

# EasyRA

## Uma ferramenta para tradução de consultas em álgebra relacional para SQL

Luís Felipe Bilecki<sup>1</sup>, Vivian Cremer Kalempa<sup>1</sup>

<sup>1</sup>Departamento de Sistemas de Informação  
Universidade do Estado de Santa Catarina (UDESC) - São Bento do Sul - SC - Brasil

felipe436@gmail.com, vivian.kalempa@udesc.br

**Abstract.** *This paper presents the project and development of a software, EasyRA, to translate expressions of relational algebra in SQL queries. One of the difficulties is that the understanding of the syntax of the relational algebra operators, becomes difficult without the use of a tool to validate the expressions. In the EasyRA, these expressions are sent to the compiler module, which produces the SQL query. After conversion, the query execution occurs in the DBMS and the student can view the results. As a way of validation, EasyRA was applied in the database subject for students validate your queries and evaluate the tool about its usability and functionality.*

**Resumo.** *Este artigo apresenta o projeto e o desenvolvimento de uma ferramenta, EasyRA, para tradução de expressões da álgebra relacional em consultas SQL. Uma das dificuldades é que a compreensão da sintaxe dos operadores da álgebra relacional, torna-se difícil sem o uso de uma ferramenta para validar as expressões da álgebra relacional. Já na EasyRA, essas expressões são enviadas para o módulo compilador, que produz a consulta SQL. Após a conversão, a consulta é executada no SGBD e o aluno pode visualizar os resultados obtidos. Como forma de validação, a ferramenta EasyRA foi aplicada na disciplina de Banco de Dados para os alunos validarem suas consultas e avaliarem a ferramenta quanto a sua usabilidade e funcionalidade.*

### 1. Introdução

A manipulação de registros em banco de dados é possível através do uso de um Sistema de Gerenciamento de Banco de Dados Relacional (SGBDR). Um SGBDR é o *software* responsável por todo o acesso aos bancos de dados relacionais, sendo todas as requisições de acesso tratados pelo SGBDR [Date 2003]. O SGBDR disponibiliza uma interface para fornecer uma maneira de recuperar e realizar a manutenção das informações de forma conveniente e eficiente. Nos modelos relacionais, a linguagem SQL se tornou a linguagem padrão, tanto em *Data Definition Language* (DDL) quanto em *Data Manipulation Language* (DML) [Silberschatz et al. 2006].

A linguagem SQL tem como base formal a álgebra relacional e o cálculo relacional. A álgebra relacional é o conjunto básico de operações para o modelo relacional, permitindo ao usuário formar novas relações com base em uma ou mais relações. Normalmente é considerada parte integral do modelo relacional e suas operações se dividem em dois grupos: unárias e binárias [Elmasri and Navathe 2011].

Uma série de fatores justifica o estudo da álgebra relacional e demonstra que sua compreensão é de extrema importância. Primeiro, ela oferece uma base formal para as operações do modelo relacional. Segundo, ela é utilizada como base para implementação e otimização de consultas nos módulos de otimização e processamento de consultas que são partes integrais dos SGBDR. Terceiro, alguns de seus conceitos são incorporados na base da linguagem de consulta padrão SQL para SGBDR [Elmasri and Navathe 2011].

Diversas ferramentas foram desenvolvidas para serem utilizadas em ambiente acadêmico, como SIMALG [Lautert 2010], EnsinAR [Paes 2004], ProgramAR [Prates et al. 2013], iDFQL [Appel and Traina Jr. 2004]. Porém, as ferramentas existentes possuem limitações como: dificuldades no processo de instalação, sintaxe e simbologia não adequada, falta de documentação, interface pouco intuitiva, entre outras. Considerando este problema, o presente artigo apresenta o projeto e o desenvolvimento de uma ferramenta, denominada de EasyRA, para traduzir as expressões da álgebra relacional em consultas SQL explorando as carências dos *softwares* analisados e implementando novas funcionalidades, que irão auxiliar durante a resolução dos exercícios da álgebra relacional na disciplina de banco de dados e na compreensão da sintaxe e do funcionamento dos operadores.

Esse artigo está organizado da seguinte maneira: a seção 2 apresenta os conceitos de álgebra relacional, a seção 3 descreve os trabalhos relacionados, a seção 4 apresenta o funcionamento e as funcionalidades da ferramenta EasyRA, a seção 5 apresenta a validação da ferramenta na disciplina de Banco de Dados e, por fim, a seção 6 apresenta a conclusão e os trabalhos futuros.

## 2. Álgebra Relacional

A álgebra relacional consiste em um conjunto de operações que usam uma ou duas relações de entrada e como resultado produzem uma nova relação [Silberschatz et al. 2006]. Embora a álgebra relacional defina um conjunto de operações específico para o modelo relacional, o cálculo relacional oferece uma linguagem de nível mais alto para especificar consultas relacionais [Elmasri and Navathe 2011]. Os dois conceitos são de extrema importância, pois formam a base da linguagem padrão comercial (SQL) para consultas em bancos de dados relacionais.

As operações são utilizadas para a seleção de linhas de uma determinada tabela ou a combinação de linhas relacionadas a diversas tabelas com a finalidade de especificar uma consulta ou uma requisição de recuperação sobre o banco de dados [Takai et al. 2005].

As operações da álgebra relacional são divididas em dois grupos: os operadores unários e binários. Os operadores unários, são destacados pelas operações: seleção, projeção e renomeação. Enquanto, os operadores binários, destacam-se as operações baseadas na teoria dos conjuntos e as junções [Elmasri and Navathe 2011].

### 2.1. Operadores unários

Os operadores unários atuam sobre uma única relação, produzindo um subconjunto dos elementos especificados na condição. Neste grupo destacam-se, as operações: seleção, projeção e renomeação.

A operação de seleção, representada pelo símbolo  $\sigma$  (sigma), tem por finalidade selecionar as tuplas que satisfazem uma determinada condição predefinida. Divide a relação em dois conjuntos, o que satisfaz a condição é apresentado e o que não está dentro das condições estabelecidas é descartado [Silberschatz et al. 2006]. A operação trabalha de forma distinta da cláusula SELECT da SQL, pois, é análoga a cláusula WHERE [Elmasri and Navathe 2011].

A operação de projeção, representada pelo símbolo  $\pi$  (pi), retorna sua relação de argumento, com certos atributos omitidos. Como a relação é um conjunto, quaisquer linhas duplicadas são eliminadas [Silberschatz et al. 2006].

A operação de renomeação, identificada pelo símbolo  $\rho$  (rho), é utilizada para renomear os atributos de uma relação ou algum atributo da lista. Se nenhuma operação de renomeação for aplicada na relação resultante, os nomes dos atributos serão iguais ao da relação original e estarão na mesma ordem original [Elmasri and Navathe 2011].

## 2.2. Operadores com base na teoria dos conjuntos

Os operadores baseados na teoria dos conjuntos são operações binárias, aplicadas a dois conjuntos de tuplas. Quando aplicada nos bancos de dados relacionais, é necessário existir a compatibilidade de união ou de tuplas, ou seja, as relações precisam ter o mesmo número de atributos (grau) e cada par de atributos deve estar inserido no mesmo domínio [Elmasri and Navathe 2011]. Conforme [Silberschatz et al. 2006], as operações são definidas a seguir:

- União: indicada pelo símbolo  $\cup$ . A expressão  $R \cup S$  fornece como resultado, um conjunto das tuplas que estão presentes em  $R$  ou em  $S$  ou tanto em  $R$  quanto em  $S$ ;
- Interseção: indicada pelo símbolo  $\cap$ . A operação de interseção fornece as tuplas que as relações possuem em comum;
- Diferença: indicada pelo símbolo  $-$ . A expressão  $R - S$  tem como resultado, as tuplas que estão presentes na relação  $R$ , mas não estão em  $S$ .

## 2.3. Operadores binários de junção

As operações binárias que pode-se citar são: junção, junção natural, junções externas, divisão, produto cartesiano e atribuição.

A operação de junção, indicada pelo símbolo  $\theta \times$ , tem como finalidade, combinar as tuplas de duas relações dentro de uma tupla única. Pode ser definida como a combinação de dois operadores: a seleção e o produto cartesiano [Elmasri and Navathe 2005].

A operação de junção natural, representada pelo símbolo  $\bowtie$ , forma um produto cartesiano de duas relações e realiza uma seleção forçando a igualdade dos atributos que aparecem nas relações e, por fim, remove os atributos que estejam duplicados [Silberschatz et al. 2006].

A operação de junção externa é considerada uma extensão da operação de junção. A junção externa esquerda, representada por  $\ltimes$ , adiciona todas as tuplas da relação esquerda, mesmo quando não existir tuplas correspondentes na relação direita. A junção externa direita, representada por  $\rtimes$ , é inversa à junção externa esquerda, pois adiciona

todas as tuplas da relação direita, mesmo quando não há tuplas correspondentes na relação esquerda. A junção externa completa, representada por  $\bowtie$ , apresenta uma combinação dos dados das relações, trazendo como resultado todas as tuplas de ambas as relações e apresenta valores *NULL* para as tuplas sem correspondência [Silberschatz et al. 2006].

A operação de divisão, identificada pelo símbolo  $\div$ , é aplicada a duas relações  $R(Z) \div S(X)$ , onde os atributos de  $R$  estão contidos em  $S$ . Considerando que  $Y$  seja o conjunto dos atributos de  $R$ , que não são atributos de  $S$ , então,  $Y = Z - X$ . O resultado da divisão é uma relação  $T(Y)$ , para cada tupla  $t$  desta relação, os valores deverão aparecer em  $R$  combinando com cada tupla em  $S$  [Elmasri and Navathe 2011].

A operação de produto cartesiano, representada pelo símbolo  $\times$ , tem como principal função a combinação das tuplas de duas relações [Elmasri and Navathe 2011].

A operação de atribuição, determinada pelo símbolo  $\leftarrow$ , tem funcionalidade semelhante a atribuição, em linguagem de programação. É utilizada para atribuir o resultado de uma expressão da álgebra relacional a uma variável temporária [Silberschatz et al. 2006].

### 3. Trabalhos Correlatos

As quatro ferramentas analisadas foram desenvolvidas para serem utilizadas no meio acadêmico e apresentam a finalidade de auxiliar os alunos durante o processo de ensino-aprendizagem dos conceitos e operações da álgebra relacional.

Proposta por [Lautert 2010], a ferramenta SIMALG um simulador de consultas em álgebra relacional, foi desenvolvida em Java e foram implementadas sete operações (Seleção, Projeção, União, Diferença, Produto Cartesiano, Junção Natural e Interseção). A ferramenta não possibilita a troca de banco de dados e não é flexível ao desenvolver expressões mais complexas, ou seja, para realizar uma consulta mais extensa, é necessário subdividir a consulta e executar cada uma separadamente. Tendo em vista as desvantagens citadas, a ferramenta apresenta um bom *feedback* ao executar as consultas e um menu de ajuda para auxiliar os usuários durante o uso.

A ferramenta EnsinAR, proposta por [Paes 2004], foi desenvolvida para auxiliar no ensino de álgebra relacional e banco de dados, baseando-se nas conversões de expressões da álgebra relacional para SQL. As tabelas utilizadas na ferramenta, devem ser construídas na própria aplicação, impossibilitando a troca de banco de dados ou o uso de um banco já existente. Além disso, a ferramenta não possui alguns detalhes que possibilitam a portabilidade, como por exemplo, a capacidade de trabalhar com mais de um SGBD (trabalha somente com SQL *Server*) e a conversão de SQL para álgebra relacional (compilação reversa), descrita como trabalhos a serem realizados.

Proposta por [Prates et al. 2013], a ferramenta ProgramAR, foi desenvolvida na UFBA para ser aplicada nos cursos de Bacharelado em Sistemas de Informação e Ciência da Computação. Através desta ferramenta, o discente pode montar as expressões em álgebra relacional e acompanhar o resultado dessas consultas. Esta ferramenta contempla as principais operações da álgebra relacional, somente as operações da teoria dos conjuntos (união, interseção e diferença) foram deixadas para ser implementadas em uma próxima versão. Apesar da incapacidade de trabalhar com mais de um SGBD (somente foi implementado MySQL), a ferramenta ProgramAR auxiliou os discentes da

disciplina de banco de dados durante a aplicação dos conceitos de álgebra relacional.

A ferramenta iDFQL, proposta por [Appel and Traina Jr. 2004], utiliza uma abordagem construcionista para que o aluno esteja exposto ao processo de aprendizado prático, por meio da construção de uma consulta em álgebra relacional através de elementos gráficos. Diferentemente das outras ferramentas apresentadas, a iDFQL utiliza árvores de expressões algébricas relacionais para representar operações da álgebra relacional.

Apesar das ferramentas possuírem características em comuns, algumas não implementam certos requisitos essenciais como, por exemplo, uma interface simples e a mesma notação utilizada na literatura, para o bom funcionamento e usabilidade de uma ferramenta aplicada no ambiente educacional. Visando ser uma ferramenta diferencial, a EasyRA apresenta as seguintes características: portabilidade de SGBD; tradução de álgebra relacional para SQL; conversão reversa (SQL para álgebra relacional); multiplataforma; manual de uso e instalação; funcionalidade de salvar a consulta que for construída; visualização da consulta SQL; flexibilidade; boa usabilidade; expressões da álgebra relacional e operações fundamentais da álgebra relacional.

#### 4. Ferramenta EasyRA

A ferramenta EasyRA, desenvolvida na linguagem Java, é composta de um módulo compilador, que tem como entrada as expressões em álgebra relacional ou consultas SQL dependendo do modo selecionado, realiza a conversão da expressão para SQL e, como saída, exibe o resultado em uma tabela na interface e informa a consulta SQL gerada para o usuário.

De acordo com a Figura 1, o fluxo de funcionamento segue as etapas:

1. A expressão da álgebra relacional é a entrada da ferramenta;
2. A expressão é encaminhada ao módulo compilador que retorna a consulta convertida em SQL;
3. A consulta SQL é executada no SGBD;
4. O resultado obtido e a consulta convertida são visualizados pelo usuário.

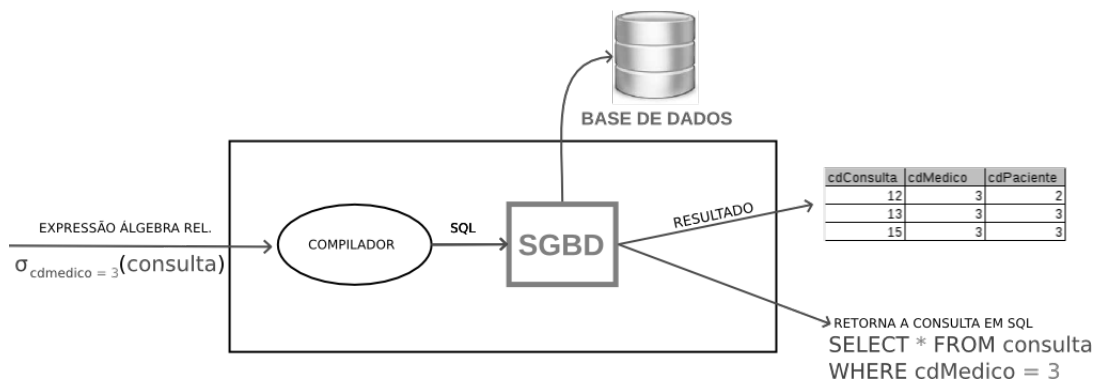


Figura 1. Diagrama de funcionamento da ferramenta EasyRA

Considerando que os trabalhos correlatos contribuíram para a definição das principais características da ferramenta EasyRA, visando se tornar uma ferramenta diferencial, a EasyRA apresenta as seguintes funcionalidades, além das características já apresentadas:

- Portabilidade de SGBD: permite conexão com mais de um SGBD (MySQL, Firebird e PostgreSQL);
- Tradução reversa: traduzir uma consulta SQL em uma expressão da álgebra relacional;
- Multiplataforma: permite o uso em Windows ou Linux;
- Documentação: manuais de instalação e de utilização da ferramenta são disponibilizados aos docentes;
- Manipulação de arquivos: permite salvar o trabalho realizado para efetuar alterações futuras no arquivo;
- Flexibilidade: possibilita a construção de expressões complexas da álgebra relacional;
- Expressões e operações fundamentais da álgebra relacional: implementa as operações ensinadas na disciplina de banco de dados.

Nas seções 4.1 e 4.2 são descritos os módulos da ferramenta EasyRA e na seção 4.3 é definido a interface principal da ferramenta.

#### 4.1. Módulo Compilador de Álgebra Relacional para SQL

A ferramenta EasyRA dispõe de um módulo compilador que recebe como entrada uma expressão da álgebra relacional. A expressão é enviada para os analisadores léxico, sintático e semântico até alcançar a etapa de geração do código (conversão para SQL).

Para o desenvolvimento deste módulo, foram analisadas três ferramentas, buscando selecionar uma ferramenta que possua uma vasta documentação e exemplos práticos. Foram selecionadas as seguintes ferramentas:

- JFlex: um poderoso analisador léxico que gera analisadores escritos na linguagem Java. A sintaxe, inspirada na sintaxe de especificações Lex, facilita o desenvolvimento de gramáticas. Também apresenta um manual de instalação e uso da ferramenta, porém o aprendizado se torna difícil para quem não conhece a linguagem Java e os conceitos básicos de compiladores [Klein 2014];
- GALS: desenvolvido como TCC na UFSC em 2003, a ferramenta GALS gera automaticamente os analisadores léxico, sintático e semântico, através do fornecimento de uma gramática. Todas as documentações assim como a ferramenta estão disponíveis no *website* <http://gals.sourceforge.net/> [Gesser 2003];
- JavaCC: JavaCC acrônimo para Java *Compiler Compiler*, é um gerador de *parser* e analisador léxico para a linguagem Java, que lê a especificação da linguagem e converte para Java, podendo reconhecer programas para a devida linguagem na qual a gramática foi construída [Norvell 2011].

Após a análise das ferramentas apresentadas, optou-se por utilizar o GALS como ferramenta para geração das classes dos analisadores, tendo em vista que o mesmo possui uma extensa documentação e três opções de linguagens (Java, C++ e Delphi) para a geração das classes dos analisadores.

Depois da escolha da ferramenta GALS, a gramática utilizada na ferramenta EasyRA, foi desenvolvida buscando alterar constantemente a sua definição, para que se adaptasse a execução e sintaxe das operações da álgebra relacional.

A ferramenta GALS, gerou onze classes (AnalysisError, Constants, LexicalError, Lexico, ParserConstants, ScannerConstants, SemanticError, Semantico, Sintatico, SyntaticError e Token), somente a classe Semantico foi adaptada para ser utilizada na ferramenta EasyRA. A classe Semantico redireciona todas as ações semânticas geradas pelo compilador para a classe Analise, essas ações são numeradas e precedidas de um símbolo como, por exemplo, (#1).

A entrada do compilador é uma expressão da álgebra relacional traduzida. Caso ocorra algum erro sintático ou léxico (caractere não esperado), eles são tratados pelos métodos (analisarErroSintatico e analisarErroLexico) que analisam outras condições para trazer uma resposta mais precisa ao usuário. Considerando que a etapa anterior foi realizada com sucesso, o compilador trata as ações semânticas geradas, conforme ilustrado na Figura 2, e a execução é repassada para o método montar da classe Analise, que analisa as ações geradas, realizando as verificações necessárias para traduzir a expressão em consulta SQL.

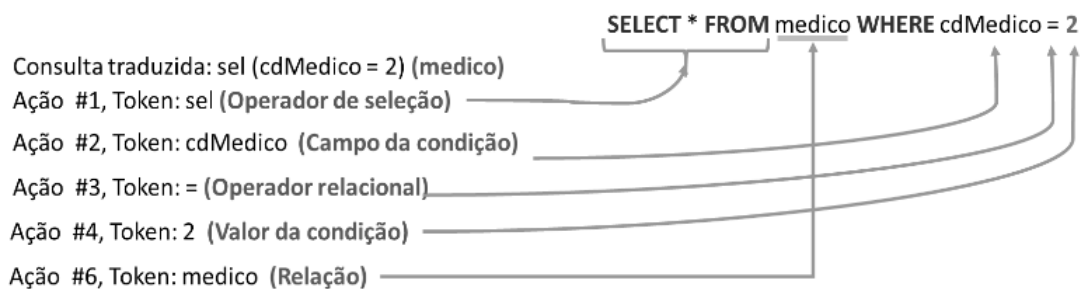


Figura 2. Ações semânticas e a tradução no compilador

Com a finalização da construção da gramática, as classes geradas foram anexadas à ferramenta EasyRA.

#### 4.2. Módulo Compilador de SQL para Álgebra Relacional

Além do módulo compilador apresentado na seção 4.1, a ferramenta EasyRA também disponibiliza a opção de traduzir uma consulta SQL para uma expressão da álgebra relacional. Esta tradução é realizada através da biblioteca JSQParser.

A biblioteca JSQParser, analisa uma instrução SQL e a converte para uma hierarquia de classes em Java. Por meio desta separação em hierarquia de classes, se torna possível a visualização de cada elemento da consulta SQL, bem como o tipo da consulta se é um SELECT, SELECT com SUBSELECT, INSERT, UPDATE ou DELETE. Por exemplo, a biblioteca JSQParser processa a instrução SQL abaixo, conforme a Figura 3.

**SELECT \* FROM medico WHERE cdMedico = 2**

Após a separação, o método gerarExpressaoAlgebra da classe AnaliseReversa, constrói a expressão da álgebra relacional, conforme os elementos preenchidos. Para o exemplo, ilustrado na Figura 3, a expressão resultante é:

$$\sigma_{(cdMedico=2)}(medico)$$

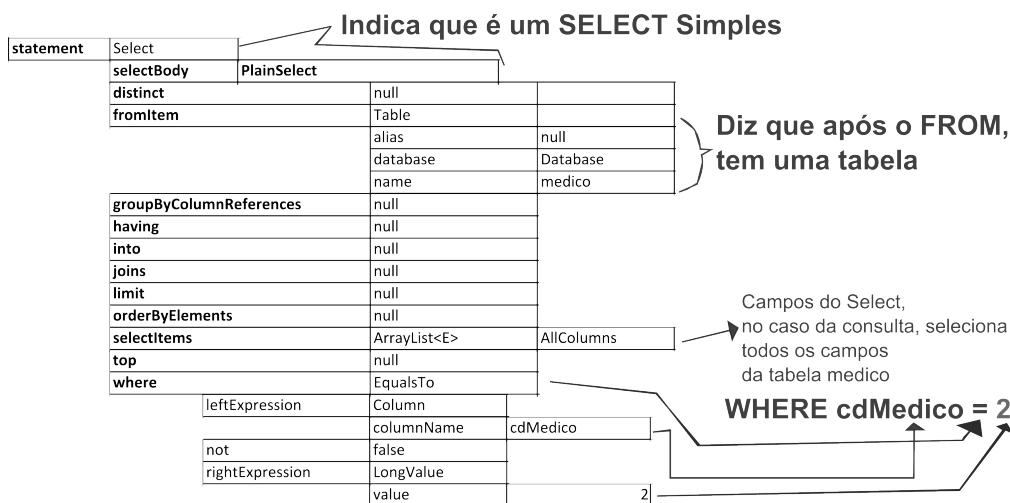


Figura 3. Consulta SQL no JSQParser

Na subseção 4.3 é apresentada a interface principal da ferramenta EasyRA.

### 4.3. Interface Gráfica

A interface gráfica da ferramenta EasyRA ilustrada na Figura 4, conta com um menu de opções e uma barra de ferramentas com ícones de acesso rápido. A janela principal da ferramenta é dividida em quatro áreas.

A área I representa o painel para a construção de expressões da álgebra relacional, contendo as operações da álgebra relacional (seleção, projeção, produto cartesiano, união, interseção, diferença, divisão, junção, junção natural, junções externas (completa, esquerda e direita), renomeação e atribuição).

A área II exibe a consulta SQL retornada pelo compilador e neste campo o usuário pode também digitar a consulta SQL e optar por executá-la ou traduzi-la para álgebra relacional.

A área III apresenta as abas Resultado, Console e Histórico. O resultado exibe as linhas e colunas que a consulta SQL retornou. A aba Console é utilizada para a interação entre a aplicação e o usuário. E, por último, a aba Histórico exibe as consultas executadas, podendo o usuário removê-las ou recuperá-las.

A área IV apresenta uma lista com as tabelas e os campos do banco de dados conectado.

A seção 5 descreve a aplicação prática da ferramenta EasyRA na disciplina de Banco de Dados do Curso de Bacharelado em Sistemas de Informação da UDESC/CEPLAN.





Figura 4. Interface principal da ferramenta EasyRA

## 5. Validação da ferramenta EasyRA

Para validar a ferramenta EasyRA, foi proposto a aplicação prática na disciplina de Banco de Dados, durante o período de ensino da álgebra relacional, que ocorreu nos dias 20 e 21 de outubro de 2014 e também nos dias 03, 10 e 11 de novembro de 2014.

Foi disponibilizado aos alunos um arquivo compactado com os arquivos para a instalação da ferramenta e as dependências necessárias para o funcionamento. Na instalação da ferramenta EasyRA, não foram sentidas dificuldades enquanto os discentes seguiam os procedimentos descritos no manual, indiferentemente do sistema operacional utilizado.

Após uma explicação rápida sobre as funcionalidades disponíveis na ferramenta, foi distribuído uma lista de exercícios, referente ao conteúdo da aula, para os discentes construírem as expressões da álgebra relacional e executarem no SGBD através da ferramenta.

A ferramenta foi avaliada sob a ótica dos discentes, através da aplicação de um questionário, composto por sete perguntas. Os resultados obtidos durante a coleta de informações, através da distribuição do questionário à treze alunos, contribuíram para a definição de novas funcionalidades na ferramenta EasyRA e a sua adequação ao uso acadêmico através das correções e sugestões solicitadas pelos discentes.

Por fim, a aplicação prática da ferramenta EasyRA na disciplina de Banco de Dados auxiliou durante a resolução dos exercícios da álgebra relacional e na compreensão da sintaxe e funcionamento dos operadores.

## 6. Conclusão

Os SGBDs não disponibilizam uma interface destinada a elaboração de expressões da álgebra relacional e as ferramentas existentes não disponibilizam uma interface simples e amigável ao usuário. Buscando solucionar este problema, a ferramenta EasyRA foi desenvolvida de acordo com as necessidades levantadas na análise de quatro ferramentas aplicadas em ambiente acadêmico.

O uso da ferramenta EasyRA, em ambiente educacional, agregou maior qualidade e desempenho ao ensino de álgebra relacional. A ferramenta EasyRA possibilitou a construção das expressões de uma forma simples, auxiliando os discentes durante a resolução dos exercícios propostos na disciplina de Banco de Dados. O uso desta ferramenta possibilitou expandir os conhecimentos em álgebra relacional e SQL, considerando que ao construir uma expressão é possível visualizar a consulta em SQL e o resultado da execução desta consulta no SGBD.

Como trabalhos futuros, é interessante adicionar à ferramenta EasyRA os operadores estendidos. Outra sugestão é desenvolver uma interface que possibilitasse através de *drag and drop* a construção de árvores de expressões, com a visualização da expressão em álgebra relacional e a consulta em SQL. Também é possível a partir de uma expressão em álgebra relacional, visualizar a árvore de expressões gerada. E, por fim, é interessante transformar a ferramenta EasyRA para um ambiente *web*, acrescentando-a plataforma Moodle, desenvolvendo um *plugin*, ou criando um ambiente independente.

## Referências

- Appel, A. P. and Traina Jr., C. (2004). iDFQL - uma ferramenta de apoio ao processo de ensino-aprendizagem da álgebra relacional baseado no construcionismo. *Instituto de Ciências Matemáticas e de Computação-USP. São Carlos, SP.*
- Date, C. J. (2003). *Introdução a sistemas de banco de dados*. Elsevier, Rio de Janeiro, 8 edition.
- Elmasri, R. and Navathe, S. B. (2005). *Sistemas de Banco de dados*. Pearson, São Paulo, 4 edition.
- Elmasri, R. and Navathe, S. B. (2011). *Sistemas de Banco de dados*. Pearson, São Paulo, 6 edition.
- Gesser, C. E. (2003). GALS. TCC (Graduação) - Curso de Bacharelado em Ciências da Computação, Universidade Federal de Santa Catarina, 2003. 150 f. TCC (Graduação) - Curso de Bacharelado em Ciências da Computação, Departamento de Informática e Estatística, Universidade Federal de Santa Catarina. Florianópolis.
- Klein, G. (2014). JFlex user's manual.
- Lautert, L. R. (2010). Implementação de um simulador de consultas em álgebra relacional. Technical report, 2010. 46f. TCC (Graduação) - Curso de Ciência da Computação, Departamento de Centro de Tecnologia, Universidade Federal de Santa Maria, Santa Maria.
- Norvell, T. S. (2011). The JavaCC FAQ.
- Paes, E. L. (2004). Ferramenta didática para o ensino de álgebra relacional. Technical report, Universidade Federal de Santa Catarina - Centro Tecnológico. Departamento de Informática e Estatística. Florianópolis.
- Prates, A., Claro, D. B., Proencia, E., and Queiroz, J. (2013). ProgramAR.
- Silberschatz, A., Korth, H. F., and Sudarshan, S. (2006). *Sistema de Banco de Dados*. Elsevier, Rio de Janeiro, 5 edition.
- Takai, O. K., Italiano, I. C., and Ferreira, J. E. (2005). *Introdução a banco de dados*.