios de ordenação, filtros e sumarização de informações, sem a necessidade de codificação pelo usuário, ao contrário de diversas ferramentas de apoio à bioinformática citadas. Os princípios de apreensibilidade, efetividade e adaptabilidade também foram aplicados ao sistema.

A apreensibilidade é relacionada com o aproveitamento da experiência anterior do usuário. Como aplicação desse princípio, pode-se citar os botões *Novo*, *Abrir* e *Salvar*, que fazem analogias aos editores de texto, e que ficam no canto superior esquerdo. E os campos e filtros utilizados na consulta são apresentados no lado esquerdo da tela, assim como são apresentados parâmetros de pesquisa em *sites* de pesquisa de preços.

A efetividade é definida como a capacidade de mudar decisões anteriores com flexibilidade e completar as tarefas com facilidade, eficácia e segurança. No SiglaSearch qualquer definição de consulta pode ser alterada ou excluída, através de caminhos redundantes, o que demonstra a flexibilidade completa do sistema.

A adaptabilidade está relacionada com a multiplicidade de caminhos para executar as funções. A maioria das funcionalidades do SiglaSearch tem mais de um caminho para acesso. No editor de consultas, os campos podem ser adicionados pelo assistente ou pelo botão de adição de campo. A alteração e a exclusão de campos selecionados podem ser feitas acessando os botões do lado esquerdo do editor de consulta ou acessando o menu de contexto para cada coluna na tabela de resultados. A sumarização ou a alteração de descrição de campo são formas de atualização de campo selecionado, disponibilizadas também no menu de contexto, para agilizar tais alterações de forma

simplificada sem acessar todos os passos de alteração de campo selecionado. Para adição de filtros e critérios de ordenação existem os respectivos botões no canto superior esquerdo da tela. Mas para aplicação de um desses dois itens que utilizem um campo selecionado, basta clicar na coluna do respectivo item para filtrar ou ordenar.

Os conceitos da Engenharia Semiótica são aplicados ao SiglaSearch para melhorar a eficácia da comunicação do sistema com o usuário. Os signos estáticos, dinâmicos e metalinguísticos são explorados para transmitir conhecimentos adequados ao usuário, como documentações, instruções iniciais, dicas nos passos menos intuitivos, ícones de botões e de menus e mensagens de validação.

Para equilibrar facilidade de uso e expressividade da consulta foram eliminadas funções que são raramente usadas. Os filtros, por exemplo, são reunidos somente por conjunção, em que todos os filtros devem ser satisfeitos, o que atualmente é aplicado na maioria dos sistemas de busca. Assim, o sistema não requer raciocínio lógico do usuário para combinações de conjunção (todas as condições satisfeitas) e disjunções (basta que uma das condições seja satisfeita). As opções padrão também foram aplicadas, de forma que o usuário só precisa confirmá-las, se não optar por redefinir o item. As junções entre os dados são feitas automaticamente pelo sistema, de forma que o usuário se concentra nos campos a serem buscados, filtragem de resultados e ordenações.

O SiglaSearch foi projetado para atender todas as partes interessadas, através de diferentes visões de sistema para suprir as diferentes necessidades. O sistema tem o perfil de editor de consulta para o administrador do sistema e os perfis de execução de relatórios para que usuários especializados consultem informações específicas.

A maioria das interfaces de consulta para sistemas de bioinformática apresenta baixa usabilidade, exigindo que os usuários editem as próprias consultas através de codificação como o MIMIC-II [Scott *et al.*, 2013], onde o usuário deve consultar os dados através da linguagem SQL e conhecer o esquema de dados previamente, o PACSY [Lee & Markley, 2012], que requer que o usuário edite consultas SQL quando desejar recursos diferentes de seleção de tabelas e atributos, e o dbSNP [Saccone *et al.*, 2011], que oferece consulta somente através de edição de códigos SQL. O SiglaSearch, por outro lado, procura conciliar usabilidade e eficácia da consulta. Através do SiglaSearch, o usuário pode consultar quaisquer subconjuntos de dados de um *workflow*, com aplicação de um conjunto variado de filtros sobre os dados, critérios de ordenação e sumarização de dados, através de uso de interface que busca ser intuitiva, sem nenhuma edição de consultas SQL pelo usuário.

Há sistemas que só admitem a execução de consultas pré-compiladas, sem flexibilidade, como o sistema FlyAtlas [Robinson *et al.*, 2013]. O SiglaSearch capacita o usuário para criação de consultas *ad hoc*, com uma grande expressividade, ainda

que não seja especialista em tecnologia da informação. E permite que as consultas sejam salvas como relatórios parametrizados, para que outros usuários possam realizar consultas pré-definidas, com um número mínimo de passos. A criação e a edição de relatórios pelo editor de consulta em poucos minutos apresentam vantagens em relação aos sistemas de consultas pré-compiladas que exigem customizações de *software* demoradas. O SiglaSearch estreita ainda mais a distância entre a tecnologia da informação e os profissionais de outras áreas do conhecimento, como a medicina, a biologia e a física, que já usam o LIMs SIGLa em suas pesquisas.

Existem interfaces muito limitadas, que restringem a consulta a uma única entidade de dados, sem relacionamento entre as informações do sistema ou com os filtros muito limitados. No LabKey Server [Nelson *et al.*, 2011], a interface limita a consulta em uma tabela específica e conjuntos de colunas de tabelas relacionadas e, para aumentar a abrangência da consulta, deve-se aplicar consultas SQL. A interface de consulta proposta por Vijayprasath & Rajan [2015] permite que o usuário consulte uma única tabela a cada consulta. O sistema MIMIC-II [Scott *et al.*, 2013] admite apenas filtros de igualdade e de intervalo. O sistema MaxQB [Schaab *et al.*, 2012] admite apenas filtros de igualdade. O SiglaSearch, por outro lado, permite o uso dos dados do sistema como um todo, reunidos por relacionamentos concisos entre as entidades de cada laboratório de pesquisa específico. Também maximiza as possibilidades de filtros, ordenações e sumarizações de dados. O SiglaSearch oferece filtros de *Igual*, *Diferente*, *Maior*, *Menor*, *Maior ou igual*, *Menor ou igual*, *Intervalo*, *Como* e *Não é como*. E oferece funções de sumarização dos tipos *Contagem*, *Somatório*, *Máximo*, *Mínimo* e *Média*.

Existem interfaces que admitem apenas pesquisas por palavras-chave, como o iLAP [Stocker *et al.*, 2009], o iSearch [Zeng *et al.*, 2012], o sistema de consulta em estruturas de dados hierárquica proposto por Wilson [2003], o sistema de consulta para busca da menor sub-árvore possível do *workflow* que contenha todos os termos pesquisados proposto por Liu e colaboradores [2010] e o sistema WISE [Shao *et al.*, 2009]. Embora só existam sistemas de consulta em *workflow* através de palavras-chave [Cohen-Boulakia & Leser, 2011], o SiglaSearch, no ponto de vista do usuário, se comporta como um sistema comum de entidades e conceitos de dados pré-definidos, permitindo a consulta aos dados de forma estruturada. Ao fornecer uma interface de consulta com boa expressividade, sem a necessidade de edição de código, o SiglaSearch sobressai sobre vários outros sistemas de consulta aplicados à bioinformática.

Capítulo 4

Implementação da Engenharia de Consulta

O SiglaSearch é um módulo construído dentro do projeto SIGLa e, dessa forma, utiliza as mesmas ferramentas de desenvolvimento e de banco de dados. Para desenvolvimento da ferramenta foi utilizada a tecnologia Java para Web (J2EE), versão 1.7, e o SBDG MySQL [MySQL, 2015]. O projeto é compatível com o servidor Web Apache Tomcat [Foundation, 2015].

O SIGLa utiliza o *framework* Web Java Server Faces (JSF), versão 2.2. Conforme especificado no *framework* JSF, a aplicação possui arquitetura MVC (Model-View-Controller - Modelo-Visão-Controle). Esse modelo MVC define as camadas modelo, visão e controle na aplicação, garantindo um desacoplamento das funcionalidades e manutenção da aplicação com maior facilidade, o que significa baixo acoplamento e alta coesão do código-fonte [Attorre, 2014]. A camada de Visão é responsável pela interface do sistema com o usuário e é composta pelos códigos em XHTML, JavaScript e folhas de estilo (CSS). A camada de Controle gerencia requisições do sistema, controla consultas e persistência de dados e implementa as regras de negócio. É composta pelas classes Beans, implementadas em Java. A camada Modelo, também implementada em Java, representa os dados a serem persistidos na aplicação. A persistência de dados é feita conforme especificação Java Persistence API (JPA), com implementação do Hibernate. O SIGLa utiliza o PrimeFaces [Civici & DARÇIN, 2015] como *framework* complementar ao JSF para a construção ágil da interface do sistema, pelo reuso de componentes.

O sistema de consulta foi construído mantendo a compatibilidade com os demais módulos do sistema. Os resultados da consulta são filtrados pelos controles de acesso lógico, criados no trabalho de Hanke [2014] para o SIGLa, para assegurar a proteção

dos dados, programas e sistemas contra acessos indevidos de pessoas ou outros sistemas não autorizados. O módulo de consulta SiglaSearch foi desenvolvido em um total de 5338 linhas de código.

Conforme revisão bibliográfica, não existe pesquisa em dados de *workflow* senão através de palavras-chave. Nos Capítulos 3 e 5 são mostrados exemplos de pesquisa em dados nos *workflows* do SIGLa, com uso dos dados estruturados e identificação dos relacionamentos existentes entre as atividades. Neste capítulo é explicada a especificação e implementação da engenharia de consulta do SIGLa, denominada SiglaSearch, que possibilita a geração desses resultados de pesquisa. Para explicar o funcionamento, será utilizado o *workflow* de Gestão de Projetos, mas o sistema pode ser aplicado em qualquer *workflow* carregado no SIGLa.

4.1 Modelo de dados do SIGLa

Para entender o funcionamento da engenharia de consulta, é necessário entender o modelo de dados do SIGLa. Esse modelo contém 65 tabelas básicas, para gravação das informações do laboratório de pesquisa, auditoria, controle de acesso de usuários, informações de controle de modos de gravação e persistência de consultas. As tabelas de persistência de consultas são as criadas para o SiglaSearch e serão tratadas em seção específica. Como o foco dessa dissertação é a consulta aos dados de laboratório, serão apresentadas nesta seção somente as tabelas para gravação de dados dos laboratórios. As tabelas básicas para gravação de dados são *workflow*, *instance*, *activity*, *attribute*, *transition*, *record* e *recordAttribute*. A Figura 4.1 mostra o modelo de dados dessas tabelas. A descrição de cada uma dessas tabelas está listada na Tabela 4.1.

Tabela	Descrição
<i>workflow</i>	<i>Workflows</i> gerenciados pelo SIGLa
<i>instance</i>	Instâncias de <i>workflows</i> executadas
<i>activity</i>	Atividades executadas
<i>attribute</i>	Atributos das atividades executadas
<i>transition</i>	Transições entre pares de atividades executadas
<i>record</i>	Registros para representação de atributos <i>register</i> e <i>select</i>
<i>recordAttribute</i>	Campos de registros que representam atributos <i>register</i> e <i>select</i>

Tabela 4.1. Descrições das tabelas de gravação dos dados dos laboratórios no SIGLa

O modelo de dados do SIGLa foi criado para gravação altamente flexível. Alguns sistemas de apoio à bioinformática possuem uma flexibilidade parcial como o LabKey

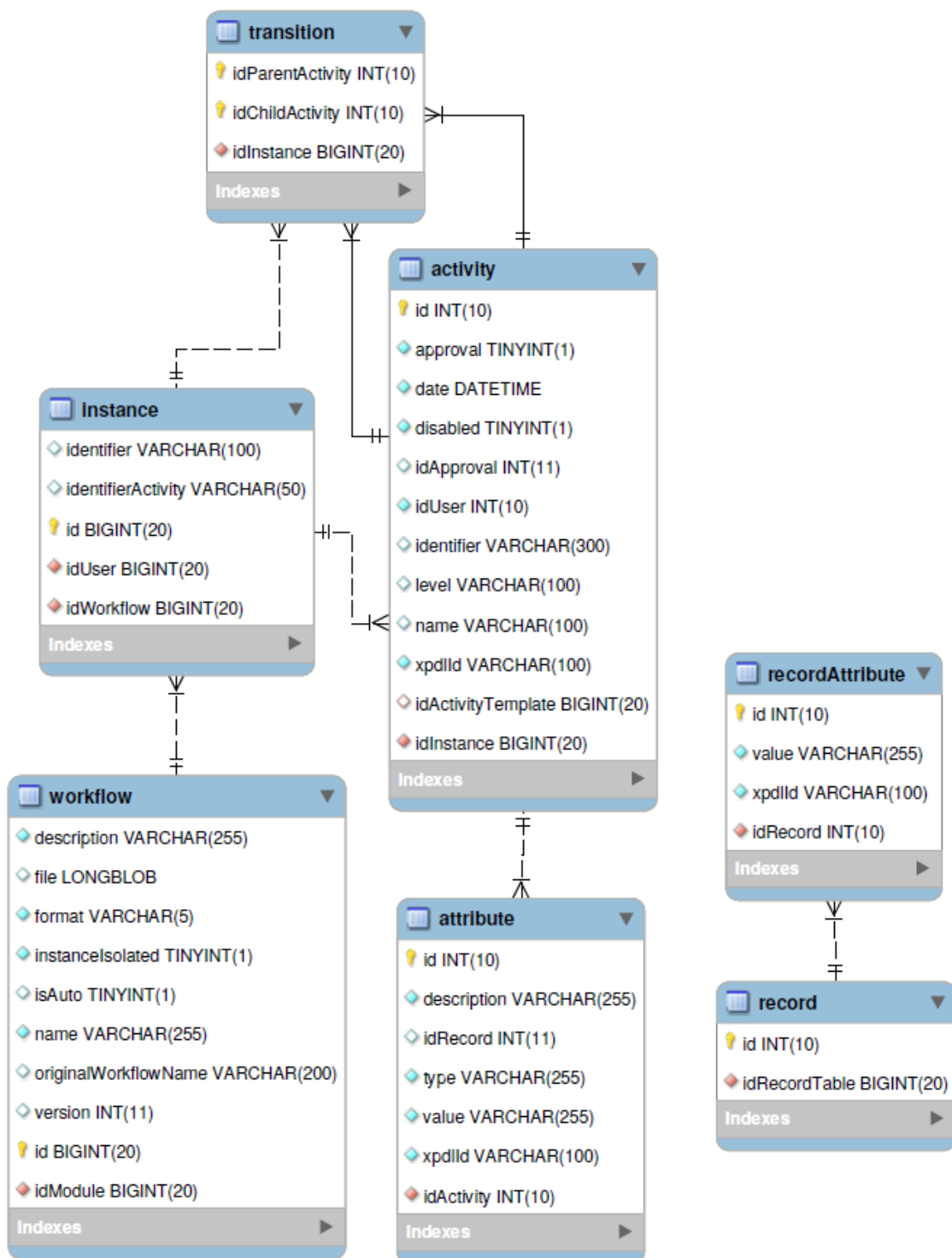


Figura 4.1. Esquema das tabelas básicas para gravação das informações dos laboratórios do sistema SIGLA.

Server e OpenSDE, para gravação de qualquer quantidade e conjuntos de atributos para cada entidade, através de modelo EAV (Entidade-Atributo-Valor). Esse módulo de gravação possui tabela à parte para a gravação de atributos, um por registro de tabela. Mas as entidades nesses sistemas são fixas. No SIGLa, por outro lado, todo o processo de gravação é flexível e as entidades dos laboratórios são gravadas no sistema sob a forma de atividades. E as colunas dessas entidades são gravadas sob a forma de atributos das atividades. A tabela de atributos possui colunas que descrevem o tipo de atributo e coluna *valor* para gravar o valor do atributo. A coluna *valor* da tabela *attribute* é a única do sistema que pode ser multivalorada, ou seja, gravar uma lista de informações em um único registro de atributo. Os atributos possuem 27 tipos. Entre esses, existem os tipos *register* e *select*. Para os atributos desses dois tipos, as informações são gravadas nas tabelas específicas de registros. As referências a esses registros são gravados no campo de valor da tabela de atributo, funcionando como uma chave estrangeira.

O modo de gravação dos dados de laboratório não é fixa. Eles podem ser gravados em tabelas básicas ou em tabelas externas. As tabelas básicas são aquelas presentes na arquitetura central do SIGLa. E as tabelas externas são as tabelas criadas especificamente para determinados laboratórios. As tabelas externas são usadas tanto para a gravação de registros quanto para a gravação de atributos de atividades. As tabelas externas para atributos de atividades podem gravar parte dos atributos ou todos os atributos das atividades. Quando alguns atributos são gravados em tabelas externas, tal tabela aponta para o registro de atividade correspondente da tabela *activity*. Os registros que representam os tipos *register* e *select* também podem ser gravados como tabelas externas ou internas. Quando gravados sob a forma de tabelas externas, o valor de atributo aponta para a tabela externa. Quando gravados em tabelas internas, aponta para a tupla da tabela *record* que representa o registro. No caso de registro de tabela interna, as colunas do registro são gravados na tabela *recordAttribute*, similar à gravação de atributos das atividades. Essa criação de tabelas específicas para determinados laboratórios não fere a flexibilidade do modelo de dados do sistema, pois funciona de forma incremental através de referências às tabelas fixas, sem alterar a arquitetura central do sistema.

O *workflow* de Gestão de Projetos possui quatro tabelas externas para gravação de parte dos dados das atividades *projeto*, *despesas*, *metas* e *atividade*, representadas na Figura 4.2. Esse *workflow* utiliza três especificações de registro: *agência financiadora*, *beneficiário* e *membros de equipe*. Registros de agência financiadora armazenam o conjunto de Agências de Fomento. Os registros de beneficiário armazenam os beneficiários para as despesas do projeto. E os registros de membros de equipe representam as

informações de coordenador do projeto, responsável por gestão de equipe e por gestão financeira, membros do projeto, responsável de cada meta e executores de cada meta. Os registros de beneficiário e de membros de equipe são gravados nas tabelas internas de dados de registro. E o registro de agência financiadora é gravado em tabela externa, conforme Figura 4.3.

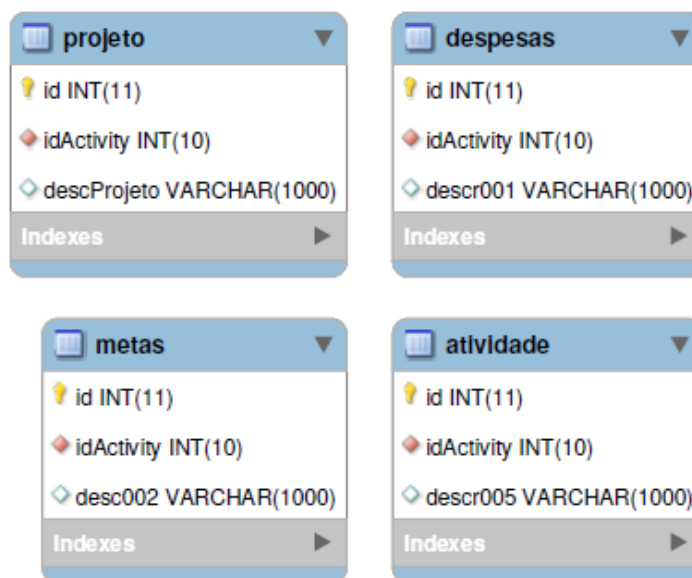


Figura 4.2. Tabelas externas para gravação de dados de atividades do *workflow* de Gestão de Projetos.

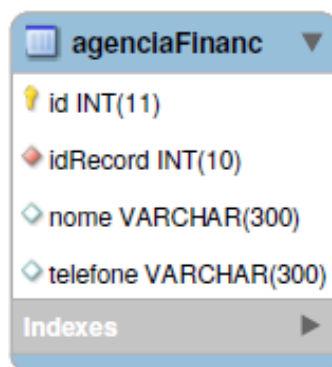


Figura 4.3. Registro do *workflow* de Gestão de Projetos gravado em tabela externa.

Os modos de armazenamento em tabelas básicas e em tabelas externas exercem o mesmo efeito de armazenamento e busca. No entanto, devem ser bem definidos ao projetar o *workflow*, de modo a oferecer maior eficiência de armazenamento, execução da consulta e robustez nos sistemas. Tabelas internas de atividades só utilizam espaço

para armazenar os atributos não nulos. Dados heterogêneos, com poucos atributos aplicados em cada instância de *workflow*, devem ser gravados em tabelas internas, para evitar tabelas esparsas que ocupam mais espaço em disco e prejudicam o desempenho das consultas. Tabelas externas são ideais em conjuntos de atributos que são frequentemente usados, o que significa que a tabela terá poucos dados esparsos. As tabelas externas, quando armazenam atributos frequentemente aplicados, também melhoram o desempenho das consultas, pois diminuem a necessidade de junções entre várias relações da mesma tabela *attribute*. Por outro lado, a utilização exclusiva de tabelas internas, com a finalidade de diminuir o total de tabelas, é importante quando o SIGLa é aplicado no contexto de aplicações móveis. A versão do SIGLa para dispositivos móveis está sendo desenvolvida pelo LUAR. Como esses dispositivos normalmente usam SGBD SQLite [Mednieks *et al.*, 2012], sistema gerenciador de banco de dados de aplicações móveis menos robusto, é necessário que o banco de dados seja mais enxuto para ser executado nesses ambientes de menor robustez.

4.2 Definição de relacionamento dos dados de *workflow*

Hoje existem várias ferramentas para construção de consulta *ad hoc* para sistemas modelados em RDBMS. Através da análise das restrições de integridade, as consultas são montadas e executadas, a partir dos campos selecionados e filtros informados pelo usuário. O SIGLa foi modelado usando RDBMS, mas o relacionamento entre as entidades modeladas, através de chave estrangeira, não busca os valores que o usuário deseja. Ao cruzar as chaves estrangeiras das tabelas de *instância*, *atividade* e *atributo*, por exemplo, obtemos resultados com centenas de linhas, com um atributo por linha, que não promovem clareza aos usuários para análise das informações. Isso ocorre porque o SIGLa foi modelado em função dos conceitos de *workflow* para ser altamente flexível. As consultas *ad hoc* para RDBMS, portanto, não atendem ao problema de consulta *ad hoc* do SIGLa.

O estado da arte atual de pesquisa em *workflow* é a realização de consultas por palavras-chave. Como resultado, a consulta mostra um subconjunto de nós ou de sub-árvores onde as palavras-chave são encontradas. Para contornar o problema de pesquisa em *workflow*, para relacionamento dos conceitos da situação modelada em um cenário específico, alguns projetos de pesquisa utilizam ontologias, o que exige tempo de processamento, pessoal especializado e redundância da informação, pois são criados dois bancos de dados, um para gravação de dados e outro com a ontologia para a pesquisa.

A aplicação das ontologias é adequada para bases de dados estáticas. No SIGLa, por outro lado, os dados gravados devem estar prontos para consulta. E a alteração, a inserção e a pesquisa de dados são frequentes. Por isso, foi necessária a criação de uma engenharia de consulta capaz de buscar informações com relacionamento entre as entidades trabalhadas por cada laboratório específico sem a conversão dos dados para ontologias ou para modelos que se relacionam pelas entidades do laboratório específico.

O sistema de consulta deve mostrar entidades específicas usadas em cada laboratório com seus dados e seus relacionamentos. Essas entidades, embora não sejam modeladas no banco de dados SIGLa, devem ser extraídas no momento da consulta. Como exemplos de entidades do laboratório específico, pode-se citar *sintoma*, *exame*, *identificação*, *responsável*, *meta* e *projeto*. Podemos concluir, portanto, que as entidades específicas de cada laboratório são os diferentes tipos de atividades especificadas em um *workflow*. No *workflow* de Gestão de Projetos, por exemplo, as entidades do laboratório são *projeto*, *gestão de equipe*, *gestão financeira*, *despesas*, *gestão científica*, *metas* e *atividades*.

Para realizar a consulta foi necessária a especificação de como as atividades executadas, que representam as entidades modeladas para um *workflow* específico, se relacionam. A Figura 4.4 ilustra uma instância do *workflow* de Gestão de Projetos, onde cada nó da árvore representa uma atividade executada. A especificação do *workflow* também é um grafo, conforme Figura 5.1, em que cada nó representa o conceito de uma atividade. Uma instância do *workflow*, por outro lado, é uma árvore. Os relacionamentos entre as entidades do laboratório específico são extraídos das instâncias executadas. Seja $G(V,E)$ o grafo que representa uma instância de um *workflow*, onde V é o conjunto de nós que representam as atividades executadas e E , o conjunto de arestas que representam as transições entre as atividades executadas. Seja v_0 a atividade raiz da árvore que representa a instância. Os nós folha de uma árvore são aqueles que não possuem filhos. Consideramos que duas atividades v_a e v_b estão relacionadas se existe um caminho de vértices formado pela sequência de vértices v_0, v_1, \dots, v_n , onde v_0 é o nó raiz, v_n é o nó folha, tais que $(v_i, v_{i+1}) \in E$, $0 \leq i < n$ e $0 \leq a < b \leq n$.

Para realizar as consultas entre as entidades do laboratório, que não são especificadas no modelo relacional do SIGLa, foi criado o conceito de *grupo de atividades* para cada instância do *workflow*. Cada grupo de atividades é formado pelo conjunto de atividades presentes em um caminho formado pela sequência de vértices v_0, v_1, \dots, v_n , tais que $(v_i, v_{i+1}) \in E$, v_0 é o nó raiz, v_n é um nó folha, e $0 \leq i < n$. Dessa forma, quaisquer pares de atividades executadas que pertencem a um mesmo grupo são consideradas relacionadas. As atividades executadas podem estar presentes em vários grupos. Uma atividade executada de *projeto* do *workflow* de Gestão de Projetos, por exemplo, está



Figura 4.4. Representação de uma instância do *workflow* Gestão de Projetos, com atividades rotuladas com seus identificadores.

presente em todos os grupos da sua instância, por ser o nó raiz da árvore que representa tal instância. No momento em que o *workflow* é escolhido, os grupos de atividades são gerados com um identificador único. Aqui são mostrados os relacionamentos e a consulta de uma única instância do *workflow*, apenas para que a explicação fique didática. A consulta de todas as instâncias de *workflow* é análoga à consulta de dados de uma única instância. Como não existe relacionamento entre atividades executadas de instâncias diferentes, o resultado da consulta de todas as instâncias é formado pela união das tuplas encontradas na consulta de cada instância executada.

Para a geração de grupos de uma instância, é necessária a geração de todos os caminhos que ligam o nó raiz aos nós folha de uma instância. Os caminhos são gerados através de *scripts* SQL, em tabelas temporárias de nome *pathTransitions* $\langle t \rangle_ \langle u \rangle$, onde $\langle t \rangle$ é o número de nós representados e $\langle u \rangle$ é o identificador do usuário conectado ao sistema. Como não se sabe *a priori* o tamanho máximo desses caminhos nas instâncias, fazemos consulta incremental de n em n nós até que todos os vértices sejam

consultados e anexados ao caminho. Atualmente n está definido com o valor 4.

A Figura 4.4 mostra a instância rotulada com os identificadores de cada atividade executada. Para obter a raiz de cada instância, busca-se na tabela *activity* todas as atividades executadas do tipo de atividade especificado como raiz do *workflow*. No caso do *workflow* de Gestão de Projetos, a atividade raiz é *projeto*. Em seguida, são buscadas as próximas atividades através de várias seleções da tabela *transition*, onde em cada seleção são buscados os nós de um nível da árvore. O tamanho máximo de caminho para essa instância é cinco. O que significa que são necessárias duas execuções de *scripts* SQL. Uma execução para buscar os quatro primeiros níveis e outra para buscar o quinto nível. A verificação de conclusão dos caminhos é feita quando não há nó não nulo no último nível do caminhos criados.

A Figura 4.5 mostra as tabelas usadas para extração dos grupos de atividades. As tabelas *pathTransitions4_5* e *pathTransitions8_5* são tabelas temporárias para a geração dos caminhos. São geradas por demanda, para cada usuário e excluídas logo após a formação dos grupos. A tabela *groupActivity* grava todos os relacionamentos entre grupos e atividades executadas, extraída da última tabela de caminhos gerada e é usada como insumo para a produção das consultas. Os registros são gerados para cada usuário, pois devem refletir o conteúdo de dados no momento em que o usuário aciona a consulta.

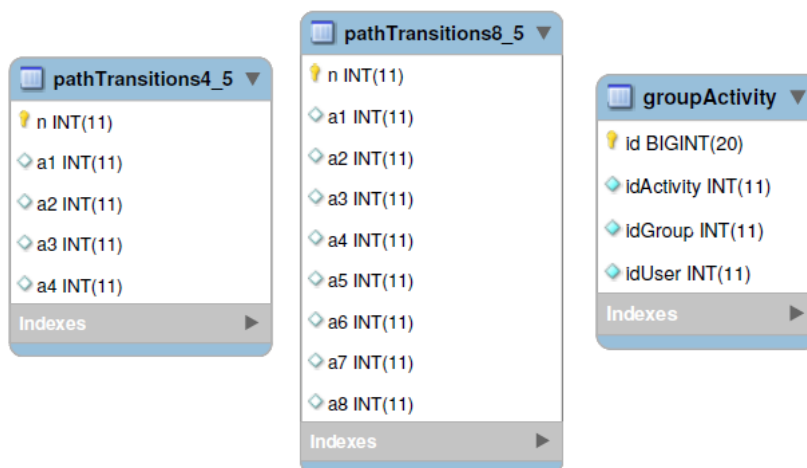


Figura 4.5. Tabelas usadas para a extração dos relacionamentos entre as atividades.

A Figura 4.6 mostra o resultado da extração dos quatro primeiros nós de cada caminho. E a Figura 4.7 mostra o resultado da extração dos oito primeiros nós de cada caminho. Nota-se que todos os caminhos foram completamente gerados, pois não

existe nenhum nó não nulo no oitavo nível. Nota-se também que o tamanho máximo para um caminho nesse *workflow* é cinco.

Seja c o tamanho do maior caminho dentre todas as instâncias executadas. Como temos que percorrer todos os níveis, para gerar os relacionamentos entre todas as atividades executadas, e certificamos que todas as atividades foram computadas quando todos os nós do último nível são nulos, o valor ideal para n para um *workflow* consultado, denominado aqui de m , é $(c+1)$. Quando $m = n$, é necessária a construção de uma única tabela para o armazenamento dos caminhos e a construção e a execução de uma única consulta no banco de dados para a geração dos caminhos. Quando $m \neq n$, o número de tabelas de armazenamento de caminhos e o número de consultas ao banco de dados, denominado aqui de e , é dado pela fórmula $\lceil \frac{m}{n} \rceil$. E o número de seleções desnecessárias à tabela *transition*, denominado aqui de d , é dado pela fórmula $\lceil \frac{m}{n} \rceil \cdot n - m$. Se, por exemplo, $n=4$ e $c=9$, então $m=10$. E consequentemente temos que $e=3$ e $d=2$, conforme mostrado a seguir:

$$\begin{aligned} e &= \lceil \frac{m}{n} \rceil \\ e &= \lceil \frac{10}{4} \rceil \\ e &= 3 \end{aligned}$$

$$\begin{aligned} d &= \lceil \frac{m}{n} \rceil \cdot n - m \\ d &= \lceil \frac{10}{4} \rceil \cdot 4 - 10 \\ d &= 3 \cdot 4 - 10 \\ d &= 2 \end{aligned}$$

Quanto maior o número de execuções e , maior a quantidade de consultas criadas e de acessos ao banco de dados. E quanto maior o número d de seleções desnecessárias à tabela *transition*, mais onerosa fica a geração de caminhos. Como melhorias para essa engenharia de consulta, pode-se otimizar a escolha do valor n , através, por exemplo, do armazenamento do tamanho máximo de instância para cada *workflow* consultado, para que nas consultas posteriores tal número seja usado como parâmetro, o que representa

mais chances de acerto ou aproximação de n com m .

#	n	a1	a2	a3	a4
1	2	169	178	NULL	NULL
2	3	169	181	191	NULL
3	4	169	198	202	210
4	5	169	198	202	217
5	6	169	198	202	224
6	7	169	198	202	231
7	8	169	198	246	254
8	9	169	198	246	283
9	10	169	198	298	306
10	11	169	198	298	313
11	12	169	198	298	320

Figura 4.6. Extração dos quatro primeiros nós dos caminhos do nó raiz aos nós folha.

#	n	a1	a2	a3	a4	a5	a6	a7	a8
1	2	169	178	NULL	NULL	NULL	NULL	NULL	NULL
2	3	169	181	191	NULL	NULL	NULL	NULL	NULL
3	4	169	198	202	210	NULL	NULL	NULL	NULL
4	5	169	198	202	217	NULL	NULL	NULL	NULL
5	6	169	198	202	224	NULL	NULL	NULL	NULL
6	7	169	198	202	231	239	NULL	NULL	NULL
7	8	169	198	246	254	262	NULL	NULL	NULL
8	9	169	198	246	254	269	NULL	NULL	NULL
9	10	169	198	246	254	276	NULL	NULL	NULL
10	11	169	198	246	283	291	NULL	NULL	NULL
11	12	169	198	298	306	NULL	NULL	NULL	NULL
12	13	169	198	298	313	NULL	NULL	NULL	NULL
13	14	169	198	298	320	NULL	NULL	NULL	NULL

Figura 4.7. Extração dos oito primeiros nós dos caminhos do nó raiz aos nós folha.

Abaixo estão listados os códigos gerados pelo SiglaSearch para geração de caminhos de quatro nós e de oito nós:

```

/* Geração dos quatro primeiros nós de cada caminho*/
insert into pathTransitions4_5
select @rownum := @rownum + 1 as n,
       activity.id a1, t12.idChildActivity a2,
       t23.idChildActivity a3, t34.idChildActivity a4
from activity
left join transition t12
  on activity.id = t12.idParentActivity
left join transition t23
  on t12.idChildActivity = t23.idParentActivity
left join transition t34
  on t23.idChildActivity = t34.idParentActivity,
(SELECT @rownum := 0) r
where activity.xpdlId = 'p001';

/* Geração dos oito primeiros nós de cada caminho*/
insert into pathTransitions8_5
select @rownum := @rownum + 1 as n,
       a1, a2, a3, a4,
t45.idChildActivity a5, t56.idChildActivity a6,
       t67.idChildActivity a7, t78.idChildActivity a8
from pathTransitions4_5
left join transition t45
  on a4 = t45.idParentActivity
left join transition t56
  on t45.idChildActivity = t56.idParentActivity
left join transition t67
  on t56.idChildActivity = t67.idParentActivity
left join transition t78
  on t67.idChildActivity = t78.idParentActivity ,
(SELECT @rownum := 0) r;

```

Após a geração dos caminhos, o sistema gera a tabela de correspondência entre os grupos de atividades e as atividades para o usuário que executa a consulta, chamada

groupActivity. A Figura 4.8 mostra parte da tabela de correspondência entre grupos de atividades e atividades para a instância ilustrada. Abaixo está listado o *script* executado pelo sistema para alimentação da tabela *groupActivity*.

```
insert into groupActivity
select @rownum := @rownum + 1 as id, a, n, 5 user
from
(select n, a1 a from pathTransitions8_5 where a1 is not null union all
select n, a2 a from pathTransitions8_5 where a2 is not null union all
select n, a3 a from pathTransitions8_5 where a3 is not null union all
select n, a4 a from pathTransitions8_5 where a4 is not null union all
select n, a5 a from pathTransitions8_5 where a5 is not null union all
select n, a6 a from pathTransitions8_5 where a6 is not null union all
select n, a7 a from pathTransitions8_5 where a7 is not null union all
select n, a8 a from pathTransitions8_5 where a8 is not null) aux,
(SELECT @rownum := 0) r;
```

4.3 Execução da Consulta

Para relacionamento entre as entidades de um laboratório específico, que não estão gravadas no modelo de dados do SIGLa, utilizam-se os grupos de atividades gerados na tabela *groupActivity*. A origem dos dados gravados, os identificadores dos tipos de atividades e atributos, as colunas e origem de gravação de registros e demais informações sobre o *workflow* são consultadas no arquivo XPDL do respectivo *workflow*, através de *parser* já existente no SIGLa.

Todas as atividades são gravadas em uma mesma tabela. Para a extração dos dados de cada uma, faz-se seleção das tuplas dessa tabela com os respectivos identificadores de cada tipo de atividade. O identificador do tipo de atividade é gravado na coluna *xpdlId*. Analogamente, todos os atributos, gravados em tabela interna, são gravados na tabela *attribute* e devem ser buscados através de seleções, especificadas pela coluna *xpdlId* dessa tabela de atributos. Alguns atributos podem ser gravados em tabela externa e tal nuance deve ser detectada através de consulta à definição do *workflow* no arquivo XPDL.

Atributos do tipo *register* e *select* são gravados sob a forma registros e a coluna *value* da tabela *attribute* faz referência ao registro ou ao conjunto de registros. Os

#	id	idActivity	idGroup
1	2	169	2
2	3	169	3
3	4	169	4
4	5	169	5
5	6	169	6
6	7	169	7
7	8	169	8
8	9	169	9
9	10	169	10
10	11	169	11
11	12	169	12
12	13	169	13
13	14	169	14
14	15	178	2
15	16	181	3
16	17	198	4
17	18	198	5
18	19	198	6
19	20	198	7
20	21	198	8
21	22	198	9
22	23	198	10
23	24	198	11
24	25	198	12
25	26	198	13
26	27	198	14
27	28	191	3
28	29	202	4
29	30	202	5
30	31	202	6
31	32	202	7

Figura 4.8. Geração dos registros de correspondência entre grupos de atividades e atividades.

registros também podem estar gravados em tabela externa ou nas tabelas internas *record* e *recordAttribute*.

Para exemplificar a execução de uma consulta pelo SiglaSearch, será explicada aqui a consulta da Figura 5.2 do Capítulo 5, para a visualização de metas e atividades de um projeto. Essa consulta envolve as atividades *meta*, *atividade* e *projeto*. A Figura 4.9 mostra o plano de execução para a busca de dados de metas, a Figura 4.10, para a busca de atividades, e a Figura 4.11, para a busca de projetos. Para qualquer consulta de dados, o SiglaSearch monta uma única consulta, de forma que as otimizações de plano de execução do SBGD possam ser aplicadas. Aqui estão separados os planos de execução em partes, apenas para efeitos didáticos.

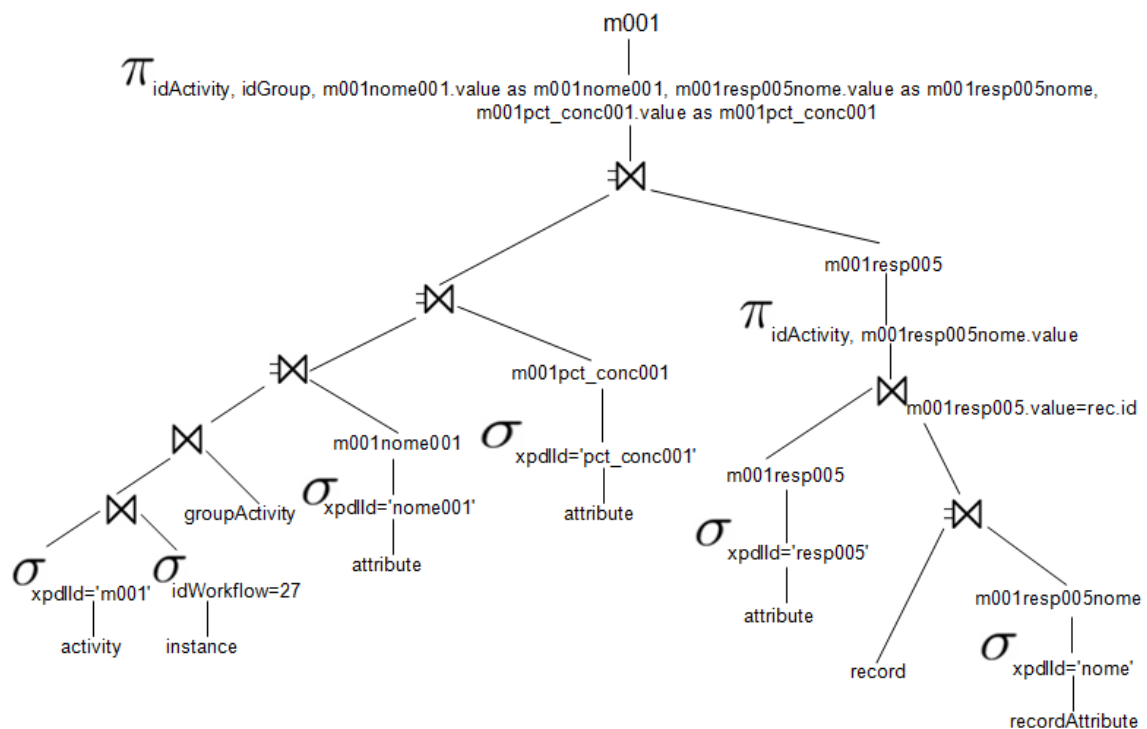


Figura 4.9. Plano de execução para a busca das metas.

Em cada uma dessas visualizações *inline*, para a busca de metas, atividades e projetos, cada tupla buscada contém o identificador do grupo. Esse identificador de grupo é utilizado para relacionar os diferentes tipos de entidades. Como os dados de *workflow* são naturalmente esparsos, a chance de atributos serem nulos ou de atividades não serem executadas é grande. Por isso, todas as junções que envolvem informações que podem ser nulas são externas, para que, ainda que alguma informação seja nula, demais informações buscadas não deixem de ser recuperadas. Para o relacionamento entre as atividades, cria-se uma visualização *inline* denominada *trunkView* que busca

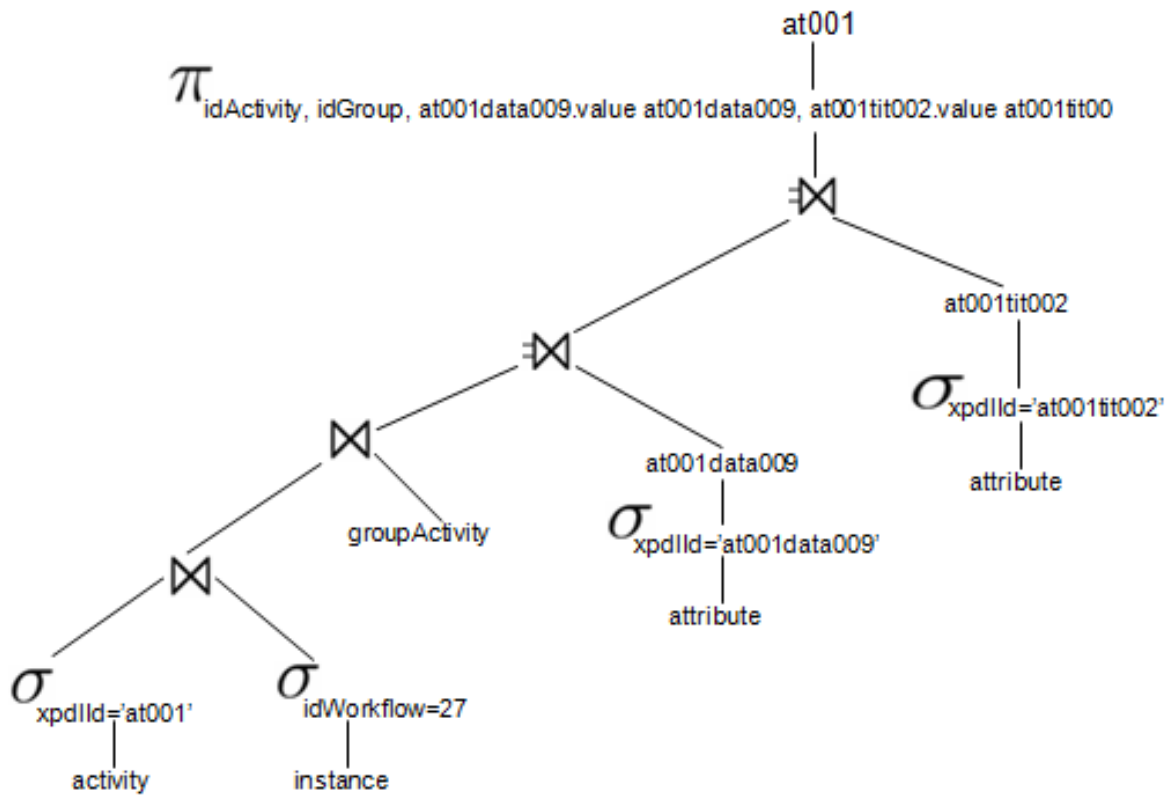


Figura 4.10. Plano de execução para a busca das atividades.

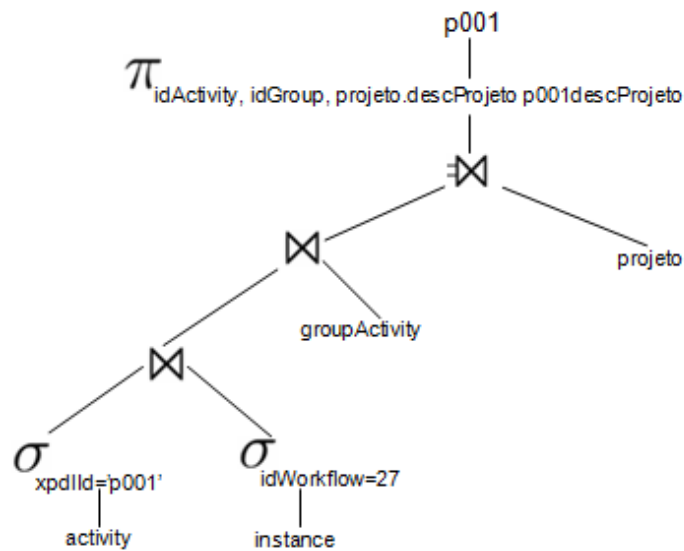


Figura 4.11. Plano de execução para a busca dos projetos.

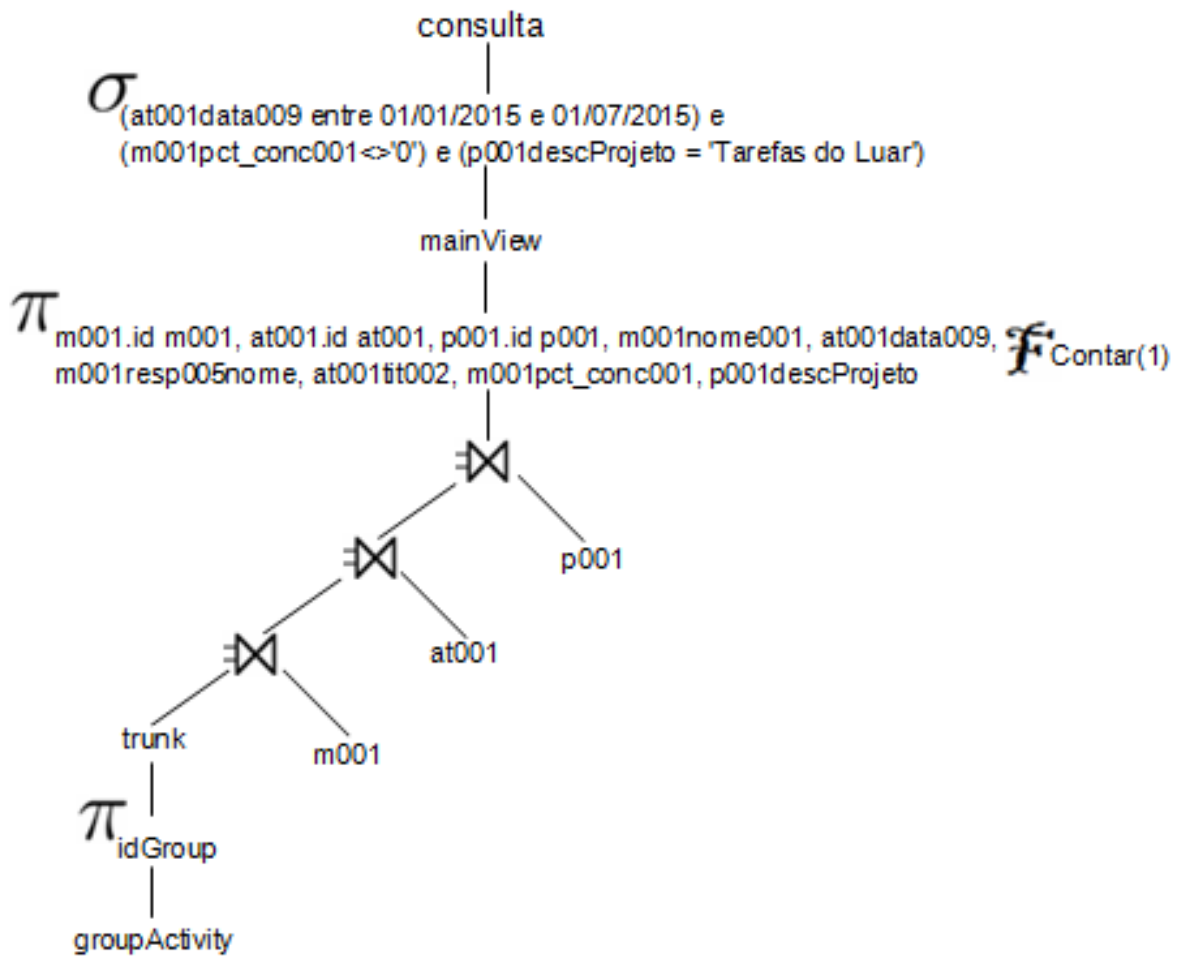


Figura 4.12. Plano de execução da consulta de metas e atividades de um projeto.

todos os identificadores de grupo geradas para o *workflow* consultado, para junção externa com as demais atividades. O plano de execução da consulta final é mostrado na Figura 4.12, onde os rótulos *m001*, *at001* e *p001* representam os planos de execução de metas, atividades e projeto, respectivamente.

Além das junções, deve-se eliminar registros duplicados que podem ser gerados quando o conjunto de atividades selecionadas está presente em mais de um caminho. Em uma consulta de projeto e metas, por exemplo, se uma meta tiver três atividades diferentes ela aparecerá no resultado em três tuplas, uma para cada atividade, pois se tratam de grupos diferentes. Para tratar o problema, basta fazer um agrupamento em função de todos os atributos selecionados e todos os identificadores das atividades selecionadas. Isso elimina os registros duplicados sem o risco de eliminar as tuplas de resultado. A visualização *inline* denominada *mainView* engloba os resultados gerados após a operação de eliminação de registros duplicados, conforme plano ilustrado. Sobre

a visualização *mainView* devem ser aplicadas as seleções de campo, filtros, critérios de ordenação e sumarizações solicitados.

Abaixo está listada a consulta gerada pelo SiglaSearch, para a geração dos resultados da consulta de metas e atividades. Para deixar a consulta mais clara e enxuta, os trechos de tratamento de permissão de acesso, tratamentos de tuplas nulas, conversões de tipo e tratamento das colunas de resultado feitos pelo SiglaSearch foram eliminados, da listagem abaixo, para facilitar o entendimento do código.

```

SELECT m001nome001 m001nome0010, m001resp005nome m001resp005nome1,
       m001pct_conc001 m001pct_conc0012, at001tit002 at001tit0023,
       DATE_FORMAT(at001data009, '%d/%m/%Y') at001data0094
FROM (select m001.id m001, at001.id at001, p001.id p001,
           m001nome001, at001data009, m001resp005nome,
           at001tit002, m001pct_conc001, p001descProjeto
      from (select groupActivity.idGroup
            from groupActivity) trunkView
LEFT JOIN
      (select activity.id, groupActivity.idGroup,
           m001nome001.value m001nome001,
           m001resp005nome,
           m001pct_conc001.value m001pct_conc001
      from activity JOIN instance
            on activity.idInstance = instance.id
      JOIN groupActivity
            ON activity.id = groupActivity.idActivity
LEFT JOIN attribute m001nome001
            ON m001nome001.idActivity = activity.id
            and m001nome001.xpdlId = 'nome001'
LEFT JOIN (select m001resp005.idActivity,
                 m001resp005nome.value m001resp005nome
           from attribute m001resp005
           JOIN record ON m001resp005.value = record.id
            and m001resp005.xpdlId = 'resp005'
           LEFT JOIN recordAttribute m001resp005nome
                ON m001resp005nome.idRecord = record.id
            and m001resp005nome.xpdlId = 'nome'
           ) m001resp005 ON m001resp005.idActivity = activity.id

```

```

LEFT JOIN attribute m001pct_conc001
      ON m001pct_conc001.idActivity = activity.id
      and m001pct_conc001.xpdlId = 'pct_conc001'
where instance.idWorkflow = 27
      and activity.xpdlId = 'm001') m001
      ON m001.idGroup = trunkView.idGroup
LEFT JOIN
(select activity.id, groupActivity.idGroup,
      str_to_date(at001data009.value, '%Y-%m-%d') at001data009,
      at001tit002.value at001tit002
from activity JOIN instance
      on activity.idInstance = instance.id
JOIN groupActivity
      ON activity.id = groupActivity.idActivity
LEFT JOIN attribute at001data009
      ON at001data009.idActivity = activity.id
      and at001data009.xpdlId = 'data009'
LEFT JOIN attribute at001tit002
      ON at001tit002.idActivity = activity.id
      and at001tit002.xpdlId = 'tit002'
where instance.idWorkflow = 27
      and activity.xpdlId = 'at001') at001
      ON at001.idGroup = trunkView.idGroup
LEFT JOIN
(select activity.id, groupActivity.idGroup,
      projeto.descProjeto p001descProjeto
from activity JOIN instance
      on activity.idInstance = instance.id
JOIN groupActivity
      ON activity.id = groupActivity.idActivity
LEFT JOIN projeto
      ON projeto.idActivity = activity.id
where instance.idWorkflow = 27
      and activity.xpdlId = 'p001') p001
      ON p001.idGroup = trunkView.idGroup
group by m001, at001, p001, m001nome001,
      at001data009, m001resp005nome, at001tit002,

```

```

        m001pct_conc001, p001descProjeto) mainView
WHERE at001data009 between str_to_date('2015-01-01', '%Y-%m-%d')
        and str_to_date('2015-07-01', '%Y-%m-%d')
and m001pct_conc001 <> '0'
and p001descProjeto = 'Tarefas do Luar'
ORDER BY m001nome001 ASC;

```

4.4 Modelo de dados para persistência da consulta

Para persistir os critérios de consulta para uso posterior e geração de relatórios, foi criado o modelo de dados da Figura 4.13. Nesse modelo podem ser gravados todos os critérios da consulta, que são as seleções de campos, os filtros e os critérios de ordenação. A Tabela 4.2 mostra a função de cada tabela de dados do modelo de persistência da consulta. A entidade *searchField* é um conceito genérico para representar qualquer campo de dados usado na consulta. Esse campo possui três especializações, denominadas *searchFieldAttributeValue*, *searchFieldAggregateFunction* e *searchFieldConstant*, para representação de campo de banco de dados, campo de sumarização e constante ou um conjunto de constantes respectivamente. Os registros de *searchField* são insumos para *searchSelectedField*, *searchFilter* e *searchSort*, para seleção de campos, filtros e critérios de ordenação. Cada filtro é composto de dois campos *searchField* e é o único item de consulta onde as constantes são aplicadas. *searchFieldAggregateFunction* também possui um *searchField* associado, com especialização *searchFieldAttributeValue*, pois o campo sumarizado representa a agregação de valores de um campo de banco de dados.

Tabela	Descrição
<i>search</i>	Consulta salva
<i>searchField</i>	Campo de dados genérico
<i>searchFieldAttributeValue</i>	Campo do banco de dados
<i>searchFieldAggregateFunction</i>	Campo sumarizado
<i>searchFieldConstant</i>	Constante
<i>searchSelectedField</i>	Campo selecionado
<i>searchFilter</i>	Filtro
<i>searchSort</i>	Critério de ordenação

Tabela 4.2. Descrições das tabelas de persistência da consulta do sistema Sigla-Search

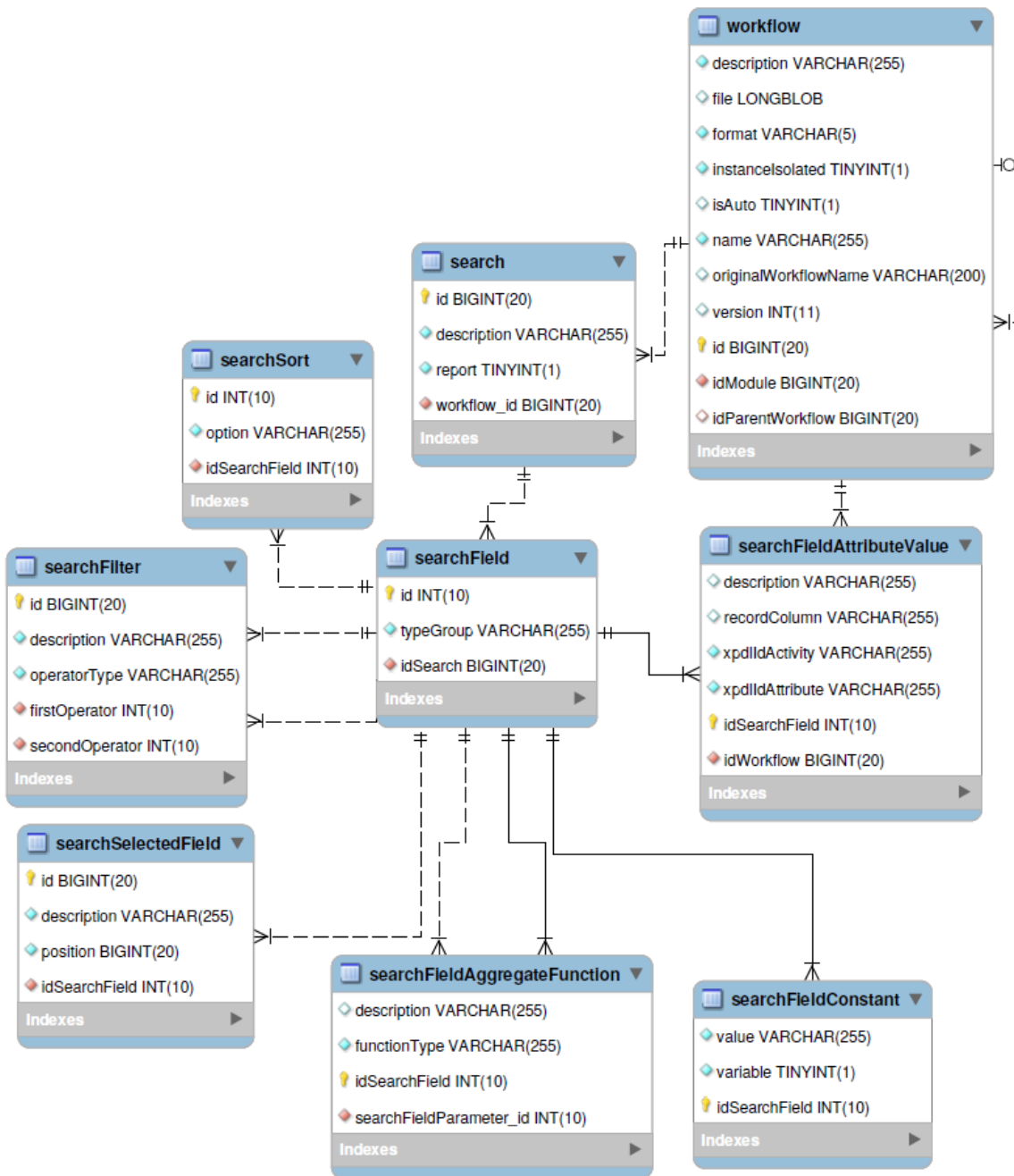


Figura 4.13. Modelo para persistência de dados das consultas do banco de dados do sistema SiglaSearch

Capítulo 5

Estudos de Caso: Modelagem de Consultas

Neste capítulo estão presentes três cenários de aplicação do SiglaSearch para busca de informação em *workflows* do SIGLa: Gestão de Projetos, estudo de Meningite e estudos da doença Paracoccidioidomicose (PCM). Esses cenários são apresentados para demonstrar a flexibilidade do SiglaSearch em consultas de diferentes *workflows* e a facilidade de geração de relatórios de detalhes e com dados sumarizados.

5.1 Gestão de Projetos

A gestão de projetos é regida pela norma ISO 21500 [Krause, 2014]. Essa norma se aplica ao gerenciamento de projetos de qualquer organização para qualquer tipo de projeto e fornece diretrizes para boas práticas de gerenciamento de projetos. A adaptação dos sistemas com as normas que os regem pode ser demorada, por depender de customizações de *software*. No SIGLa, por outro lado, a gestão de projetos foi modelada em função de *workflow*, através de parceria com o INMETRO. Para adaptação do sistema de gestão de projetos, basta a adaptação do *workflow* em um editor e carregamento no SIGLa. O *workflow* de Gestão de Projetos está ilustrado na Figura 5.1. Conforme figura, cada instância do *workflow* representa um projeto que pode ter dados de Gestão de Equipe, Gestão Financeira e Gestão Científica. A Gestão Científica é composta por metas e atividades. Metas podem ter sub-metas e cada sub-meta é composta de um conjunto de atividades. Para ilustrar o poder do SiglaSearch, são mostradas nessa dissertação duas consultas para acompanhamento de projeto: a Consulta de andamento para a busca de metas e de atividades de um projeto e a Consulta de prazos críticos para cada responsável do projeto.

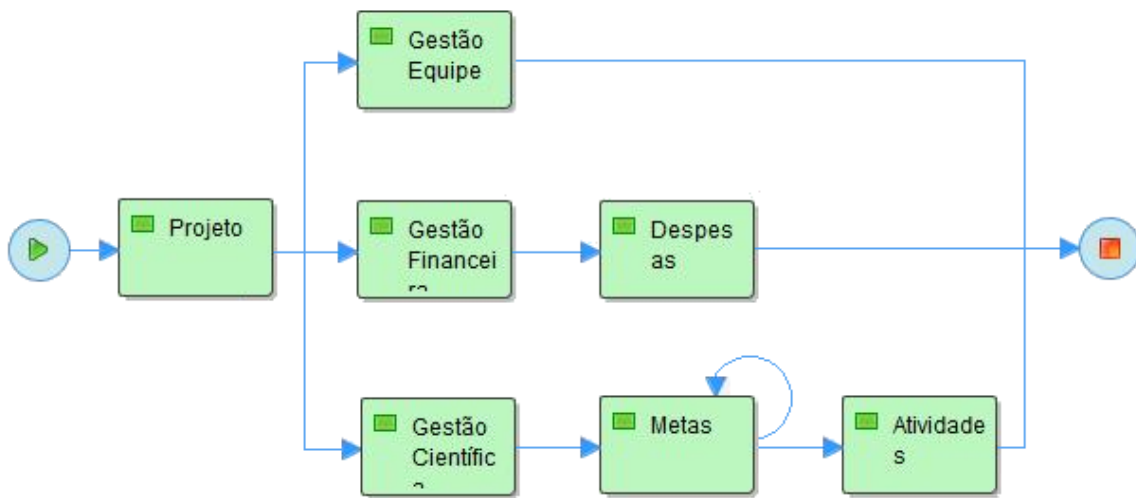


Figura 5.1. *Workflow* de Gestão de Projetos

Uma das necessidades importantes para gerência do projeto é a verificação do andamento do projeto através da consulta às metas e às suas respectivas atividades. A Figura 5.2 mostra a consulta desses dados. Essa consulta ilustra a criação de um relatório com os detalhes sobre o andamento de um projeto. Para a construção dessa consulta, basta a seleção dos campos de título, responsável e percentual de conclusão da meta e dos campos título e data de conclusão da atividade. O relacionamento entre as atividades e as metas é feito automaticamente pelo sistema. Para essa consulta foram definidos três filtros, um para intervalo de data de conclusão das atividades, um para filtrar percentual de conclusão e outro para filtrar um projeto específico. Pelos valores padrão dos parâmetros exibidos na figura, a consulta seleciona todas as metas e atividades do Projeto *Tarefas do Luar*, cujas atividades estão previstas para serem concluídas no primeiro semestre de 2015 e com as metas iniciadas. Nesse relatório os dados são ordenados pelo título da meta. O valor padrão de rótulo é o nome do atributo, mas, nesse caso, tanto o atributo da meta quanto da atividade tem rótulo *Título*. Por isso, os rótulos das colunas de título da meta e da atividade foram modificados para *Meta* e *Atividade*. Para mudar o rótulo, basta acessar o menu de contexto de cada coluna ou botão de visualização do lado esquerdo da tela para editar item. Caso essa consulta seja salva como relatório, este é disponibilizado no menu de relatórios e os três parâmetros adicionados podem ser modificados de acordo com a necessidade do usuário. O parâmetro de percentual de conclusão, por exemplo, pode ser modificado para diferente de 100%, para buscar metas não concluídas.

Outra consulta importante é verificação das últimas datas de conclusão das ati-

The screenshot shows a web application interface with a navigation menu at the top (Tela Principal, Nova Atividade, Registros, Relatórios, Ferramentas, Administração, Ajuda) and a sub-menu 'Editor de Consultas'. The main area is titled 'Consulta: Consulta Andamento Projeto' and contains a table with the following data:

Meta	Responsável Meta	Per	Atividade	Data Conclusão
Criação e geração de Reports concluídas	João Costa Goncalves	100	Criação do menu dinâmico para Report com os relatórios	24/06/2015
Criação e geração de Reports concluídas	João Costa Goncalves	100	Criação do formulário dinâmico do Report	24/06/2015
Criação e geração de Reports concluídas	João Costa Goncalves	100	Criação das funções para exportação dos resultados mod	18/06/2015
Editor de Workflows 2.0	André Pinto Ferreira	70	Parser do XPDL2 Editor	27/06/2015
Entrada de filtros com maior usabilidade	João Costa Goncalves	100	Limitação dos filtros disponíveis pelo tipo do campo seleci	27/06/2015
Flux 2	André Pinto Ferreira	40	Atributos multi-select, file	30/06/2015
Flux 2	André Pinto Ferreira	40	I18N Preview de Atributos	23/05/2015
Flux 2	André Pinto Ferreira	40	Parser do XPDL2 Editor	27/06/2015
Flux 2	André Pinto Ferreira	40	Pendências de registros/auditoria	23/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Busca de dados de registro	26/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento dos filtros para registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento da ordenação usando registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Limitação dos filtros disponíveis pelo tipo do campo seleci	27/06/2015
Registros acessíveis e utilizados na busca,	Bianca Fernandes Barros	90	Busca de dados de registro	26/06/2015
Registros acessíveis e utilizados na busca,	Bianca Fernandes Barros	90	Tratamento dos filtros para registro	25/06/2015
Registros acessíveis e utilizados na busca,	Bianca Fernandes Barros	90	Tratamento da ordenação usando registro	25/06/2015

Below the table, there are export buttons: 'Exportar Tudo' and 'Exportar Página Atual', with icons for XLS, PDF, CSV, and XML. The page status shows 'Página 1 de 1' and '16 resultado(s)'.

Figura 5.2. Consulta de andamento do Projeto

vidades de cada responsável do projeto, o que representa as atividades críticas, para evitar riscos de descumprimento dos prazos do projeto. A Figura 5.3 ilustra a última data de conclusão de atividade de cada responsável. Esse relatório ilustra a busca sumarizada com as datas mais importantes para análise da conclusão das atividades de um projeto. Para construir essa consulta, basta selecionar o campo responsável da meta e um campo de sumarização com função de máximo aplicada ao campo de data de conclusão da atividade. Nesse relatório também foi aplicado um filtro para consultar um projeto por vez, neste caso, *Tarefas do Luar*. A ordenação dos resultados é feita em função do responsável pela meta. Nesses resultados ilustrados na figura, vemos que o responsável *André Ferreira* possui a atividade mais crítica, com previsão de conclusão em *30/06/2015*.

5.2 Meningite

O *workflow* para pesquisa da doença Meningite foi desenvolvido em parceria com a Fiocruz [Fiocruz, 2015]. Essa pesquisa envolve doze atividades, conforme esquema do *workflow* (Figura 5.4). Cada instância do *workflow* representa os dados de um paciente. Nesse *workflow* estão definidas as atividades de Identificação do paciente, Internação, Exames clínicos, HIV, Exames complementares, Exame físico, Uso de antibiótico, Quadros infecciosos, Complicações imediatas, Complicações neurológicas, Diagnóstico e Encerramento do caso. Na entrada de dados para a pesquisa, não

The screenshot shows a web application interface with a menu bar at the top containing 'Tela Principal', 'Nova Atividade', 'Registros', 'Relatórios', 'Ferramentas', 'Administração', and 'Ajuda'. Below the menu is the 'Editor de Consultas' section, which includes buttons for 'Novo', 'Abrir...', and 'Salvar...', and options for '+ campo', '+ filtro', and '+ ordenação'. The main area displays a table titled 'Consulta: Prazos Críticos Projeto' with the following data:

Responsável	Data Conclusão Última Atividade
André Pinto Ferreira	30/06/2015
Bianca Fernandes Barros	27/06/2015
João Costa Gonçalves	27/06/2015

Below the table, it indicates 'Página 1 de 1' and '3 resultado(s)'. There are also buttons for 'Exportar Tudo' and 'Exportar Página Atual' with icons for various file formats like XLS, PDF, etc.

Figura 5.3. Consulta de prazos críticos do Projeto

são obrigatórias as execuções de todas as atividades. Apenas a primeira atividade de um *workflow* é obrigatória. Podemos ter, portanto, para cada instância de *workflow*, um subconjunto qualquer de atividades executadas, com a execução obrigatória da atividade de identificação.

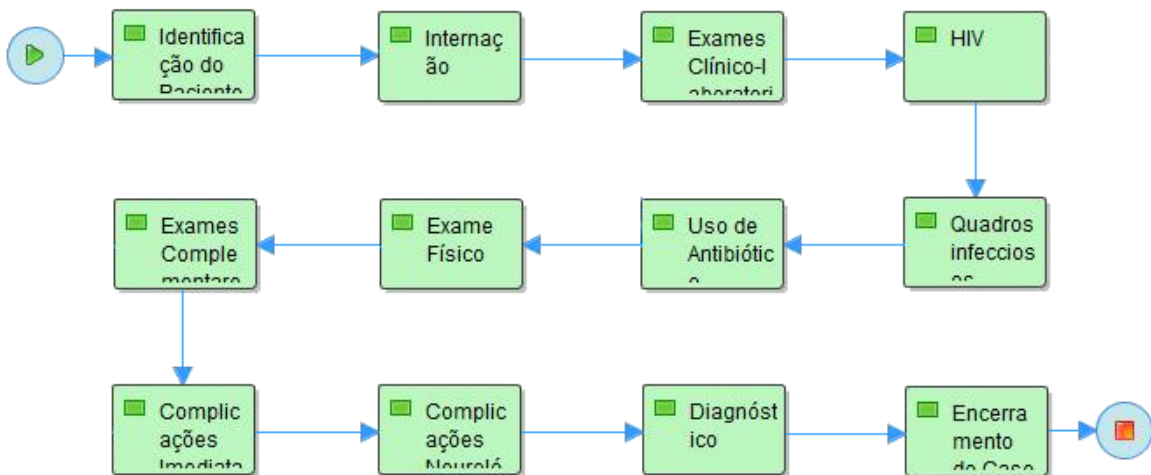


Figura 5.4. *Workflow* de pesquisa de Meningite

Os relatórios construídos para essa dissertação, relacionados ao estudo da doença Meningite, foram testados com dados fictícios. Para ilustrar a capacidade de busca do SiglaSearch, seguem abaixo exemplos de consultas para a pesquisa da doença Meningite:

1. Relação de casos de meningite por escolaridade e por estado da federação: Esse relatório pode ser obtido com a seleção dos campos de escolaridade e de estado da federação da atividade identificação e campo sumariado com a contagem

de casos. Esse último campo, de contagem de casos, pode ser obtido com a aplicação de função de contagem em qualquer campo, pois a função de contagem representa a contagem de registros para cada combinação de valores dos campos simples (sem sumarização) selecionados. Deve-se também filtrar o diagnóstico confirmatório da atividade diagnóstico, para buscar apenas os casos confirmados de meningite. Basta filtrar os diagnósticos confirmatórios que possuem a palavra *meningite*. A Figura 5.5 mostra os resultados desse relatório. Cada combinação de escolaridade e de estado da federação que possua pelo menos um caso de meningite é apresentada no relatório com a respectiva quantidade de casos. Pode-se citar, por exemplo, que entre as pessoas com ensino fundamental de Minas Gerais foram identificados 3 casos, e entre as do Rio de Janeiro, 8 casos. Entre as pessoas com ensino superior de Minas Gerais foram listados 4 casos e do Rio de Janeiro, 1 caso. Se os casos registrados no sistema representassem todos os casos registrados na rede pública de saúde, poderíamos supor que houve um surto da doença no Rio de Janeiro (9 casos), comparando com a quantidade de casos dos demais estados. Poderíamos também levantar a questão se a doença poderia estar ligada com o grau de escolaridade da população. Ou pode ser que o maior número de casos ocorridos na população de menor escolaridade respeite as proporções da quantidade de pessoas em cada grau de estudo.

The screenshot shows a web application interface with a navigation bar at the top containing links like 'Tela Principal', 'Nova Atividade', 'Registros', 'Relatórios', 'Ferramentas', 'Administração', and 'Ajuda'. Below this is the 'Editor de Consultas' section, which includes a query editor on the left and a results table on the right. The query is titled 'Consulta: Meningite por Escolaridade e Estado'. The results table has three columns: 'Escolaridade', 'Estado', and 'Quantidade Casos'. The table lists 16 rows of data, showing the number of cases for each combination of education level and state. At the bottom of the interface, there are buttons for 'Exportar Tudo' and 'Exportar Página Atual', along with icons for exporting to various formats like XLS, PDF, CSV, and XML. The page number is 'Página 1 de 1' and the total number of results is '16 resultado(s)'.

Escolaridade	Estado	Quantidade Casos
Ensino Fundamental	Espírito Santo	1
Ensino Fundamental	Minas Gerais	3
Ensino Fundamental	Paraná	1
Ensino Fundamental	Rio de Janeiro	8
Ensino Fundamental	Rio Grande do Sul	4
Ensino Fundamental	Santa Catarina	2
Ensino Fundamental	São Paulo	1
Ensino Superior	Espírito Santo	1
Ensino Superior	Minas Gerais	4
Ensino Superior	Paraná	3
Ensino Superior	Rio de Janeiro	1
Ensino Superior	Rio Grande do Sul	1
Ensino Superior	Santa Catarina	1
Ensino Superior	São Paulo	1
Nenhuma	Espírito Santo	1
Nenhuma	Santa Catarina	1

Figura 5.5. Consulta de casos de meningite por escolaridade e por estado da federação

2. Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos gerados para os casos ocorridos a partir do ano de 2013, com ordenação dos resultados por antibiótico: Essa consulta é gerada com filtro da data de admissão maior ou igual a 01/01/2013 da atividade internação, seleção do campo antibiótico da atividade antibiótico, do campo de complicações supurativas da atividade complicações imediatas, do campo de complicações neurológicas da atividade complicações neurológicas e dos campos diagnóstico presuntivo e diagnóstico confirmatório da atividade diagnóstico. A Figura 5.6 mostra os resultados desse relatório. Através da análise desse relatório podem ser analisadas a utilização dos antibióticos e as complicações geradas. Também podem ser observadas as relações entre os diferentes tipos de antibióticos utilizados para cada diagnóstico confirmatório. Esse é um exemplo de relatório analítico onde são mostrados todos os detalhes de cada caso. A partir desse relatório podem ser facilmente derivados outros relatórios sintéticos, com apresentação resumida dos dados, para análise da contagem de casos de aplicações dos antibióticos para cada diagnóstico, comparação entre as complicações e os diagnósticos e a contagem de ocorrência de cada antibiótico aplicado relacionado a cada tipo de complicação supurativa ou neurológica.

Antibiótico	Compl. Supurativas	Compl. Neurológicas	Diag. confirmatório	Diag. presuntivo
amoxicilina	bastonetes gram-positivo	Outras	meningite pneumocócica	meningite pneumocócica
amoxicilina			sepsis	pneumonia
amoxicilina	bastonetes gram-positivo		meningite bacteriana	epilepsia
amoxicilina	Empiema subdural	Outras	meningite meningocócica	meningite bacteriana
amoxicilina	bastonetes gram-positivo	Flebites e tromboflebites	meningite bacteriana	meningite bacteriana
amoxicilina	Empiema subdural	Flebites e tromboflebites	meningite viral	encefalite viral
amoxicilina	Empiema subdural		meningite meningocócica	meningite
amoxicilina	Empiema subdural	Outras	encefalite	encefalite viral
amoxicilina	bastonetes gram-positivo		encefalite viral	encefalite viral
ampicilina			febre	febre hemorrágica
ampicilina	bastonetes gram-positivo	Flebites e tromboflebites	meningite viral	crise convulsiva por encefalite viral
ampicilina	bastonetes gram-positivo		infecção das vias aéreas superiores	meningite bacteriana
ampicilina	Empiema subdural	Flebites e tromboflebites	gastroenterite	meningite meningocócica
ampicilina	convulsão febril	encefalite		
ampicilina	Empiema subdural	Outras	meningite meningocócica	meningite bacteriana
ampicilina			meningite meningocócica	meningite bacteriana
ampicilina	bastonetes gram-positivo	Outras	encefalite	meningoencefalite viral
ampicilina	bastonetes gram-positivo	Flebites e tromboflebites	otite aguda	meningite
ampicilina	bastonetes gram-positivo		encefalite	meningite
ampicilina			doença de kawasaki	encefalite bacteriana kawasaki

Figura 5.6. Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos gerados

3. Análise da relação de HIV com as sequelas de paralisia cerebral, retardo mental, hidrocefalia, surdez e cegueira para os casos ocorridos a partir do ano de 2013:

Essa consulta pode ser feita com a busca dos campos HIV na atividade HIV e dos campos paralisia cerebral, retardo mental, hidrocefalia, surdez e cegueira da atividade de encerramento do caso e com filtro da data de admissão maior ou igual a 01/01/2013 da atividade internação. A Figura 5.7 exibe os resultados desse relatório. Através dele pode-se analisar se a ocorrência de meningite em pacientes portadores de HIV pode agravar ou aumentar os casos de sequelas.

Consulta: Meningite - HIV e Sequelas

HIV	Paralisia Cerebral	Retardo Mental	Surdez	Hidrocefalia	Cegueira	Admissão
Sim						31/08/2013
Sim	Sim					17/08/2013
Sim			Sim			12/08/2013
Sim	Sim		Sim	Sim		27/05/2013
Sim			Sim			04/01/2014
Sim	Sim			Sim		04/03/2014
Sim	Sim					18/04/2015
Sim						19/08/2013
Sim			Sim			13/12/2014
Não						18/08/2015
Não						05/04/2013
Não	Sim	Sim	Sim			04/08/2014
Não	Sim	Sim	Sim			25/08/2014
Não						11/08/2014
Não		Sim	Sim			04/08/2014
Não	Sim	Sim	Sim			06/08/2013
Não	Sim	Sim				22/08/2014
Não						11/08/2014
Não						04/08/2014
Não			Sim			05/02/2014

Página 1 de 3 53 resultado(s).

Figura 5.7. Análise da relação de HIV com as sequelas

5.3 Paracoccidioidomicose (PCM)

Fernandes [2012] analisou os dados da doença Paracoccidioidomicose (PCM), micose causada pelo fungo *Paracoccidioides brasiliensis*, de pacientes diagnosticados com tal doença. O contágio ocorre com inalação do fungo e posterior disseminação para outros órgãos e tecidos do corpo humano. Os dados foram cedidos pelo Centro de Treinamento e Referência de Doenças Infecto-Parasitárias (CTR-DIP) do HC/UFGM, coordenado pelo doutor Ênio Pietra. Eles compõem uma base de dados com centenas de variáveis clínicas avaliadas pelos médicos para cada paciente. A base contém dados de 222 pacientes com análise de 314 parâmetros.

A análise é feita através da ferramenta de mineração de dados BioBI, de apoio aos pesquisadores para análise de dados biológicos e clínicos. Antes da análise, foi necessária a criação do armazém de dados com o modelo dimensional específico modelado

para essa doença. Através do modelo, é possível realizar consultas OLAP (*On-Line Analytical Processing* - Processamento Analítico On-Line) e mineração de dados.

O armazém de dados apresentado realiza funções que não são possíveis nos bancos de dados relacionais tradicionais. Entre elas estão a geração de certos dados estatísticos, como distribuição de frequência da presença de cada atributo e formação de *clusters* com conjuntos de pacientes com características similares. No entanto, uma série de consultas estatísticas levantadas pelo BioBI podem ser realizadas pelo SiglaSearch, uma vez que o *workflow* com os dados esteja carregado no SIGLa. A geração desses relatórios estatísticos no SiglaSearch facilita a análise de informações pelo usuário que pesquisa a doença, pois não carece de envolvimento de pessoal especializado em computação nem de custos de tempo e de recursos para a geração de um armazém de dados específicos para estudo da doença PCM. Aqui citamos três relatórios que podem ser realizados no próprio SiglaSearch, apresentados no trabalho de Fernandes [2012], listados abaixo.

1. Relatório da relação entre tabagismo prévio com o tempo de evolução da doença e com a quantidade de casos: São analisadas as ocorrências de PCM com presença ou ausência de lesão cutânea. No SiglaSearch, para a geração desse relatório, basta selecionar os campos de tabagismo prévio, de lesão cutânea, um campo sumarizado com a média do tempo de evolução da doença e outro campo sumarizado para a contagem de casos. A Figura 5.8 exibe os resultados desse relatório. Podemos verificar pela análise do relatório que o tempo de evolução da doença é muito maior em pacientes com tabagismo prévio, tanto nos casos de presença quanto de ausência de lesão cutânea. Com presença de lesão cutânea, o tempo de evolução da doença em pacientes com tabagismo prévio é 2,41 vezes maior (18,21 meses em tabagistas contra 5,79 meses em não tabagistas). Com ausência de lesão cutânea, o tempo de evolução da doença em pacientes com tabagismo prévio é 3,16 vezes maior (20,85 meses em tabagistas contra 8,67 meses em não tabagistas). Por esse relatório obtemos também a quantidade de casos entre os pacientes tabagistas e não tabagistas, com presença e ausência de lesão cutânea. Os campos nulos no relatório, sobre tabagismo e lesão cutânea, representam os casos em que a informação não foi registrada.
2. Análise de vômitos em pacientes com PCM: O relatório mostra a presença ou a ausência de vômito, média de idade na primeira consulta, contagem do fato e percentagem do fato sobre o total. O SiglaSearch apresenta esse relatório com a seleção do campo vômito, um campo sumarizado com a função média aplicada ao campo de idade na primeira consulta e uma função sumarizada para contagem do fato. O SiglaSearch não oferece toda a expressividade do relatório do armazém

Consulta: PCM - Tabagismo Prévio e Evolução Doença

Lista de Resultados da Consulta

Tabagismo Prévio	Lesão Cutânea	Média Tempo Evolução	Quantidade Casos
	Não	9.86	14
	Sim	14.18	17
Não		0.5	2
Não	Não	8.67	24
Não	Sim	5.79	19
Sim		15.0	2
Sim	Não	20.85	92
Sim	Sim	18.21	48

Página 1 de 1
8 resultado(s).

Figura 5.8. Relatório da relação entre tabagismo prévio com o tempo de evolução da doença e com a quantidade de casos

de dados pela incapacidade de gerar o campo de porcentagem sobre o total, mas esse campo pode ser extraído através da razão entre a contagem do fato e o total de pacientes. A Figura 5.9 mostra os resultados desse relatório. Através desse relatório, podemos inferir que a presença de vômito acelerou o diagnóstico da doença, pois a média de idade da primeira consulta é menor dentre os casos com presença de vômito. A média de idade da primeira consulta é de 31,73 anos em casos com presença de vômito contra 37,78 anos entre os casos com ausência de vômito. Podemos observar que a presença de vômito entre os casos também é pouco frequente, sendo 33 casos com presença de vômito contra 170 com ausência de vômito, representando apenas 16,26% sobre o total de casos.

Consulta: PCM - Relação de vômitos entre os casos

Lista de Resultados da Consulta

Vômitos	Média Idade Primeira Consulta	Quantidade Casos
Não	37.78	170
Sim	31.73	33

Página 1 de 1
2 resultado(s).

Figura 5.9. Análise de vômitos em pacientes

3. Relação do diagnóstico por gênero, onde se nota que a maioria dos diagnósticos foi para o gênero masculino: Nesse relatório são mostrados o gênero, a contagem do

fato, a porcentagem sobre o total, a média de idade na primeira consulta e o tempo médio de evolução. No SiglaSearch pode-se gerar esse relatório selecionando o campo gênero, um campo sumarizado com a contagem do fato, um campo sumarizado com a função média aplicada ao campo de idade da primeira consulta e um campo sumarizado com a função média aplicada ao campo de tempo de evolução da doença. A Figura 5.10 exibe os resultados desse relatório. Podemos observar nesse relatório que a doença é mais frequente entre homens. São 181 casos para o gênero masculino contra 41 do feminino, o que representa 81,53% sobre o total. O tempo médio de evolução da doença é menor em mulheres. Entre mulheres o tempo médio é de 10,02%, enquanto entre homens o tempo médio é de 17,40%. O diagnóstico da doença é mais precoce nas mulheres, pois a média de idade da primeira consulta entre mulheres é de 26,83 anos e de 39,04 anos entre homens. Isso pode ser reflexo do maior cuidado com a própria saúde entre a população feminina comparada com a masculina.

Gênero	Quantidade Casos	Média Idade Primeira Consulta	Tempo Médio Evolução
Feminino	41	26.83	10.02
Masculino	181	39.04	17.4

Página 1 de 1
2 resultado(s).

Figura 5.10. Relação de diagnósticos por gênero

Capítulo 6

Conclusão

Nesse projeto foi possível trabalhar com problemas de arquitetura de banco de dados, criação de interface para usuários, geração automática de relatórios e aplicação da tecnologia da informação em um contexto de experimentação científica.

A interface criada para consulta de bancos de dados de experimentos, apresentada nesse trabalho, possui uma interface que procura fornecer boa usabilidade. Ainda que os usuários não tenham experiência em programação, podem utilizar um sistema de consulta para obtenção dos resultados em poucos minutos. O sistema também prevê visões diferentes para diferentes perfis, onde usuários comuns executam os relatórios relacionados às suas atividades e apenas os administradores do sistema editam as consultas. O SiglaSearch possui interface intuitiva e permite o desenvolvimento de consultas com uma curva de aprendizado rápida. Dessa forma, ele promove o estreitamento da distância entre a tecnologia da informação e profissionais de outras áreas do conhecimento como a medicina, a biologia e a física.

Esse projeto apresenta uma especificação e uma implementação de sistema para consulta de dados em *workflow*, através de um conjunto de dados estruturados. Não foram encontrados trabalhos dessa natureza e a única pesquisa disponível em *workflow* é a pesquisa por palavras-chave em dados não estruturados, como textos e arquivos anexos. A consulta de dados estruturados em *workflow* trabalhada nessa dissertação representa uma melhoria para a comunidade científica que trabalha com *workflow*. E também representa uma melhoria do SIGLa como LIMS, que passa a ter um sistema de consulta altamente flexível, complementar à alta flexibilidade do armazenamento de dados já existente.

6.1 Trabalhos Futuros

Como trabalhos futuros, pode-se citar:

1. A criação de junção de diferentes *workflows* em uma mesma consulta, para que os dados de diferentes *workflows* possam ser relacionados, como o relacionamento de um *workflow* de transformação genética de sementes com outro *workflow* que gerencia a plantação e a colheita dos frutos gerados das plantações com essas sementes, por exemplo.
2. A combinação de consultas também é importante para a realização de operações com conjuntos, o que pode também significar uma aplicação alternativa e mais intuitiva de filtros de disjunção, onde apenas uma das condições é satisfeita, com a união de dois resultados. A persistência das estruturas das consultas contribui para a criação dessa combinação de consultas, pois basta a criação de uma interface que escolha estruturas de consultas salvas e verifique se a quantidade de campos e domínio dos dados de cada campo é compatível com o campo respectivo das demais consultas. As três operações de conjunto possíveis para banco de dados relacional são a união de resultados, a interseção de resultados e a diferença entre dois resultados. Um exemplo de combinação de resultados seria a interseção entre os resultados de pacientes que tomaram a vacina para a gripe em um período de um ano com os resultados dos pacientes que apresentaram pneumonia após esse período, para avaliar a eficácia da vacina.
3. A otimização das consultas dinâmicas pode ser realizada através das verificações dos planos de execução da consulta, mudança da estrutura de algumas das tabelas, inclusões de novos índices de pesquisa e pré-processamento de parte das fases de consulta.
4. Aumento da expressividade da consulta ou a melhoria da usabilidade através de testes de usabilidade com o usuário. Para aumentar a expressividade da consulta, deve-se avaliar se o benefício do novo recurso é maior que a complexidade que pode ser adicionada ao sistema.
5. Definição de grupos de usuários que podem acessar cada relatório salvo no editor de consulta.

Referências Bibliográficas

- Al-Jadir, L.; Parent, C. & Spaccapietra, S. (2010). Reasoning with large ontologies stored in relational databases: The ontomind approach. *Data & Knowledge Engineering*, 69(11):1158--1180.
- Allan, C.; Burel, J.-M.; Moore, J.; Blackburn, C.; Linkert, M.; Loynton, S.; MacDonald, D.; Moore, W. J.; Neves, C.; Patterson, A. et al. (2012). Omero: flexible, model-driven data management for experimental biology. *Nature methods*, 9(3):245-253.
- Altintas, I.; Berkley, C.; Jaeger, E.; Jones, M.; Ludascher, B. & Mock, S. (2004). Kepler: an extensible system for design and execution of scientific workflows. Em *Scientific and Statistical Database Management, 2004. Proceedings. 16th International Conference on*, pp. 423--424. IEEE.
- Altschull, S. F.; Gish, W.; Miller, W.; Myers, E. W. & Lipman, D. J. (1990). Basic local alignment search tool. *J. Mol. Biol.*, 215:4037410.
- Angles, R. & Gutierrez, C. (2008). Survey of graph database models. *ACM Computing Surveys (CSUR)*, 40(1):1.
- Attorre, B. (2014). Javaserwer faces (jsf): Melhorando a qualidade da aplicação web. Acessado em 17 de Junho de 2015.
- Batra, S. & Tyagi, C. (2012). Comparative analysis of relational and graph databases. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(2):509--512.
- Benyon, D. (2011). *Interação humano-computador*. São Paulo: Pearson Prentice Hall.
- BioPortal (2015). Bioportal. Acessado em 16 de Junho de 2015.
- Bornea, M. A.; Dolby, J.; Kementsietsidis, A.; Srinivas, K.; Dantressangle, P.; Udrea, O. & Bhattacharjee, B. (2013). Building an efficient rdf store over a relational

- database. Em *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pp. 121--132. ACM.
- Brice, J.; Phelps, J. N. & Kennedy, S. L. (2013). Configurable search graphical user interface and engine. US Patent 8,510,330.
- Britton, C. P.; Kumar, A.; Bigwood, D.; DeFusco, A. J. & Greenblatt, H. (2013). Methods and apparatus for querying a relational data store using schema-less queries. US Patent App. 13/851,654.
- Buda, T. S.; Murphy, J. & Kristiansen, M. (2013). Towards realistic sampling: generating dependencies in a relational database.
- Buil-Aranda, C.; Arenas, M.; Corcho, O. & Polleres, A. (2013). Federating queries in sparql 1.1: Syntax, semantics and evaluation. *Web Semantics: Science, Services and Agents on the World Wide Web*, 18(1):1 – 17. Special Section on the Semantic and Social Web.
- Chen, Z. & Coppé, J.-P. (2012). Method and system for building and using a centralized and harmonized relational database. US Patent App. 13/423,458.
- Civici, Ç. & DARÇIN, Y. (2015). Primefaces users guide.
- Cohen-Boulakia, S. & Leser, U. (2011). Search, adapt, and reuse: the future of scientific workflows. *ACM SIGMOD Record*, 40(2):6--16.
- Computing Solutions, I. (2015). <http://www.labsoftlims.com>. Acessado em 16 de Junho de 2015.
- Consortium, G. O. et al. (2001). Creating the gene ontology resource: design and implementation. *Genome research*, 11(8):1425--1433.
- De Souza, C. S. (2005). *The semiotic engineering of human-computer interaction*. MIT press.
- Deelman, E.; Gannon, D.; Shields, M. & Taylor, I. (2009). Workflows and e-science: An overview of workflow system features and capabilities. *Future Generation Computer Systems*, 25(5):528--540.
- Editor, T. X. W. (2015). <http://www.together.at/prod/workflow/twe>. Acessado em 16 de Junho de 2015.

- Elmasri, R. & Navathe, S. (2011). *Sistemas de banco de dados*. São Paulo: Pearson Addison-Wesley.
- Estevam dos Santos, P. (2010). Boas praticas de laboratorio (bpl) uma questão de qualidade. *Revista Intertox de Toxicologia, Risco Ambiental e Sociedade*, 3.
- Fernandes, H. R. (2012). *BioBI - Ferramenta para Analise de Dados Biologicos*. Dissertação (Mestrado em Ciências da Computação) - Departamento de Ciências da Computação, Universidade Federal de Minas Gerais, Belo Horizonte.
- Fiocruz (2015). Fiocruz. Acessado em 16 de Junho de 2015.
- Fischer, S.; Aurrecochea, C.; Brunk, B. P.; Gao, X.; Harb, O. S.; Kraemer, E. T.; Pennington, C.; Treatman, C.; Kissinger, J. C.; Roos, D. S. et al. (2011). The strategies wdk: a graphical search interface and web development kit for functional genomics databases. *Database*, 2011:bar027.
- Foundation, T. A. S. (2015). <http://tomcat.apache.org/>. Acessado em 16 de Junho de 2015.
- Goecks, J.; Nekrutenko, A.; Taylor, J. et al. (2010). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biol*, 11(8):R86.
- Goodman, N.; Rozen, S.; Stein, L. D. & Smith, A. G. (1998). The labbase system for data management in large scale biology research laboratories. *Bioinformatics*, 14(7):562--574.
- Guo, A. C.; Jewison, T.; Wilson, M.; Liu, Y.; Knox, C.; Djoumbou, Y.; Lo, P.; Mandal, R.; Krishnamurthy, R. & Wishart, D. S. (2013). Ecmdb: the e. coli metabolome database. *Nucleic acids research*, 41(D1):D625--D630.
- Han, J.; Haihong, E.; Le, G. & Du, J. (2011). Survey on nosql database. Em *Pervasive computing and applications (ICPCA), 2011 6th international conference on*, pp. 363-366. IEEE.
- Hanke, L. A.; Botelho, C. S.; Braz, F. A.; Batista, P. H.; Folgueras-Flatschart, A. V.; Noda, R. W.; Carneiro, A. A.; Faria-Campos, A. C. & Campos, S. V. (2014). Flux-transgenics: a flexible lims-based tool for management of plant transformation experimental data. *Plant Methods*, 10(1):20.

- Hanke, L. A. F. (2014). *Desenvolvimento e Implementação de um Sistema de Controle de Acesso Flexível para LIMS baseados em Workflows*. Monografia (Graduação em Engenharia de Controle e Automação) - Escola de Engenharia, Universidade Federal de Minas Gerais, Belo Horizonte.
- He-ping, C.; Lu, H. & Bin, C. (2008). Research and implementation of ontology automatic construction based on relational database. Em *Computer Science and Software Engineering, 2008 International Conference on*, volume 5, pp. 1078--1081. IEEE.
- Hensley, M. R.; Hassenplug, E.; McPhail, R. & Leung, Y. F. (2012). Zebase: an open-source relational database for zebrafish laboratories. *Zebrafish*, 9(1):44--49.
- Hill, C.; Gordon, I. E.; Rothman, L. S. & Tennyson, J. (2013). A new relational database structure and online interface for the hitran database. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 130:51--61.
- Hinton, M. D. (1995). *Laboratory Information Management Systems: development and implementation for a quality assurance laboratory*. MARCEL DEKKER, New York, New York.
- Hull, D.; Wolstencroft, K.; Stevens, R.; Goble, C.; Pocock, M. R.; Li, P. & Oinn, T. (2006). Taverna: a tool for building and running workflows of services. *Nucleic acids research*, 34(suppl 2):W729--W732.
- Jain, V. & Singh, M. (2013). A framework to convert relational database to ontology for knowledge database in semantic web. *International Journal of Scientific & Technology Research*, 2(10).
- Jarke, M. & Vassiliou, Y. (1985). A framework for choosing a database query language. *ACM Computing Surveys (CSUR)*, 17(3):313--340.
- Jin, Y.; Li, L.; Wang, L. L. & Zhang, W. C. (2014). Input/output efficiency for online analysis processing in a relational database. US Patent 8,719,312.
- Krause, W. (2014). *ISO 21500-Orientações sobre Gerenciamento de Projetos: Diretrizes para o Sucesso*. Brasport.
- Kumar, C. G.; LeDuc, R.; Gong, G.; Roinishivili, L.; Lewin, H. A. & Liu, L. (2004). Estima, a tool for est management in a multi-project environment. *BMC bioinformatics*, 5(1):176.

- LabVantage Solutions, I. (2015). <http://www.sqllims.com>. Acessado em 16 de Junho de 2015.
- Lana-Peixoto, M. A.; Talim, L. E.; Faria-Campos, A. C.; Campos, S. V.; Rocha, C. F.; Hanke, L. A.; Talim, N.; Batista, P. H.; Araujo, C. R. & Kleinpaul, R. (2011). Nmo-dbr: the brazilian neuromyelitis optica database system. *Arquivos de neuro-psiquiatria*, 69(4):687--692.
- Leavitt, N. (2010). Will nosql databases live up to their promise? *Computer*, 43(2):12-14.
- Lee, K. K.-Y.; Tang, W.-C. & Choi, K.-S. (2013). Alternatives to relational database: comparison of nosql and xml approaches for clinical data storage. *Computer methods and programs in biomedicine*, 110(1):99--109.
- Lee, W. & Markley, J. L. (2012). Pacsy database, a relational database management system for protein structure and nuclear magnetic resonance chemical shift analysis. Em *Bioinformatics and Biomedicine Workshops (BIBMW), 2012 IEEE International Conference on*, pp. 930--932. IEEE.
- Li, H.; Gennari, J. H. & Brinkley, J. F. (2006). Model driven laboratory information management systems. Em *AMIA Annual Symposium Proceedings*, volume 2006, p. 484. American Medical Informatics Association.
- Liu, Z.; Shao, Q. & Chen, Y. (2010). Searching workflows with hierarchical views. *Proceedings of the VLDB Endowment*, 3(1-2):918--927.
- Loney, K. (2008). *Oracle Database 11g The Complete Reference*. McGraw-Hill, Inc.
- Los, R. K.; van Ginneken, A. M. & van der Lei, J. (2005). Extracting data recorded with opensde: possibilities and limitations. *International journal of medical informatics*, 74(6):473--480.
- Maguire, E.; Rocca-Serra, P.; Sansone, S.-A.; Davies, J. & Chen, M. (2013). Visual compression of workflow visualizations with automated detection of macro motifs. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):2576--2585.
- Mednieks, Z.; Dornin, L.; Meike, B. & Nakamura, M. (2012). *Programando o Android*. São Paulo: Novatec Editora.
- Mendonça, L. d. M. L. (1999). *Uma Interface Grafica para Formulação de Consultas Incompletas a Bancos de Dados Relacionais*. Dissertação (Mestrado em Ciências da

- Computação) - Departamento de Ciências da Computação, Universidade Federal de Minas Gerais, Belo Horizonte.
- Murray, C. (2008). Oracle database semantic technologies developer's guide.
- Murray, G. A. & Crocker, D. P. (2011). Applying open-source software to laboratory data management. *Journal of the Association for Laboratory Automation*, 16(5):327-334.
- MySQL (2015). <http://www.mysql.com/>. Acessado em 16 de Junho de 2015.
- Nadkarni, P. M. (1998). Data extraction and ad hoc query of an entity-attribute-value database. *Journal of the American Medical Informatics Association*, 5(6):511--527.
- NCBI (2015). <http://www.ncbi.nlm.nih.gov/>. Acessado em 16 de Junho de 2015.
- Nelson, E. K.; Piehler, B.; Eckels, J.; Rauch, A.; Bellew, M.; Hussey, P.; Ramsay, S.; Nathe, C.; Lum, K.; Krouse, K. et al. (2011). Labkey server: an open source platform for scientific data integration, analysis and collaboration. *BMC bioinformatics*, 12(1):71.
- Nieuwland, J.; Sornay, E.; Marchbank, A.; de Graaf, B. & Murray, J. (2012). Phytotracker, an information management system for easy recording and tracking of plants, seeds and plasmids. *Plant methods*, 8(1):43.
- Oinn, T.; Greenwood, M.; Addis, M.; Alpdemir, M. N.; Ferris, J.; Glover, K.; Goble, C.; Goderis, A.; Hull, D.; Marvin, D. et al. (2006). Taverna: lessons in creating a workflow environment for the life sciences. *Concurrency and Computation: Practice and Experience*, 18(10):1067--1100.
- Ouyang, D.-T.; Cui, X.-J. & Ye, Y.-X. (2010). Mapping integrity constraint ontology to relational databases. *The Journal of China Universities of Posts and Telecommunications*, 17(6):113--121.
- Palla, P.; Frau, G.; Vargiu, L. & Rodriguez-Tomé, P. (2012). Qtreds: a flexible lms for omics laboratories. *EMBnet. journal*, 18(B):pp--41.
- Prates, R. O. & Barbosa, S. D. J. (2007). Introdução à teoria e prática da interação humano computador fundamentada na engenharia semiótica. *Atualizações em informática*, pp. 263--326.
- Preece, J.; Sharp, H. & Rogers, Y. (2015). *Interaction Design-beyond human-computer interaction*. John Wiley & Sons.

- Queiroz, O. V. d. (2009). *Relacionamento Probabilístico de Registros na Integração de Sistemas de Informação do SUS: O caso da Base Nacional de Dados em terapia renal substitutiva*. Dissertação (Mestrado) - Programa de Pós-Graduação em Saúde Pública, Universidade Federal de Minas Gerais, Belo Horizonte.
- Robinson, S. W.; Herzyk, P.; Dow, J. A. & Leader, D. P. (2013). Flyatlas: database of gene expression in the tissues of *Drosophila melanogaster*. *Nucleic acids research*, 41(D1):D744--D750.
- Saccone, S. F.; Quan, J.; Mehta, G.; Bolze, R.; Thomas, P.; Deelman, E.; Tischfield, J. A. & Rice, J. P. (2011). New tools and methods for direct programmatic access to the dbSNP relational database. *Nucleic acids research*, 39(suppl 1):D901--D907.
- Santoso, H. A.; Haw, S.-C. & Abdul-Mehdi, Z. T. (2011). Ontology extraction from relational database: Concept hierarchy as background knowledge. *Knowledge-Based Systems*, 24(3):457--464.
- Schaab, C.; Geiger, T.; Stoehr, G.; Cox, J. & Mann, M. (2012). Analysis of high accuracy, quantitative proteomics data in the MaxQB database. *Molecular & Cellular Proteomics*, 11(3):M111--014068.
- Schulte, O. & Khosravi, H. (2012). Learning graphical models for relational data via lattice search. *Machine Learning*, 88(3):331--368.
- Scott, D. J.; Lee, J.; Silva, I.; Park, S.; Moody, G. B.; Celi, L. A. & Mark, R. G. (2013). Accessing the public MIMIC-II intensive care relational database for clinical research. *BMC Medical Informatics and Decision Making*, 13(1):9.
- Selvakumar, R. (2010). Good laboratory practices. *Indian Journal of Clinical Biochemistry*, 25(3):221--224.
- Sequeda, J. F. & Miranker, D. P. (2013). Ultrawrap: SPARQL execution on relational data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 22:19--39.
- Shao, Q.; Sun, P. & Chen, Y. (2009). Wise: a workflow information search engine. In *Data Engineering, 2009. ICDE'09. IEEE 25th International Conference on*, pp. 1491--1494. IEEE.
- Silva, A. S.; Batista, P. H.; Andrade, T. K.; Neto, O. P. V.; Campos, S. & Faria-Campos, A. (2013). Nanotrack—sistema inteligente de gerenciamento de dados de síntese de nanoestruturas.

- Simões, A.; Faria Campos, A. C.; Delaat, D. M.; Abreu, V. & Campos, S. V. A. (2010). Sigla: An adaptable lims for multiple laboratories.
- Solutions, L. L. (2015). <http://www.labware.com/lwweb.nsf/rhome3.xsp>. Acessado em 16 de Junho de 2015.
- Stocker, G.; Fischer, M.; Rieder, D.; Bindea, G.; Kainz, S.; Oberstolz, M.; McNally, J. G. & Trajanoski, Z. (2009). ilap: a workflow-driven software for experimental protocol development, data acquisition and analysis. *BMC bioinformatics*, 10(1):390.
- Tolopko, A. N.; Sullivan, J. P.; Erickson, S. D.; Wrobel, D.; Chiang, S. L.; Rudnicki, K.; Rudnicki, S.; Nale, J.; Selfors, L. M.; Greenhouse, D. et al. (2010). Screensaver: an open source lab information management system (lims) for high throughput screening facilities. *BMC bioinformatics*, 11(1):260.
- Vijayprasath, S. & Rajan, S. P. (2015). Design of a simple graphical user interface to the relational database management system.
- Viksna, J.; Celms, E.; Opmanis, M.; Podnieks, K.; Rucevskis, P.; Zarins, A.; Barrett, A.; Neogi, S. G.; Krestyaninova, M.; McCarthy, M. I. et al. (2007). Passim—an open source software system for managing information in biomedical studies. *BMC bioinformatics*, 8(1):52.
- Waters, M.; Stasiewicz, S.; Merrick, B. A.; Tomer, K.; Bushel, P.; Paules, R.; Stegman, N.; Nehls, G.; Yost, K. J.; Johnson, C. H. et al. (2008). Cebs-chemical effects in biological systems: a public data repository integrating study design and toxicity data with microarray and proteomics data. *Nucleic acids research*, 36(suppl 1):D892-D900.
- Wendl, M. C.; Smith, S.; Pohl, C. S.; Dooling, D. J.; Chinwalla, A. T.; Crouse, K.; Hepler, T.; Leong, S.; Carmichael, L.; Nhan, M. et al. (2007). Design and implementation of a generalized laboratory data model. *BMC bioinformatics*, 8(1):362.
- WfMC (2015). <http://www.wfmc.org/>. Acessado em 16 de Junho de 2015.
- Wild, D. J.; Ding, Y.; Sheth, A. P.; Harland, L.; Gifford, E. M. & Lajiness, M. S. (2012). Systems chemical biology and the semantic web: what they mean for the future of drug discovery research. *Drug discovery today*, 17(9):469--474.
- Wilson, E. (2003). Search engine result reporter. US Patent App. 10/539,851.

Yang, X.; Procopiuc, C. M. & Srivastava, D. (2011). Summary graphs for relational database schemas.

Zeng, Z.; Bao, Z.; Ling, T. W. & Lee, M. L. (2012). isearch: an interpretation based framework for keyword search in relational databases. Em *Proceedings of the Third International Workshop on Keyword Search on Structured Data*, pp. 3--10. ACM.

Anexo A

Manual do usuário

O SiglaSearch é um módulo de consulta criado para atender a todos os usuários do SIGLa. Ele possui três partes importantes: a edição de consultas, a geração de relatórios e a administração das consultas. Você poderá acessar o Editor de Consultas do SiglaSearch se for um administrador do sistema. Esse editor permite que você acesse qualquer conjunto de informações de um *workflow*, com a execução de poucos passos. O sistema disponibiliza um assistente de consulta para obtenção dos primeiros resultados. Em seguida, pode-se usar as outras funções do editor, para refinar a consulta e adicionar, excluir ou alterar itens selecionados, filtros ou critérios de ordenação. As consultas podem ser salvas e abertas posteriormente, para novas alterações. Você pode salvar a consulta como um relatório com os parâmetros adequados, para que esse relatório seja disponibilizado em um menu de relatórios. O menu de relatórios atende aos usuários especialistas em determinados ramos da pesquisa gerenciada pelo *workflow*. Ao escolher um relatório do menu, o sistema exibe a tela de relatório para que você entre com os parâmetros, se necessários, e gere os relatórios. Além da edição de consultas e da execução de relatórios, o SiglaSearch possui o módulo de administração, para você, caso seja administrador(a) do sistema, gerencie as consultas e os relatórios disponíveis no SIGLa.

A.1 Entrada de Dados no SIGLa

Após os *workflows* serem carregados no SIGLa, eles ficam prontos para serem utilizados. Para cada *workflow* carregado, o sistema exibe as opções para criar nova instância do *workflow* ou alterar instâncias criadas. Uma instância executada de um *workflow* possui uma estrutura em árvore, onde cada nó da árvore representa uma atividade executada. A Figura A.1 ilustra um exemplo de tela para entrada de dados

de *workflow* no SIGLa. Você pode navegar pelas atividades selecionando o respectivo nó da árvore de atividades exibida no lado esquerdo da tela. Para cada atividade o sistema provê um formulário, à direita, com todos os atributos da atividade, para preenchimento e atualização dos dados ou para consulta aos dados gravados, conforme figura.

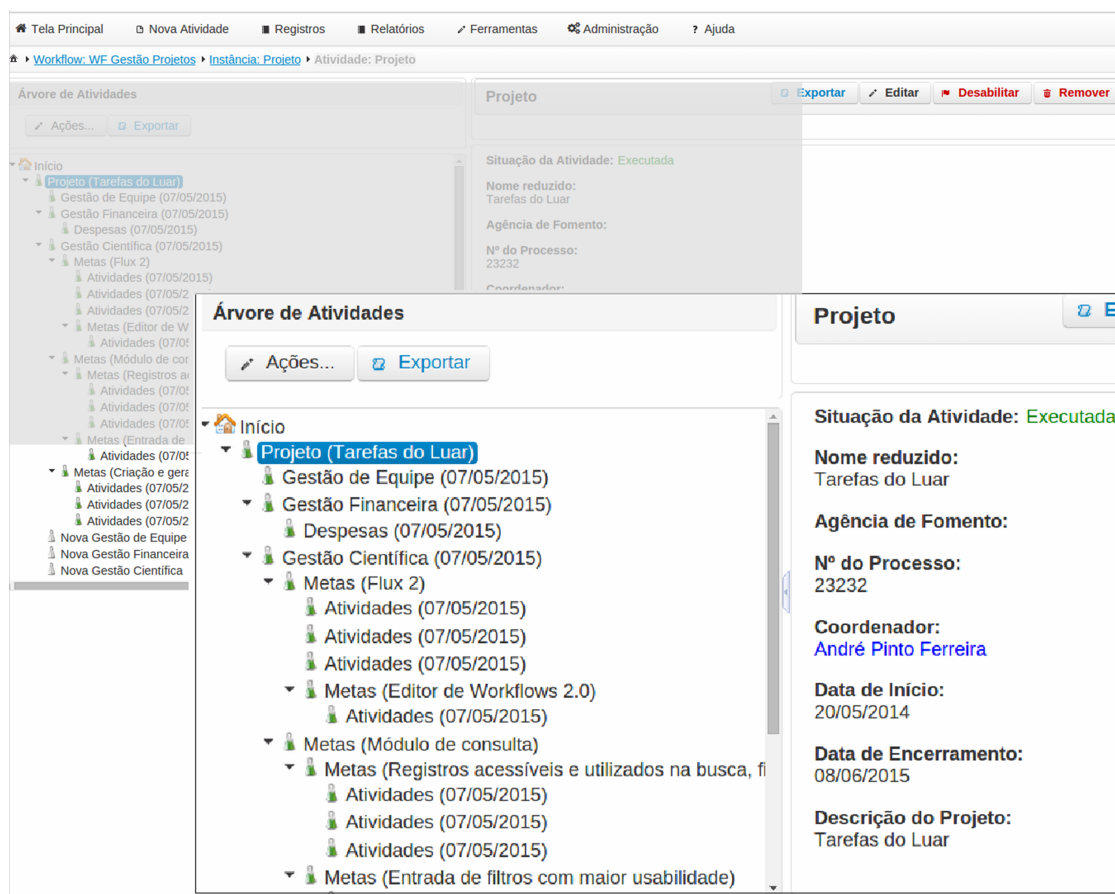


Figura A.1. Tela de entrada de dados de instância de *workflow* do SIGLa

A.2 Edição de Consulta

O editor de consulta é acessado por um menu restrito aos administradores do sistema. Através desse módulo é possível criar, editar e salvar consultas. O sistema permite que você, se for administrador(a) do sistema, escolha os campos a serem exibidos, que podem ser atributos de atividades ou campos de registros. Você também pode fazer sumarização dos dados de algum(s) dos campos selecionados em função dos demais campos. O sistema fornece suporte através de documentação, de instruções de introdução ao módulo de consulta, na tela inicial da aplicação, e durante o acesso

às funcionalidades, através da exibição de dicas e de alertas quanto a procedimentos incorretos. Esse conjunto de informações facilita o aprendizado rápido sobre o funcionamento do SiglaSearch, além da obtenção ágil dos dados procurados.

A.2.1 Tela principal

Quando você acessa o Editor de Consulta do SiglaSearch, o sistema exhibe as principais instruções para sua familiarização com a interface em pouco tempo. A Figura A.2 ilustra a tela principal com as primeiras instruções. Para cada instrução inicial, você pode acessar figuras ilustrativas ao passar o mouse sobre os botões de informação, para que o aprendizado seja mais intuitivo. Para criar nova consulta, o assistente de consulta deve ser executado obrigatoriamente, para obtenção dos primeiros resultados. As instruções iniciais são exibidas até que o assistente de nova consulta seja executado ou até que uma consulta salva seja aberta. Para executar o assistente, basta clicar no botão *Iniciar*.

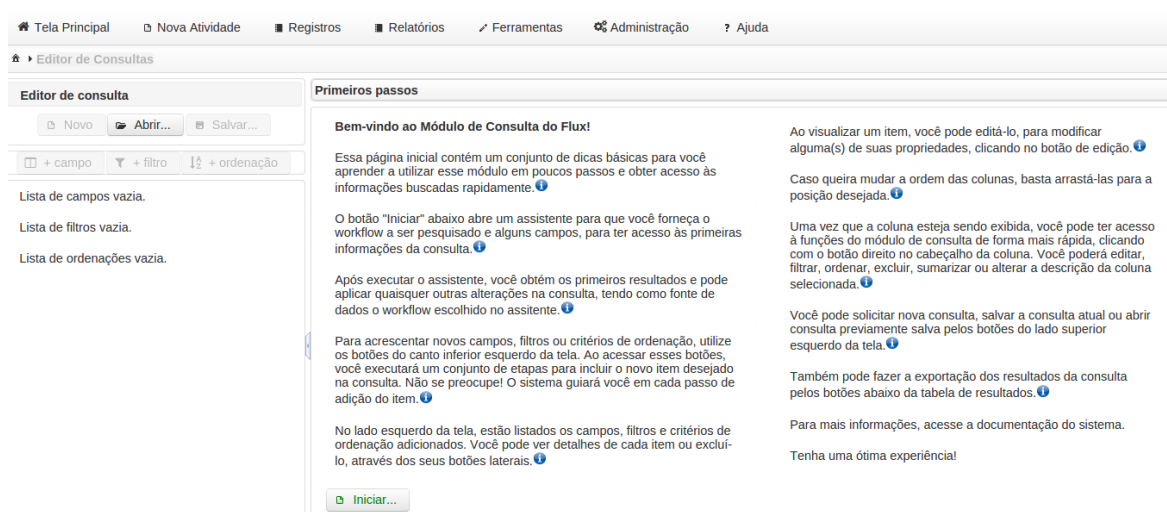


Figura A.2. Tela principal do editor de consulta do sistema SiglaSearch com as principais instruções

Após a execução do assistente, com os primeiros passos da consulta, ou após abrir uma consulta salva, o sistema passa a exhibir a tela com os resultados e habilita os botões com as funções de consulta (Figura A.3).

A tela principal habilita todos os recursos que estiverem disponíveis para cada cenário de uso. Na região superior esquerda há três botões. O primeiro botão, de rótulo *Novo*, permite criar uma nova consulta, voltando o sistema para situação inicial de execução dos primeiros passos. Quando você pede uma nova consulta com outra consulta não salva em desenvolvimento, o sistema confirma a operação, para prevenir

Figura A.3. Tela principal do editor de consultas do sistema SiglaSearch com exibição dos resultados

a perda de dados. O botão *Abrir* exibe a tela para abrir uma das consultas salvas (Figura A.4). E o botão *Salvar* abre a tela para salvar a consulta que estiver sendo editada (Figura A.5). Para disponibilizar a consulta em formato de relatório para todos os usuários do sistema, basta marcar opção de relatório ao salvar a consulta. Se a consulta não é salva como relatório, fica disponível apenas para os administradores do sistema.

Figura A.4. Tela para abrir consulta salva no sistema SiglaSearch

Na região superior esquerda da tela, logo abaixo dos botões *Novo*, *Abrir* e *Salvar*, há outros três botões. O primeiro, de rótulo *+ campo*, permite que você adicione um

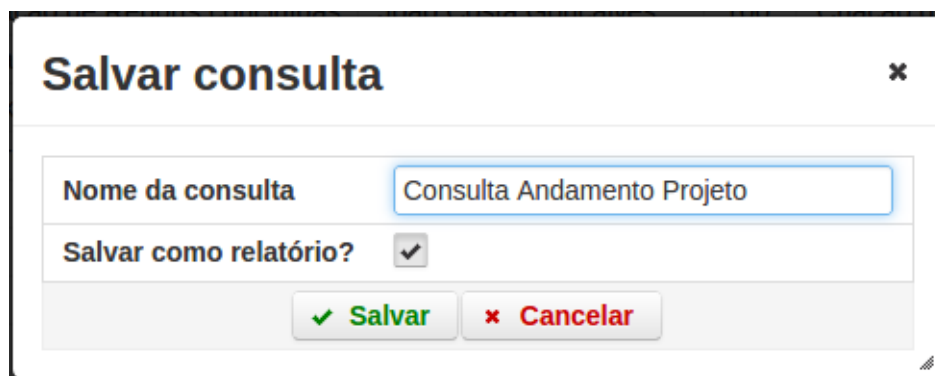
A screenshot of a web application dialog box titled "Salvar consulta" with a close button (x) in the top right corner. The dialog contains two input fields: "Nome da consulta" with the text "Consulta Andamento Projeto" and "Salvar como relatório?" with a checked checkbox. At the bottom, there are two buttons: "Salvar" with a green checkmark icon and "Cancelar" with a red x icon.

Figura A.5. Tela para salvar consulta no sistema SiglaSearch

novo campo de dados à consulta. O segundo botão, de rótulo *+filtro*, permite que você inclua novo filtro comparando um campo da consulta com um valor fixo ou com outro campo. O terceiro botão, de rótulo *+ordenação*, permite que você inclua novo critério de ordenação, com opção crescente ou decrescente. No lado esquerdo do editor são exibidos todos os itens adicionados à consulta, ou seja, os campos selecionados, os filtros e os critérios de ordenação. Na região central da tela, são mostrados os resultados gerados para a consulta, atualizados a cada adição ou alteração de item. Você pode exportar os dados em arquivos de formato Excel, CSV, PDF ou XML, através dos botões exibidos abaixo da tabela de resultados.

A.2.2 Criar nova consulta

Quando você acessa o menu do Editor de Consulta, o sistema é iniciado em modo de criação de nova consulta, exibindo as instruções iniciais. Durante o uso do editor, basta clicar no botão *Novo* para iniciar nova consulta. Em uma nova consulta, o assistente deve ser executado. Ele é acessado através do botão *Iniciar*. Nesse assistente você define o *workflow* a ser consultado e os primeiros atributos que desejar consultar. A escolha de *workflow* em uma consulta é feita uma única vez, pois toda a consulta será feita em um mesmo *workflow*. Após a escolha do *workflow*, você pode escolher sucessivas atividades e respectivos atributos a consultar. O fluxo de uso do assistente está esquematizado na Figura A.6.

As telas que compõem o assistente estão ilustradas na Figura A.7. O *workflow* é selecionado na tela da Figura A.7(a). Essa tela é exibida apenas uma vez a cada nova consulta. A atividade é escolhida na tela da Figura A.7(b). E a tela da Figura A.7(c) mostra a lista de campos a serem selecionados, que pertençam à atividade selecionada no passo de escolha da atividade. Nessa última tela, você pode marcar vários atributos

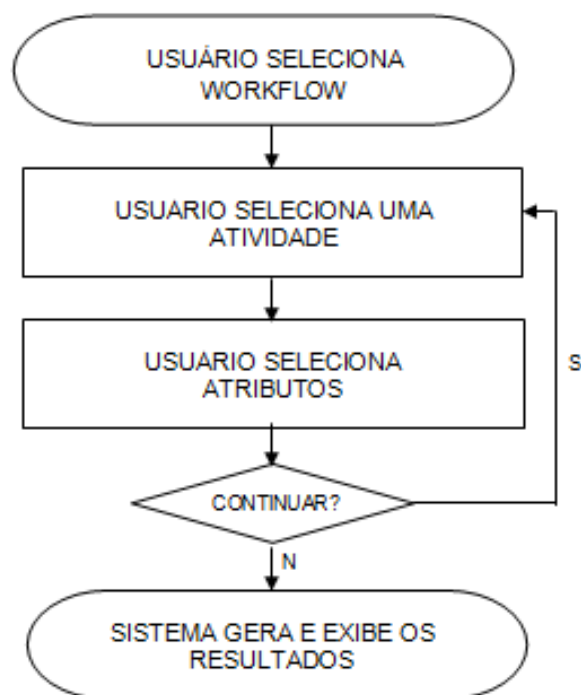
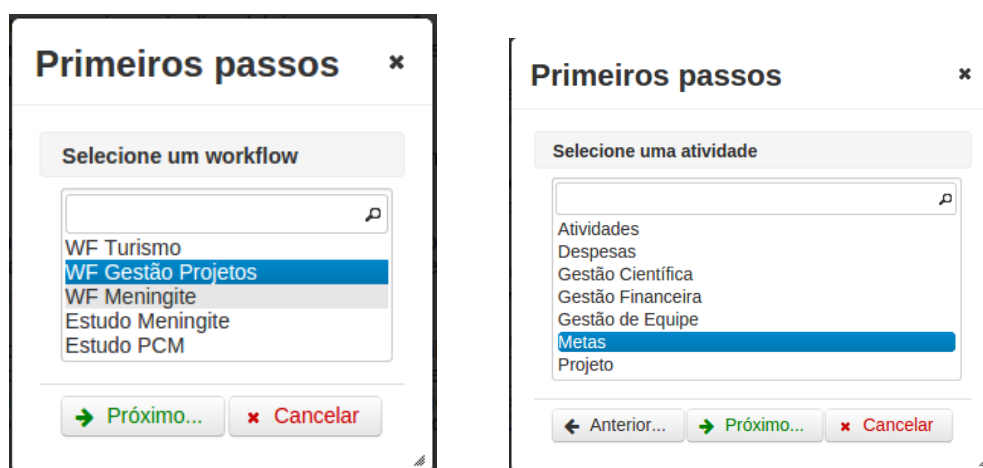


Figura A.6. Fluxo de uso do assistente de consulta

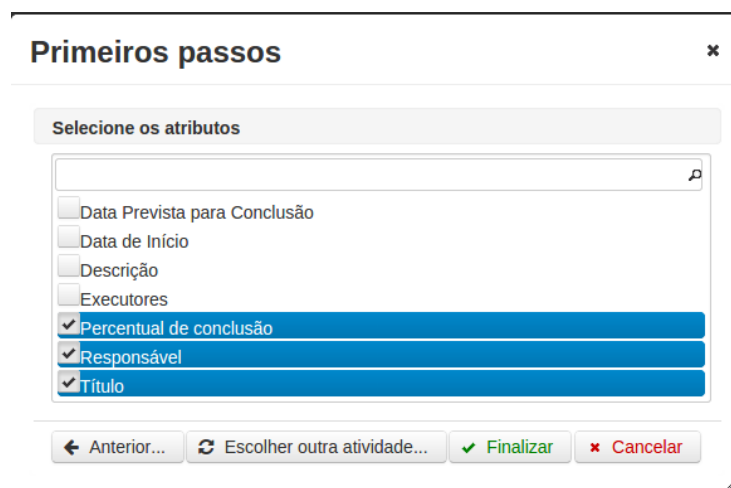
da atividade escolhida. Após a escolha dos campos da atividade selecionada, pode-se continuar adicionando campos, clicando no botão *Escolher outra atividade* ou finalizar o assistente para visualizar os primeiros resultados, através do botão *Finalizar*. Ao finalizar o assistente, o sistema exibe os resultados da consulta na região da tela anteriormente usada para mostrar as dicas iniciais. O assistente é executado uma única vez a cada consulta. Para refinar os critérios da consulta, deve-se usar os recursos disponíveis na tela, para adicionar outros campos, filtros ou critérios de ordenação, além de alterações e exclusões dos itens já incluídos.

A.2.3 Campo de Dados da Consulta

Para entender o refinamento da consulta, é necessário entender o conceito de campo no SiglaSearch. O conceito de *campo* é utilizado para descrever o componente fundamental de um item da consulta. Um *item de consulta* pode ser de três tipos: um campo selecionado, um filtro ou um critério de ordenação. E um *campo de consulta* também possui três classificações: um campo de banco de dados, um campo de sumarização e uma constante. Um *campo de banco de dados* representa todo campo concreto, extraído diretamente do banco de dados, sendo originado de um atributo de atividade ou de uma coluna de registro. Um *campo de sumarização* representa a sumarização de dados, através de uma função de agregação aplicada a um campo de banco

(a) Escolha do *workflow*.

(b) Escolha da atividade.



(c) Escolha dos atributos da atividade selecionada.

Figura A.7. Telas de execução do assistente de consulta do sistema SiglaSearch para os primeiros passos da edição de uma consulta

de dados. Uma *constante* serve para armazenar qualquer valor fixo, lista de valores ou intervalo informado, como um CPF, um intervalo de datas ou uma lista de nomes. A lista completa de tipos de atributos disponibilizados pelo SIGLa é mostrada na Tabela A.1. Um campo de sumarização possui como parâmetro um campo de banco de dados. Os itens de consulta também são parametrizados pelos campos de consulta. É possível selecionar um campo de banco de dados ou campo de sumarização. A ordenação também pode ser aplicada em um desses dois tipos de campos. O filtro possui dois campos da consulta e um operador como parâmetros. Pode-se criar filtro entre dois campos de banco de dados, entre dois campos de sumarização, entre um campo de banco de dados e um campo de sumarização ou entre um campo de banco de dados ou campo de sumarização e uma constante. Ou seja, é possível qualquer combinação de classificação

de campo em um filtro, exceto a combinação de constante com constante.

Classificação	Atributos
Básicos	boolean, date, datetime, float, integer, string, time.
Declarados	barcode, button_many, button_one, CEP, checkbox, CNPJ, CPF, currency, dropdown, email, file, information, menu_many, menu_one, picture, radio, register, select, text, textbox.

Tabela A.1. Tipos de atributos do SIGLa

A fonte de dados de um campo de consulta pode ser um atributo de uma atividade ou uma coluna de dados de um registro. Cada atributo de uma atividade possui um tipo, como *integer*, *CPF*, *register* e *select*, por exemplo. Quando o atributo é do tipo *register* ou *select*, significa que seu valor é representando por uma tupla de registro ou por um conjunto de tuplas de um registro respectivamente. Nesse caso pode-se navegar em um novo nível de detalhe, para escolher uma coluna específica do registro ou deixar como padrão a coluna que nomeia o registro, definida pelo administrador no momento de carga da definição do registro no sistema.

O tipo de dado do campo de consulta é importante para definir compatibilidade com os filtros e com as funções de sumarização. Não podemos, por exemplo, fazer a média de nomes de pacientes, pois se trata de um campo de texto. Mas podemos fazer a média de um campo data ou o somatório de despesas. O sistema também verifica a compatibilidade dos campos da consulta com os filtros. Não podemos comparar data com valor inteiro, por exemplo, ou aplicar filtro de intervalo em campo de nome. O sistema se responsabiliza por verificar os campos disponíveis para cada filtro e para cada função de sumarização e a compatibilidade entre campos na aplicação dos filtros.

A.2.4 Adição de campo selecionado

Para adicionar um campo selecionado à consulta após a execução do assistente, basta clicar no botão *+campo*. O sistema guia você em um conjunto de passos, com apresentação de dicas nos passos menos intuitivos. O fluxograma de passos para adição do campo está ilustrado na Figura A.8. Você deve escolher atividade e atributo. Caso o atributo seja do tipo *register* ou *select*, o sistema apresenta a escolha do campo de registro como próximo passo. Você pode apenas manter a opção padrão, para usar nome do registro, ou escolher outra coluna. O próximo passo é a escolha de função de sumarização, onde você pode escolher uma função de sumarização ou apenas manter a

opção padrão de não aplicar função. Se optar pela aplicação de função, adicionará um campo de sumarização. Se não aplicar função, definirá um campo de banco de dados, oriundo de atributo de atividade ou de campo de registro, que é parte de atributo de tipo *register* ou *select* de uma atividade. O último passo consiste em informar uma descrição para o campo, se não quiser utilizar a opção padrão de nome do atributo.

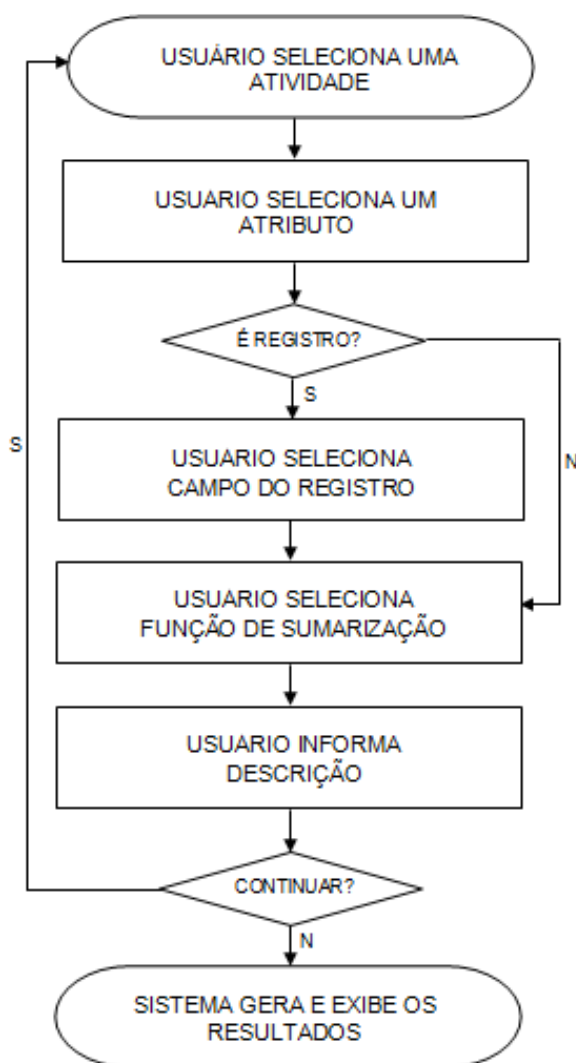


Figura A.8. Fluxo de passos para adição de novo campo à consulta do sistema SiglaSearch

A Figura A.9 mostra as telas para cada passo de adição de campo. O primeiro passo é a escolha da atividade (Figura A.9(a)). O segundo passo é a escolha do atributo (Figura A.9(b)). Caso o atributo seja do tipo *register* ou *select*, o sistema apresenta o passo da Figura A.10. Caso contrário, mostra a tela para escolha da função de sumarização (Figura A.9(c)). Quando a adição de campo envolve registro, o passo de escolha de função é apresentado logo após a tela de escolha de coluna de registro. O

último passo de adição de campo está ilustrado na Figura A.9(d) e é utilizado para informar uma descrição do campo diferente do nome padrão.

(a) Escolha da atividade.

(b) Escolha do atributo.

(c) Escolha de função de sumarização.

(d) Descrição do campo selecionado.

Figura A.9. Passos para seleção de campo de uma consulta do sistema SiglaSearch

O procedimento para adicionar um campo de sumarização é similar ao de adição de um campo de dados. Basta que se escolha uma função de sumarização no passo de adição de função. Como exemplos de campos de sumarização, pode-se citar a média de temperatura corporal por diagnóstico de uma doença ou a data máxima de conclusão das atividades para cada meta e para cada responsável pela execução. A função de sumarização é aplicada em todos os valores encontrados para cada item do conjunto de registros formado pelos campos de bancos de dados (não sumarizados). Se forem selecionados apenas campos de sumarização, as funções de sumarização serão aplicadas em todo o banco de dados, retornando uma única tupla de resultados. Funções de sumarização são aplicadas principalmente em relatórios estatísticos, para análise

Adicionar um campo... ✕

Esse atributo é um registro. Escolher outro campo do registro em vez do nome?

Sim Não

Agencia Financiadora
CPF
Data de Inicio
Data de Nascimento
Data de Termino
Nivel
Nome
Número do Processo da Bolsa
Obs
Tipo da Bolsa
Valor da Bolsa

Um atributo do tipo registro tem outras informações. Este passo permite que você escolha outra informação do registro em vez do nome dele.

← Anterior... → Próximo... ✕ Cancelar

Figura A.10. Escolha de outra coluna do registro do sistema SiglaSearch

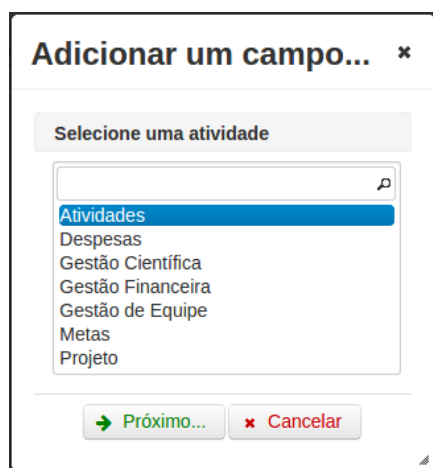
resumida dos dados da aplicação. Os tipos de dados compatíveis para cada função de sumarização estão especificados na Tabela A.2.

Função	Tipos Compatíveis
Contagem	Todos
Somatório	Valores numéricos
Máximo	Valores numéricos e datas
Mínimo	Valores numéricos e datas
Média	Valores numéricos e datas

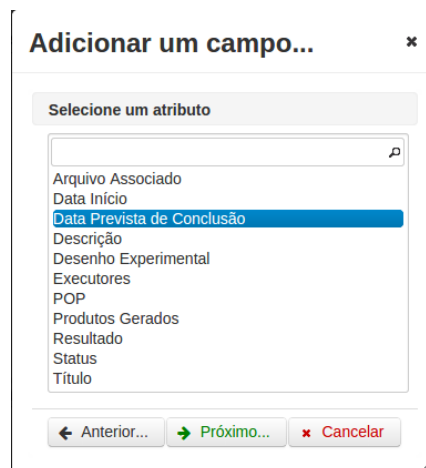
Tabela A.2. Funções de sumarização e tipos compatíveis do sistema SiglaSearch

A Figura A.11 exemplifica os passos para adição de um campo de sumarização. Esse é um exemplo de busca da data máxima de conclusão das atividades em função dos demais campos selecionados no *workflow* de Gestão de Projetos. Se, além desse campo de sumarização, selecionarmos o campo de responsável, o resultado da consulta será a data máxima de conclusão das atividades para cada responsável. Os resultados dessa consulta mostram quais responsáveis estão com prazos mais apertados para a conclusão do projeto, uma função importante para o gerenciamento de projetos. A Figura A.11(a) mostra a escolha da atividade *Atividades*, a Figura A.11(b) mostra a

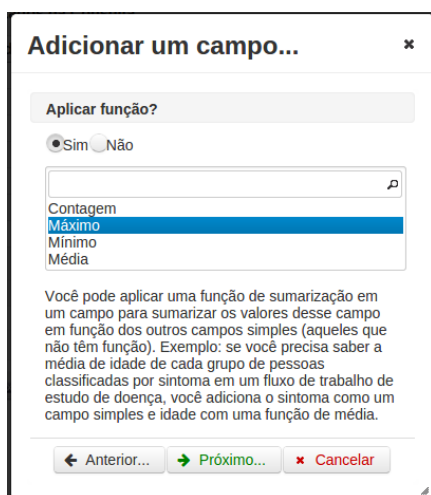
escolha do atributo *Data prevista de conclusão* e a Figura A.11(c) mostra a escolha da função *Máximo* para busca da data máxima de conclusão de atividade, o que representa a data de conclusão de todas as atividades, para cada responsável selecionado. A Figura A.11(d) ilustra a tela de edição do rótulo do campo de sumarização.



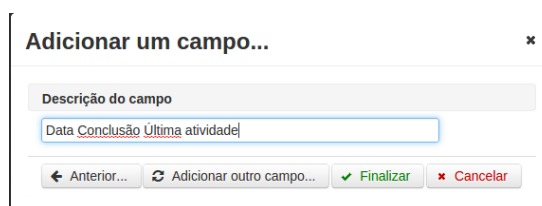
(a) Escolha da atividade.



(b) Escolha do atributo.



(c) Escolha da função de sumarização.



(d) Entrada da descrição do campo de sumarização.

Figura A.11. Passos para adição de campo de sumarização de dados do sistema SiglaSearch

Para alterar a ordem de exibição dos campos selecionados, basta arrastar a coluna da tabela, conforme ilustrado na Figura A.12.

Responsável Meta	Meta
Bianca Fernandes Barros	Criação e geração de Report

Figura A.12. Para modificar a ordem das colunas de resultado do sistema SiglaSearch, basta arrastar as colunas com o *mouse*

A.2.5 Adicionar critério de ordenação

Para adicionar um critério de ordenação à consulta, basta clicar no botão *+ordenação*. O sistema guia você em cada passo. O procedimento para adição de critério de ordenação é bem semelhante ao de escolha de campo. A diferença está no último passo. Não se define descrição de critério de ordenação, mas deve-se definir a opção de ordenação, como crescente ou decrescente, através da tela ilustrada pela Figura A.13.

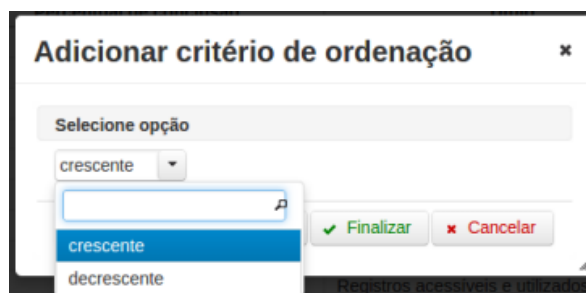


Figura A.13. Escolha da opção de ordenação do campo como crescente ou decrescente no sistema SiglaSearch

A.2.6 Adicionar filtro

Para facilitar o entendimento dos filtros do SiglaSearch, é conveniente fazer uma simplificação dos conceitos de campo de consulta. Conforme seção anterior, um campo de consulta admite as classificações de campo de banco de dados, campo de sumarização e uma constante. Sabemos que tanto o campo de banco de dados quanto o campo de sumarização é extraído do banco de dados do SIGLa. A diferença é que o campo de banco de dados é uma informação concreta e o campo de sumarização é uma informação derivada. Nesta seção sobre filtros de consulta, os campos de banco de dados e os campos de sumarização serão denominados campos de dados do *workflow* ou apenas *campos*. E o campo de constante será denominado aqui como apenas *constante*. As nuances entre escolha de campo de banco de dados e de campo de sumarização são as mesmas tratadas na seção sobre seleção de campo, portanto não serão tratadas nesta seção.

Para adicionar um filtro à consulta, basta clicar no botão *+filtro*. O fluxograma da Figura A.14 esquematiza os passos para a adição de um novo filtro. Após a escolha do primeiro campo, deve-se fazer a escolha do operador do filtro. O sistema exibe apenas os operadores compatíveis com o campo escolhido. Um campo de data, por exemplo, aceita o operador *maior* (*>*), mas não aceita o operador de comparação de textos. Alguns filtros aceitam operação entre dois campos, outros admitem apenas a operação de campo com constante. O filtro de igualdade, por exemplo, aceita a comparação de campo com campo, mas o filtro de intervalo só opera entre campo e constante. A Tabela A.3 mostra os tipos compatíveis com cada tipo de filtro e mostra quais filtros aceitam outro campo, para operação de campo com campo. Caso se escolha um filtro que aceite comparação com outro campo, o sistema solicita a escolha da opção para o segundo operador, para usar outro campo ou uma constante. Se o filtro não aceitar outro campo como opção, o sistema passa diretamente para a escolha da constante. Caso você escolha outro campo como opção, o sistema exibe os passos para a escolha do segundo campo, exibindo apenas os campos compatíveis nas listas de opções. Se escolher filtrar campo com constante ou o filtro admitir apenas constante, o sistema mostra a tela para a entrada da constante com dicas customizadas para cada tipo de filtro e de campo. Em seguida, basta entrar com a descrição do filtro. Você pode encerrar a operação de inclusão de filtro ou solicitar outro filtro no botão *Adicionar outro filtro...* dessa mesma tela.

Filtro	Tipos de dados compatíveis	Aceita dois campos?
Igual	Todos	Sim
Diferente	Todos	Sim
Maior	Valores numéricos e datas	Sim
Menor	Valores numéricos e datas	Sim
Maior ou igual	Valores numéricos e datas	Sim
Menor ou igual	Valores numéricos e datas	Sim
Intervalo	Valores numéricos e datas	Não
Como	Cadeias de caracteres	Não
Não é como	Cadeias de caracteres	Não

Tabela A.3. Especificação dos filtros de consulta do sistema SiglaSearch

As telas específicas para adição de filtro estão ilustradas na Figura A.15. Os passos para seleção de campo não são mostrados aqui, pois são os mesmos aplicados na seleção de campo. A Figura A.15(a) ilustra a escolha do filtro. A Figura A.15(b) mostra a tela para escolha de opção para o segundo operador. A Figura A.15(c) mostra a tela de definição de constante para o segundo operador, que é customizada para orientá-lo a inserir constante compatível com o primeiro operador e com o filtro

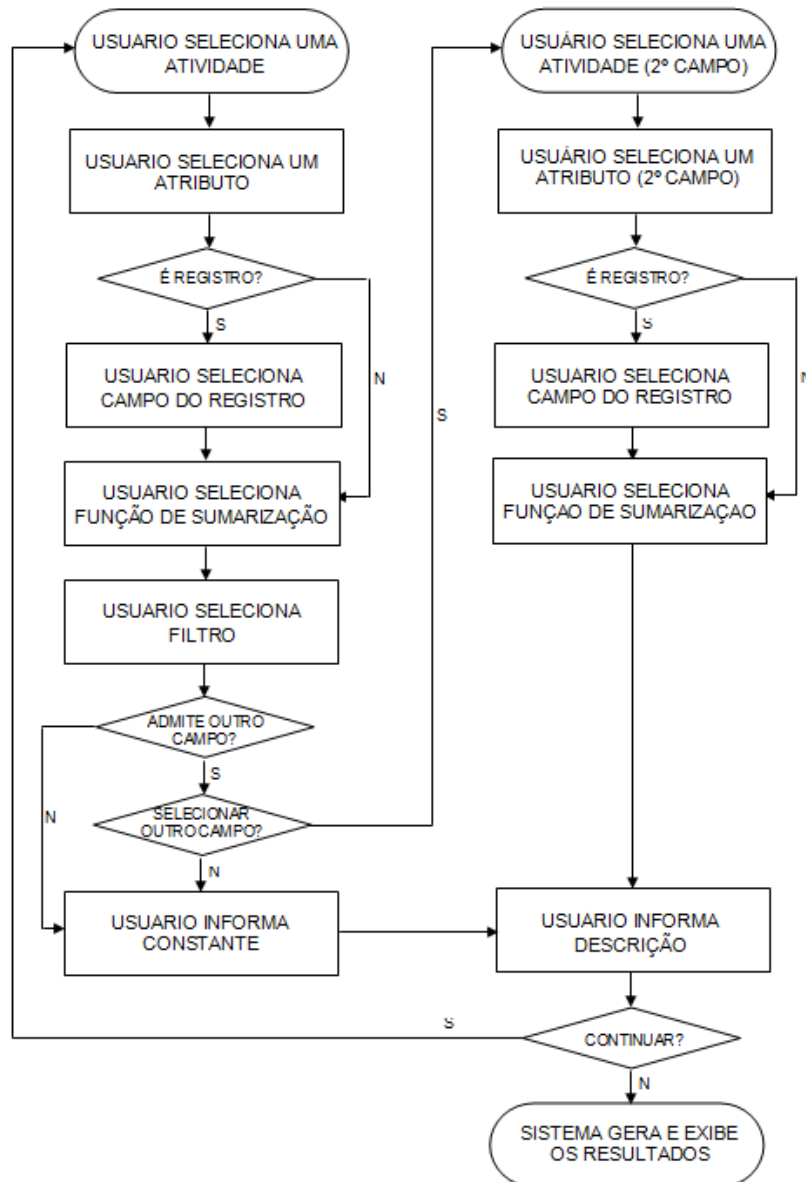
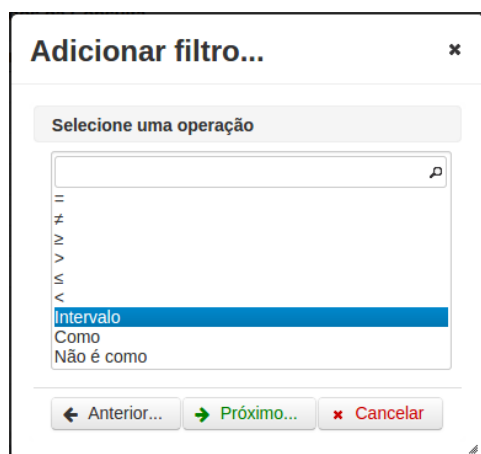


Figura A.14. Fluxo de passos para adição de filtro à consulta do sistema Sigla-Search

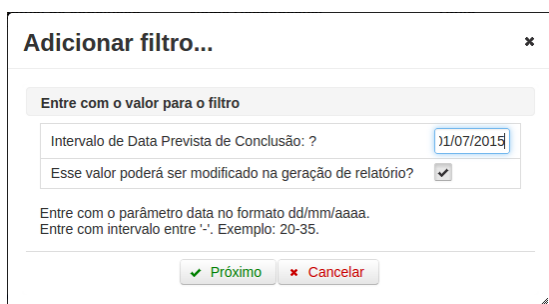
escolhido. Essa tela também exibe a opção para que a constante seja modificada durante a execução do relatório. Essa opção é aplicada quando a consulta editada é salva como relatório para ser disponibilizada aos usuários de áreas especializadas da pesquisa. Por último, deve-se inserir uma descrição para o filtro na tela ilustrada pela Figura A.15(d), para que ele seja entendido de forma clara durante a execução do relatório. O filtro editado na consulta se tornará um parâmetro para o relatório, se a constante puder ser modificada e se a consulta for salva como relatório.



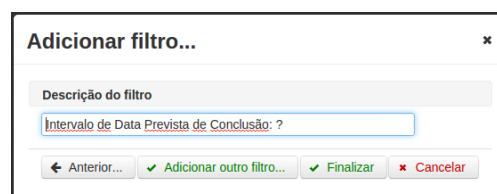
(a) Escolha do filtro.



(b) Escolha da opção para o segundo operador do filtro.



(c) Definição da constante para o segundo operador.



(d) Descrição do filtro.

Figura A.15. Telas dos passos específicos para adição de filtro no sistema Sigla-Search

A.2.7 Alterar e excluir itens da consulta

Todos os itens adicionados à consulta podem ser vistos no lado esquerdo da tela de edição. Para acessar detalhes de cada item, basta clicar no botão de visualização à esquerda do item desejado. Um item de consulta pode ser um campo selecionado, um filtro ou um critério de ordenação. A Figura A.16 mostra a visualização de um campo

selecionado. Há duas telas de visualização de filtros, uma para visualização de filtro entre dois campos (Figura A.17) e outra para filtro entre campo e constante (Figura A.18). A Figura A.19 mostra a visualização de critério de ordenação. Para editar o item, basta clicar no botão *Editar* da mesma tela. Para excluir o item da consulta, basta clicar no botão de exclusão à esquerda do respectivo item.

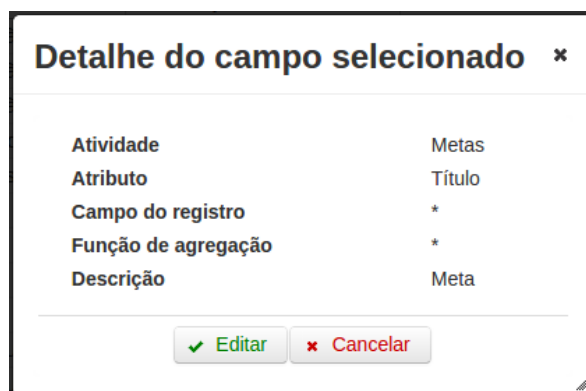


Figura A.16. Tela de visualização dos detalhes de um campo selecionado



Figura A.17. Tela de visualização dos detalhes de um filtro entre dois campos

Detalhe do filtro ✕

	Atividade	Atividades
Primeiro operador	Atributo	Data Prevista de Conclusão
	Campo do registro	*
	Função de agregação	*
Operação	Intervalo (Intervalo de Primeiro operador: Segundo operador)	
Segundo operador	Parâmetro	01/01/2015-01/07/2015
	Esse valor poderá ser modificado na geração de relatório?	<input checked="" type="checkbox"/>
Descrição	Intervalo de Data Prevista de Conclusão: ?	

✓ Editar
✕ Fechar

Figura A.18. Tela de visualização dos detalhes de um filtro entre um campo e uma constante

Detalhe de ordenação ✕

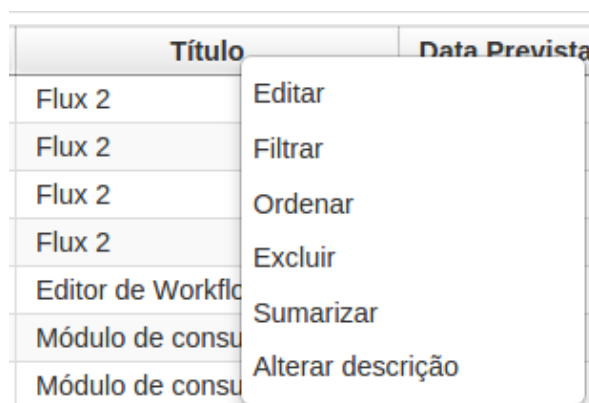
Atividade	Metas
Atributo	Título
Campo do registro	*
Função de agregação	*
Opção	crescente

✓ Editar
✕ Cancelar

Figura A.19. Visualização de critério de ordenação no sistema SiglaSearch

A.2.8 Caminhos alternativos de funcionalidades

Para facilitar o acesso ao sistema, foram criados caminhos alternativos para determinadas funcionalidades. Para cada coluna selecionada e mostrada na tabela de resultados, é exibido um menu de contexto, ilustrado na Figura A.20. O menu *Editar* tem a mesma função do botão de edição da tela de visualização dos detalhes dos itens. O menu *Excluir* tem a mesma função do botão de exclusão exibido no lado esquerdo da tela para o respectivo item. Os menus *Filtrar* e *Ordenar* são atalhos para adição de filtro e critério de ordenação, respectivamente, utilizando como parâmetro a coluna selecionada. O menu *Sumarizar* permite transformar um campo de dados em campo de sumarização, útil para a construção progressiva de relatórios sumarizados. E o menu *Alterar Descrição* é um atalho para edição da coluna, para alterar apenas a descrição do campo, o que é uma das ações mais frequentes de alteração de item.



The image shows a table with two columns: 'Título' and 'Data Prevista'. The 'Título' column contains several rows of text: 'Flux 2', 'Flux 2', 'Flux 2', 'Flux 2', 'Editor de Workflo', 'Módulo de consu', and 'Módulo de consu'. A context menu is open over the 'Título' column, listing the following options: 'Editar', 'Filtrar', 'Ordenar', 'Excluir', 'Sumarizar', and 'Alterar descrição'.

Título	Data Prevista
Flux 2	
Flux 2	
Flux 2	
Flux 2	
Editor de Workflo	
Módulo de consu	
Módulo de consu	

Figura A.20. Menu de contexto para colunas selecionadas no sistema SiglaSearch

A.3 Consulta de Relatórios

Todas as consultas salvas como relatórios são disponibilizadas para os usuários do sistema no menu específico de relatórios (Figura A.21). Ao salvar a consulta como relatório, este é automaticamente exibido como opção no menu de relatórios.

Quando um relatório é acessado, o sistema exibe uma tela inicial com as principais instruções para uso do mesmo (Figura A.22). Cada botão de informação exibe figura ilustrativa para facilitar o entendimento da instrução. Essas instruções iniciais são exibidas até que você execute o relatório pela primeira vez.

Você deve informar os valores para os parâmetros dos relatórios, exibidos no formulário à esquerda da tela. Também podem ser utilizados os valores padrão, definidos



Figura A.21. Menu de relatórios disponíveis do sistema SiglaSearch

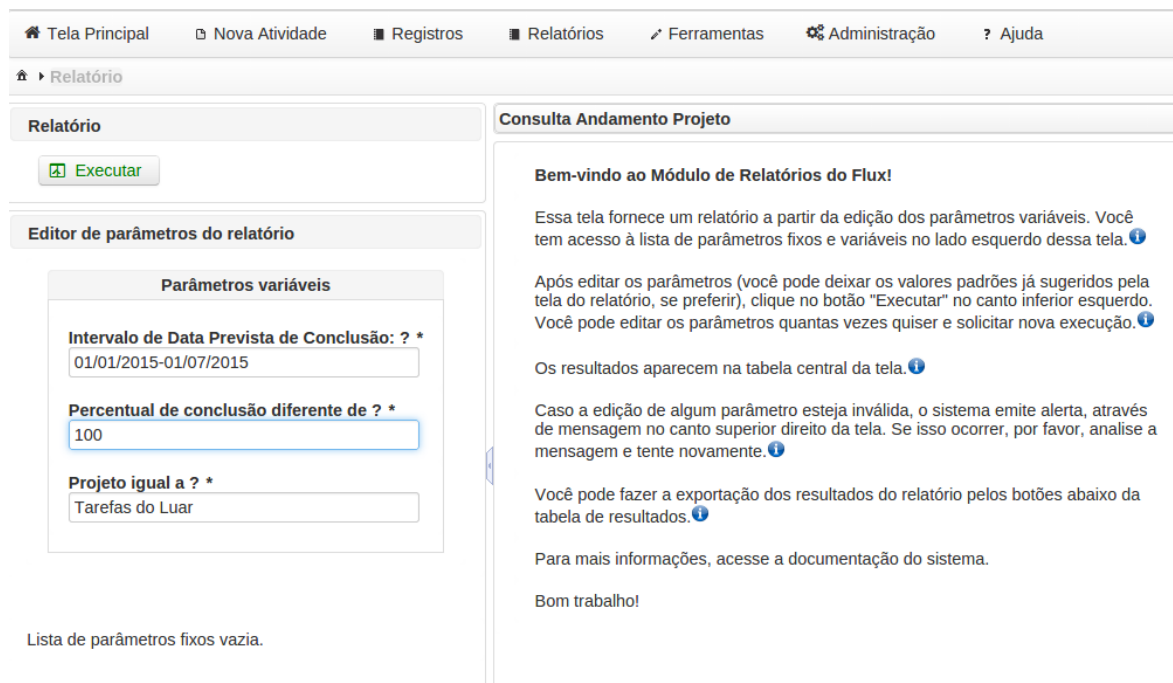


Figura A.22. Tela inicial de Relatório do sistema SiglaSearch

na edição da consulta. Você pode executar o relatório quantas vezes desejar, clicando no botão *Executar* para atualizar os resultados. A Figura A.23 ilustra uma tela de relatório com exibição dos resultados. Os resultados dos relatórios também podem ser exportados em arquivos de formato Excel, CSV, PDF ou XML, através dos botões exibidos abaixo da tabela de resultados.

The screenshot shows a web application interface for a report. At the top, there is a navigation bar with links: Tela Principal, Nova Atividade, Registros, Relatórios, Ferramentas, Administração, and Ajuda. Below this, the report title is 'Consulta Andamento Projeto'. On the left, there is a sidebar with 'Relatório' and 'Executar' buttons, and an 'Editor de parâmetros do relatório' section with 'Parâmetros variáveis' (Intervalo de Data Prevista de Conclusão, Percentual de conclusão diferente, Projeto igual a) and 'Lista de parâmetros fixos vazia.'.

The main content area displays a table titled 'Lista de Resultados da Consulta' with columns: Meta, Responsável Meta, Per, Atividade, and Data Conclusão. The table contains 12 rows of data. Below the table, there is a pagination control showing 'Página 1 de 1' and '12 resultado(s)'. At the bottom, there are buttons for 'Exportar Tudo' and 'Exportar Página Atual', followed by icons for exporting to Excel, CSV, PDF, and XML.

Meta	Responsável Meta	Per	Atividade	Data Conclusão
Editor de Workflows 2.0	André Pinto Ferreira	70	Parser do XPDL2 Editor	27/06/2015
Flux 2	André Pinto Ferreira	40	Pendências de registros/auditoria	23/06/2015
Flux 2	André Pinto Ferreira	40	Atributos multi-select, file	30/06/2015
Flux 2	André Pinto Ferreira	40	I18N Preview de Atributos	23/05/2015
Flux 2	André Pinto Ferreira	40	Parser do XPDL2 Editor	27/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento dos filtros para registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento da ordenação usando registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Limitação dos filtros disponíveis pelo tipo do campo	27/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Busca de dados de registro	26/06/2015
Registros acessíveis e utilizados na busca	Bianca Fernandes Barros	90	Busca de dados de registro	26/06/2015
Registros acessíveis e utilizados na busca	Bianca Fernandes Barros	90	Tratamento dos filtros para registro	25/06/2015
Registros acessíveis e utilizados na busca	Bianca Fernandes Barros	90	Tratamento da ordenação usando registro	25/06/2015

Figura A.23. Tela de relatório com exibição dos resultados do sistema SiglaSearch

A.4 Administração de Consultas

O administrador do sistema pode gerenciar as consultas e relatórios salvos através da tela de administração das consultas (Figura A.24). Através dessa tela, é possível alterar a descrição de consultas e relatórios, disponibilizar relatório ou suspender exibição, através da mudança do indicador de relatório, e excluir consultas e relatórios. O sistema também exibe um atalho para acesso à tela de edição da consulta, através dos botões de edição, para cada consulta exibida.

The screenshot displays the 'Administração de Consultas' (Consultation Administration) interface. At the top, there is a navigation menu with items: 'Tela Principal', 'Nova Atividade', 'Registros', 'Relatórios', 'Ferramentas', 'Administração', and 'Ajuda'. Below the menu, the page title is 'Administração de Consultas'. The main content area is titled 'Lista de consultas' and includes a search filter: 'Filtre por todos os campos: Informe um valor...' and a 'Colunas' button. A table lists the following items:

Descrição	Relatório?	Editar	Excluir
Metas e percentual de conclusão	<input type="checkbox"/>		
Consulta Andamento Projeto	<input checked="" type="checkbox"/>		
Prazos Críticos Projeto	<input checked="" type="checkbox"/>		
Paracoccidiodomicose - Sintomas Frequentes	<input checked="" type="checkbox"/>		
PCM - Contagem Casos Lesão Cutânea	<input checked="" type="checkbox"/>		
Meningite - Casos Fatais	<input checked="" type="checkbox"/>		
Meningite - Contagem Casos por Região	<input checked="" type="checkbox"/>		

Below the table, there is a pagination control showing 'Página 1 de 1' and '7 consulta(s)'. At the bottom, there are two buttons: 'Exportar Tudo' and 'Exportar Página Atual', each with icons for XLS, PDF, CSV, and XML.

Figura A.24. Tela de administração das consultas e relatórios salvos.

Anexo B

Resultados dos Relatórios dos Casos de Uso

B.1 Gestão de Projetos

Para o *workflow* de Gestão de Projetos foram criadas duas consultas. Uma é para verificação do andamento do projeto (Tabela B.1) e a outra para consulta das atividades críticas (últimas atividades a serem concluídas) de cada responsável por meta (Tabela B.2).

B.2 Meningite

Para o *workflow* de Estudo de Meningite foram criadas três consultas. A primeira, para exibir a relação de casos de meningite por escolaridade e por estado da federação (Tabela B.3). A segunda, para exibir a relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos (Tabelas B.4, B.5 e B.6). E a última foi criada para análise da relação de HIV com as sequelas (Tabelas B.7 e B.8).

B.3 Paracoccidiodomicose (PCM)

Foram criadas três consultas para o *workflow* de Paracoccidiodomicose (PCM). A primeira para exibir a relação entre tabagismo prévio com o tempo de evolução da doença e com a quantidade de casos (Tabela B.9). A segunda, para análise de

Tabela B.1. Consulta de andamento do Projeto

Meta	Responsável Meta	Perc Conc	Atividade	Data Conclusão
Criação e geração de Reports concluídas	João Costa Goncalves	100	Criação do menu dinâmico para Report com os relatórios salvos e válidos.	24/06/2015
Criação e geração de Reports concluídas	João Costa Goncalves	100	Criação do formulário dinâmico do Report	24/06/2015
Criação e geração de Reports concluídas	João Costa Goncalves	100	Criação das funções para exportação dos resultados modo Report	18/06/2015
Editor de Workflows 2.0	André Pinto Ferreira	70	Parser do XPDL2 Editor	27/06/2015
Entrada de filtros com maior usabilidade	João Costa Goncalves	100	Limitação dos filtros disponíveis pelo tipo do campo selecionado	27/06/2015
Flux 2	André Pinto Ferreira	40	Atributos multi-select, file	30/06/2015
Flux 2	André Pinto Ferreira	40	I18N Preview de Atributos	23/05/2015
Flux 2	André Pinto Ferreira	40	Parser do XPDL2 Editor	27/06/2015
Flux 2	André Pinto Ferreira	40	Pendências de registros/auditoria	23/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Busca de dados de registro	26/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento dos filtros para registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Tratamento da ordenação usando registro	25/06/2015
Módulo de consulta	Bianca Fernandes Barros	80	Limitação dos filtros disponíveis pelo tipo do campo selecionado	27/06/2015
Registros acessíveis e utilizados na busca, filtro e ordenação	Bianca Fernandes Barros	90	Busca de dados de registro	26/06/2015
Registros acessíveis e utilizados na busca, filtro e ordenação	Bianca Fernandes Barros	90	Tratamento dos filtros para registro	25/06/2015
Registros acessíveis e utilizados na busca, filtro e ordenação	Bianca Fernandes Barros	90	Tratamento da ordenação usando registro	25/06/2015

Tabela B.2. Consulta de prazos críticos do Projeto

Responsável	Data Conclusão Última Atividade
André Pinto Ferreira	30/06/2015
Bianca Fernandes Barros	27/06/2015
João Costa Goncalves	27/06/2015

Tabela B.3. Consulta de casos de meningite por escolaridade e por estado da federação

Escolaridade	Estado	Quantidade Casos
Ensino Fundamental	Espírito Santo	1
Ensino Fundamental	Minas Gerais	3
Ensino Fundamental	Paraná	1
Ensino Fundamental	Rio de Janeiro	8
Ensino Fundamental	Rio Grande do Sul	4
Ensino Fundamental	Santa Catarina	2
Ensino Fundamental	São Paulo	1
Ensino Superior	Espírito Santo	1
Ensino Superior	Minas Gerais	4
Ensino Superior	Paraná	3
Ensino Superior	Rio de Janeiro	1
Ensino Superior	Rio Grande do Sul	1
Ensino Superior	Santa Catarina	1
Ensino Superior	São Paulo	1
Nenhuma	Espírito Santo	1
Nenhuma	Santa Catarina	1

vômitos em pacientes com PCM (Tabela B.10). E a terceira, para exibir a relação de diagnósticos por gênero (Tabela B.11).

Tabela B.4. Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos (Parte I)

Antibiótico	Complicações Supurativas	Complicações Neurológicas	Diagnóstico presuntivo	Diagnóstico confirmatório
amoxicilina	Empiema subdural	Flebites e tromboflebites	encefalite viral	meningite viral
amoxicilina	Empiema subdural		meningite	meningite meningocócica
amoxicilina	Empiema subdural	Outras	encefalite viral	encefalite
amoxicilina	bastonetes gram-positivo		encefalite viral	encefalite viral
amoxicilina	bastonetes gram-positivo	Outras	meningite pneumocócica	meningite pneumocócica
amoxicilina			pneumonia	sepsis
amoxicilina	bastonetes gram-positivo		epilepsia	meningite bacteriana
amoxicilina	Empiema subdural	Outras	meningite bacteriana	meningite meningocócica
amoxicilina	bastonetes gram-positivo	Flebites e tromboflebites	meningite bacteriana	meningite bacteriana
ampicilina			encefalite bacteriana kawasaki	doença de kawasaki
ampicilina	Empiema subdural	Flebites e tromboflebites	meningite bacteriana	meningite meningocócica
ampicilina	Empiema subdural	Flebites e tromboflebites	encefalite	encefalite zoster
ampicilina	bastonetes gram-positivo	Outras	meningite	meningite viral
ampicilina			febre hemorrágica	febre
ampicilina	bastonetes gram-positivo	Flebites e tromboflebites	crise convulsiva por encefalite viral	meningite viral
ampicilina	bastonetes gram-positivo		meningite bacteriana	infecção das vias aéreas superiores
ampicilina	Empiema subdural	Flebites e tromboflebites	meningite meningocócica	gastroenterite
ampicilina	convulsão febril	encefalite		

Tabela B.5. Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos (Parte II)

Antibiótico	Complicações Supurativas	Complicações Neurológicas	Diagnóstico presuntivo	Diagnóstico confirmatório
ampicilina	Empiema subdural	Outras	meningite bacteriana	meningite meningocócica
ampicilina			meningite bacteriana	meningite meningocócica
ampicilina	bastonetes gram-positivo	Outras	meningoencefalite viral	encefalite
ampicilina	bastonetes gram-positivo	Flebites e tromboflebites	meningite	otite aguda
ampicilina	bastonetes gram-positivo		meningite	encefalite
ceftriaxona	Empiema subdural	Outras	meningite bacteriana	meningite pneumocócica
ceftriaxona	Empiema subdural	Outras	febre hemorrágica	meningite meningocócica
ceftriaxona		Outras	meningite	meningite meningocócica
ceftriaxona			meningite	meningite meningocócica
ceftriaxona	Empiema subdural	Outras	encefalite viral	meningite viral
ceftriaxona	Empiema subdural	Flebites e tromboflebites	epilepsia	meningite bacteriana
ceftriaxona	bastonetes gram-positivo	Flebites e tromboflebites	meningite	encefalite
ceftriaxona	Empiema subdural	Flebites e tromboflebites	meningite	meningoencefalite
ceftriaxona	bastonetes gram-positivo	Outras	meningite	meningite viral
ceftriaxona			meningite bacteriana	meningite meningocócica
ceftriaxona	convulsão febril		encefalite viral	meningite meningocócica
ceftriaxona	bastonetes gram-positivo	Flebites e tromboflebites	meningite	meningite sem etiologia definida
ceftriaxona	bastonetes gram-positivo	Outras	meningite bacteriana	meningite bacteriana

Tabela B.6. Relação de antibióticos utilizados com as complicações supurativas e neurológicas e com os diagnósticos (Parte III)

Antibiótico	Complicações Supurativas	Complicações Neurológicas	Diagóstico presuntivo	Diagóstico confirmatório
ceftriaxona	Empiema subdural	Outras	meningite	meningite meningocócica
clavulin	bastonetes gram-positivo	Flebites e tromboflebitis	encefalite viral	meningite viral
clavulin	meningite	Outras	meningite	meningite viral
clavulin	bastonetes gram-positivo	Flebites e tromboflebitis	encefalite viral	encefalite
clavulin	Empiema subdural	Flebites e tromboflebitis	meningoencefalite viral	encefalite
clavulin	Empiema subdural	Flebites e tromboflebitis	infecção das vias aéreas superiores	infecção das vias aéreas superiores
clavulin	Empiema subdural	Flebites e tromboflebitis	encefalite viral	meningite meningocócica
gentamicina			meningite	meningite viral
gentamicina			meningite	meningite viral
gentamicina	bastonetes gram-positivo	Outras	meningite meningocócica	meningite meningocócica
gentamicina	bastonetes gram-positivo	Flebites e tromboflebitis	meningite	meningite viral
gentamicina	Empiema subdural	Outras	meningite meningocócica	meningoencefalite
gentamicina	convulsão febril	Outras	meningite	meningite bacteriana
gentamicina	convulsão febril	Flebites e tromboflebitis	meningite	meningite viral
gentamicina	bastonetes gram-positivo		meningite bacteriana	infecção das vias aéreas superiores
penicilina	Empiema subdural	Flebites e tromboflebitis	leptospirose com acometimento	meningite viral

Tabela B.7. Análise da relação de HIV com as sequelas (Parte I)

HIV	Paralisia Cerebral	Retardo Mental	Surdez	Hidrocefaleia	Cegueira	Admissão
Sim	Sim					31/08/2013
Sim						17/08/2013
Sim			Sim			12/08/2013
Sim	Sim		Sim	Sim		27/05/2013
Sim			Sim			04/01/2014
Sim	Sim			Sim		04/03/2014
Sim	Sim					18/04/2015
Sim						19/08/2013
Sim			Sim			13/12/2014
Não						18/08/2015
Não						05/04/2013
Não	Sim	Sim	Sim			04/08/2014
Não	Sim	Sim				25/08/2014
Não						11/08/2014
Não		Sim	Sim			04/08/2014
Não	Sim	Sim				06/08/2013
Não	Sim	Sim				22/08/2014
Não						11/08/2014
Não						04/08/2014
Não			Sim			05/02/2014
Não						11/08/2014
Não				Sim		05/03/2014
Não						19/08/2014
Não				Sim		20/08/2013
Não						05/11/2013
Não						04/08/2014
Não	Sim	Sim				11/04/2015

Tabela B.8. Análise da relação de HIV com as sequelas (Parte II)

HIV	Paralisia Cerebral	Retardo Mental	Surdez	Hidrocefalia	Cegueira	Admissão
Não						26/04/2014
Não						10/06/2013
Não						18/10/2014
Não						18/08/2013
Não						04/08/2014
Não						11/08/2014
Não						12/08/2014
Não						12/08/2014
Não						26/05/2014
Não						04/08/2013
Não					Sim	10/08/2013
Não						27/08/2013
Não	Sim	Sim				20/04/2015
Não						11/03/2015
Não						10/08/2013
Não				Sim		24/07/2013
Não						05/08/2013
Não						04/08/2013
Não						24/08/2014
Não		Sim				11/04/2013
Não					Sim	19/08/2013
Não						19/08/2013
Não						21/03/2015
Não			Sim	Sim		19/04/2013
Não					Sim	04/10/2014
Não						12/03/2013

Tabela B.9. Relatório da relação entre tabagismo prévio com o tempo de evolução da doença e com a quantidade de casos

Tabagismo Prévio	Lesão Cutânea	Média Tempo Evolução	Quantidade Casos
	Não	9.86	14
	Sim	14.18	17
Não		0.5	2
Não	Não	8.67	24
Não	Sim	5.79	19
Sim		15.0	2
Sim	Não	20.85	92
Sim	Sim	18.21	48

Tabela B.10. Análise de vômitos em pacientes com PCM

Vômitos	Média Idade Primeira Consulta	Quantidade Casos
Não	37.78	170
Sim	31.73	33

Tabela B.11. Relação do diagnóstico por gênero

Gênero	Quantidade Casos	Média Idade Primeira Consulta	Tempo Médio Evolução
Feminino	41	26.83	10.02
Masculino	181	39.04	17.4