

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE  
DO NORTE

ELIONAI MOURA CORDEIRO

**EDUCASE – SISTEMA ESPECIALISTA APLICADO AO ACONSELHAMENTO  
PEDAGÓGICO UTILIZANDO RACIOCÍNIO BASEADO EM CASOS**

Natal - RN  
2014

ELIONAI MOURA CORDEIRO

**EDUCASE – SISTEMA ESPECIALISTA APLICADO AO ACONSELHAMENTO  
PEDAGÓGICO UTILZANDO RACIOCÍNIO BASEADO EM CASOS**

Monografia apresentada à Banca Examinadora do Trabalho de Conclusão do Curso de Tecnologia em Análise e Desenvolvimento de Software, em cumprimento às exigências legais como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Software.

Orientador: José Antônio da Cunha, MSc.  
Engenharia da Produção - UFSC

Co-orientador: César Analide Rodrigues, DSc.  
Inteligência Artificial – Universidade do Minho

Natal - RN  
2014

ELIONAI MOURA CORDEIRO

**EDUCASE – SISTEMA ESPECIALISTA APLICADO AO ACONSELHAMENTO  
PEDAGÓGICO UTILIZANDO RACIOCÍNIO BASEADO EM CASOS**

Aprovado em: \_\_\_/\_\_\_/\_\_\_

BANCA EXAMINADORA

---

Prof. MSc. José Antônio da Cunha (orientador)  
Instituto Federal do Rio Grande do Norte

---

Prof. DSc. Robinson Luis de Souza Alves  
Instituto Federal do Rio Grande do Norte

---

Prof. MSc. Marília Aranha Freire  
Instituto Federal do Rio Grande do Norte

Dedico este trabalho a meu avô Geraldo Francisco de Moura (in memoriam) e minha avó Maria de Lourdes Alves de Moura (in memoriam) que passavam dias e noites, sempre alertas e num contínuo esforço para garantir a educação e o sustento de nossa família. Muito obrigado.

## **AGRADECIMENTOS**

À Deus, acima de todas as coisas.

À minha Miriam Alves de Moura, um exemplo de moral e civilidade. Mãe, tudo que faço é pensando em você.

A Daniele Montenegro da Silva Barros, minha noiva, meu amor.

Aos meus professores no IFRN, nominalmente aos professores Leonardo Lucena, que sempre trouxe aconselhamentos práticos para minha vida pessoal, Alexandre Gomes e Felipe Aleixo, que através da cobrança e do esforço me ensinaram a crescer.

Aos fiéis e verdadeiros, em especial Eveline Barros, Katia Viana, Giancarlo Silva, Allyson Barros e Sedir Moraes, que foram imprescindíveis nesta jornada e sem os quais eu provavelmente teria desistido da ciência e teria ido morar numa casinha de taipa numa praia deserta sem contatos com a civilização.

A Maria da Salete de Souza, Marcelo Odir, Maria José, Fátima e a todos da Diretoria Acadêmica de Gestão e Tecnologia da Informação, obrigado por me demonstrarem a importância de nunca desistir da incessante busca de conhecimento necessária à vida acadêmica.

À José Antônio da Cunha, professor orientador neste trabalho, e Erica Luana Galvão Torres Gomes, orientadora pedagógica da ETEP, que desde nossas primeiras conversas a respeito do desenvolvimento de uma ferramenta com uso de IA, sempre acreditaram e deram todo o apoio necessário para a realização deste trabalho.

Aquele que só conhece a pesquisa científica por seus efeitos práticos vê depressa demais e incompletamente, a mentalidade de homens que, rodeados de contemporâneos céticos, indicaram caminhos aos indivíduos que pensavam como eles. Albert Einstein, Como Vejo o Mundo, 1953.

## RESUMO

Este trabalho apresenta proposta de aplicação de Raciocínio Baseado em Casos (em inglês, *Case-Based Reasoning*), afim de buscar soluções consolidadas, para aconselhamento pedagógico, dos problemas acadêmicos apresentados, facilitando substancialmente a função de acompanhamento e aconselhamento da equipe pedagógica. Pretendemos através desse trabalho apresentar as motivações que nos levaram a decidir por esta estrutura de algoritmos e dados, de forma a justificar um sistema inteligente e incremental que vai aprendendo as melhores soluções para novos casos a medida que é utilizado, demonstrando as técnicas e tecnologias que permitiram seu desenvolvimento.

Palavras-chave: Raciocínio Baseado em Casos. Inteligência Artificial. Educação. Sistema Especialista

## **ABSTRACT**

This work introduces a proposal scheme for an Inteligente System Applied to Educational Advising using Case-Based Reasoning, to find consolidated solutions before used for the new problems, making easier the task of advising students to the pedagogical staff. We do intend, through this work, introduce the motivation behind the choices for this system structure, justifying the development of an incremental and smart web system who learns bests solutions for new cases when it's used, showing technics and technology.

Keywords: Case-based Reasoning. Artificial Intelligence. Education. Specialist System.



## LISTA DE ILUSTRAÇÕES

FIGURA 1. CICLO DO RACIOCÍNIO BASEADO EM CASOS.....	7
FIGURA 2. DESCRIÇÃO DE UM CASO IDEAL UTILIZANDO A NOTAÇÃO JSON .....	11
FIGURA 3. FLUXOGRAMA SIMPLIFICADO DO ALGORITMO KNN .....	17
FIGURA 4. IMPLEMENTAÇÃO DO CÁLCULO DE DISTÂNCIA DOS CASOS NA LINGUAGEM JAVASCRIPT.....	18
FIGURA 5. MODELO ARQUITETURAL DE APLICAÇÃO CLIENTE-SERVIDOR.....	19
FIGURA 6. ESQUEMA DE <i>TWO-WAY DATA BINDING</i> NA BIBLIOTECA ANGULARJS.....	20
FIGURA 7. MECANISMO EVENT-LOOP DO NODEJS.....	22
FIGURA 8. COMPARAÇÃO DA REPRESENTAÇÃO DE UM MESMO DOCUMENTO USANDO JSON EM TEXTO PLENO E NA SUA REPRESENTAÇÃO EM BINÁRIO, BSON. ....	23
FIGURA 9. INTERFACE COM O USUÁRIO DO SISTEMA. PASSO 1 – INFORMAÇÕES PESSOAIS DO ALUNO .....	24
FIGURA 10. PASSO 2 – SELEÇÃO DE DEMANDAS DO NOVO CASO. ....	25
FIGURA 11. RELATÓRIO DE DEMANDAS MAIS ATENDIDAS. ....	28

## LISTA DE TABELAS E EQUAÇÕES

TABELA 1. DESCRIMINAÇÃO DE CLASSES E SEUS RESPECTIVOS PESOS, EM ORDEM DE RELEVÂNCIA. ....	14
TABELA 2. DISTRIBUIÇÃO DE DEMANDAS NAS RESPECTIVAS CLASSES E PESOS, CONFORME ANÁLISE DA ESPECIALISTA PSICOPEDAGÓGICA POR ORDEM ALFABÉTICA. ....	14
EQUAÇÃO 1 .....	4

## LISTA DE ABREVIATURAS E SIGLAS

BSON	<i>Binary JSON</i> – Notação de Objetos Javascript Binário
ETEP	Equipe Técnico-Pedagógica
HTML	<i>Hypertext Markup Language</i> – Linguagem de Marcação de Hipertextos
IFRN	Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte
JSON	<i>Javascript Object Notation</i> – Notação de Objetos Javascript
MIT	<i>Massachusetts Institute of Technology</i>
NoSQL	Não-somente SQL
RBC	Raciocínio Baseado em Casos
SGBD	Sistema de Gerenciamento de Banco de Dados
SQL	<i>Structured Query Language</i> – Linguagem de Consultas Estruturadas
SSOT	<i>Single-Source-Of-Truth</i> – Recurso Único da Verdade.
TCC	Trabalho de Conclusão de Curso
URL	<i>Universal Resource Location</i> – Localização Universal de Recursos
W3C	<i>World Wide Web Consortium</i>
Web	Plataforma Web e suas tecnologias

## SUMÁRIO

<b>1.</b>	<b>INTRODUÇÃO</b>	<b>1</b>
<b>1.1.</b>	<b>OBJETIVOS</b>	<b>2</b>
1.1.1.	OBJETIVO GERAL	2
1.1.2.	OBJETIVOS ESPECÍFICOS	2
<b>1.2.</b>	<b>CONTEXTUALIZAÇÃO</b>	<b>3</b>
<b>1.3.</b>	<b>METODOLOGIA</b>	<b>4</b>
<b>1.4.</b>	<b>ESTRUTURA DO TRABALHO</b>	<b>6</b>
<b>2.</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>7</b>
<b>2.1.</b>	<b>RACIOCÍNIO BASEADO EM CASOS</b>	<b>7</b>
2.1.1.	CASO	8
2.1.2.	BASE DE CASOS	9
2.1.3.	INDEXAÇÃO	11
2.1.4.	RECUPERAÇÃO	12
2.1.5.	REUTILIZAÇÃO	12
2.1.6.	REVISÃO E RETENÇÃO	13
<b>2.2.</b>	<b>SIMILARIDADE E O PROBLEMA DE AGREGAÇÃO</b>	<b>13</b>
<b>2.3.</b>	<b>TECNOLOGIAS E MÉTODOS EMPREGADOS</b>	<b>15</b>
2.3.1.	K-VIZINHOS MAIS PRÓXIMOS (KNN)	16
<b>3.</b>	<b>DESENVOLVIMENTO DO PROJETO</b>	<b>19</b>
<b>3.1.</b>	<b>ARQUITETURA DO SISTEMA</b>	<b>19</b>
3.1.1.	INTERFACE COM USUÁRIOS	20
3.1.2.	SERVIDOR DA APLICAÇÃO	21
3.1.3.	PERSISTÊNCIA DOS DADOS	22
<b>3.2.</b>	<b>RESULTADOS</b>	<b>23</b>
<b>4.</b>	<b>Considerações Finais</b>	<b>29</b>
<b>4.1.</b>	<b>CONCLUSÃO</b>	<b>29</b>
<b>4.2.</b>	<b>SUGESTÕES E RECOMENDAÇÕES</b>	<b>30</b>
	<b>REFERÊNCIAS</b>	<b>31</b>
	<b>GLOSSÁRIO</b>	<b>33</b>
	<b>ANEXO I – FICHA DE ATENDIMENTO PEDAGÓGICO</b>	<b>35</b>
	<b>ANEXO II - LICENSE</b>	<b>36</b>

# 1. INTRODUÇÃO

Segundo Blay Whitby (2003), Inteligência Artificial (IA) é o estudo do comportamento inteligente (em homens, animais e máquinas) e a tentativa de encontrar formas pelas quais esse comportamento possa ser transformado em qualquer tipo de artefato por meio da engenharia. Entre os diversos processos de IA está o aprendizado de máquinas, que por sua vez pode ser seccionado em aprendizado supervisionado e aprendizado não-supervisionado.

Aprendizado Não-supervisionado “cobre aqueles casos em que não conhecemos o padrão, ou nem mesmo se há um padrão” (Whitby 2003). Entre as diversas técnicas e algoritmos utilizados neste modelo de aprendizado, inclui-se os algoritmos genéticos e de *clustering*, entre outros.

Por sua vez, o Aprendizado Supervisionado é utilizado, por exemplo, para resolver classificação de casos ou regressões, para reconhecer spams ou ainda, no reconhecimento de impressões digitais. Uma das técnicas amplamente utilizadas para este tipo de aprendizado é um algoritmo para geração de uma árvore de decisão.

Nagy *et al* (2013), afirmam que o algoritmo de aprendizado com árvore de decisão foi proposto primeiramente por Ross Quinlan como uma extensão do algoritmo ID3. Pode lidar com atributos contínuos e discretos, além de conjunto de dados com valores de atributos desconhecidos.

Como podemos ver nesta breve introdução, existe uma ampla gama de abordagens de aprendizado de máquinas e de IA. Esse trabalho não aprofunda-se nos diversos subcampos de IA e seus algoritmos. Ao invés disso, debruça-se sobre um conjunto de processos, que por sua vez compõem a técnica comumente denominada Raciocínio Baseado em Casos (RBC).

Aliamos esta técnica ao algoritmo de *clustering* denominado n-Vizinhos Mais Próximos (kNN) e à plataforma Web e suas tecnologias na construção de um Sistema Especialista. Este sistema deve utilizar os princípios e conceitos de Inteligência Artificial, para servir como ferramenta de apoio à decisão no contexto educacional, notadamente ao aconselhamento pedagógico dos alunos.

RBC é um conjunto de técnicas para o funcionamento de um sistema inteligente, tendo como centro de operações, uma base de conhecimento prévio, composta de experiências contextualizadas, que descrevem os problemas e suas respectivas

soluções, chamada Base de Casos. Estas soluções podem ser sugeridas (reutilização) para serem aplicadas a quaisquer novos casos, ou problemas, que sejam apresentados, a partir da relação de similaridade (recuperação) do problema apresentado com os casos existentes, podendo inclusive, se servir destes novos casos (retenção) ampliar a base de conhecimento (Watson 1997).

## **1.1. Objetivos**

### **1.1.1. Objetivo Geral**

O principal objetivo deste Trabalho de Conclusão de Curso é aplicar o Raciocínio Baseado em Casos (RBC) como técnica de Inteligência Artificial, no desenvolvimento de um sistema especialista que auxilie uma equipe de acompanhamento pedagógico na tarefa de aconselhamento de alunos, através da proposição de soluções aos problemas atuais baseando-se no conhecimento adquirido com os casos anteriores que por sua vez foram inseridos na Base de Casos.

### **1.1.2. Objetivos Específicos**

- Apresentar proposta de Sistema Especialista como solução de apoio à decisão no âmbito de aconselhamento pedagógico.
- Desenvolver uma arquitetura ideal utilizando linguagem de programação e ambiente de desenvolvimento e produção amplamente utilizado.
- Demonstrar Raciocínio Baseado em Casos como metodologia de Inteligência Artificial para aprendizado de máquina como alternativa prática para desenvolvimento de sistema especialista.
- Utilizar a plataforma web, a linguagem de programação Javascript e as notações JSON (Javascript Object Notation) e BSON (Binary Javascript Object Notation) no desenvolvimento do sistema proposto.
- Demonstrar os resultados do Estudo de Caso utilizando Raciocínio Baseado em Casos com dados obtidos com auxílio da equipe de Orientação Pedagógica do Instituto Federal de Educação, Ciência e Tecnologia do RN – IFRN.

## 1.2. Contextualização

Notamos a necessidade do desenvolvimento de uma ferramenta de apoio neste contexto de aconselhamento pedagógico, em entrevista realizada com a pedagoga Erica Luana Galvão Torres Gomes, da Equipe Técnica Pedagógica – ETEP da Diretoria Acadêmica de Gestão e Tecnologia da Informação do IFRN, onde foi exposto como os procedimentos de atendimento pedagógico, e de que forma os registros de casos evoluíram nos últimos anos, porém sem ainda atingir a problemática central do atendimento que é a qualificação das informações e a transformação do conhecimento tácito de cada membro da equipe em conhecimento explícito de toda ETEP.

Em anos anteriores a 2012, o atendimento sequer contava com um documento padronizado para registro dos casos, sendo posteriormente criada uma ficha de atendimento, que pode ser encontrada no Anexo I – Ficha de Atendimento Pedagógico. Em um momento posterior, foi desenvolvido um formulário utilizando a plataforma Google Docs, que gerava uma planilha compartilhada entre os membros da ETEP as principais informações cadastradas.

A pedagoga relata na entrevista que:

“O acompanhamento dos alunos acontecia por verificação das notas nos boletins no final dos semestres, escrita de situações em livro ATA (registro das situações problemas) ou agenda da equipe (Até 2011). A partir de 2012, adotou-se um modelo de ficha individual que facilitava o acompanhamento da evolução dos encaminhamentos e das soluções e permitia identificar recorrências. Esse modelo foi importante, também, pela possibilidade que trouxe de partilhar informações entre as pedagogas da Diretoria sobre os atendimentos. [...] Verificou-se já um grande avanço com as fichas, mas começou-se a perceber que algumas situações de atendimento eram bem comuns (problemas que requeriam acompanhamento psicológico, por exemplo). E como verificar isso? Olhando de ficha em ficha a natureza do problema? Sem falar que requer um tempo para descrever a situação na ficha, descrever o problema, descrever o encaminhamento dado e outras informações importantes.”

A pedagoga afirma ainda a necessidade de uma ferramenta que tornasse os dados mais “palpáveis” e mais “úteis”, de forma que fossem permitidos produzir relatórios e gráficos que exibissem as problemáticas mais comuns, a evolução dos casos e ainda a evolução dos encaminhamentos abordados e as melhores soluções ou ainda mais úteis. A abordagem mais eficiente discutida para abordar neste trabalho foi a reutilização das soluções anteriores nas novas situações problema, aplicando assim o conceito de IA com o RBC.

### 1.3. Metodologia

Como primeiro passo para a estruturação do sistema especialista, que utilize RBC, no auxílio de aconselhamento pedagógico aos alunos, realizamos uma entrevista com a Equipe Técnico Pedagógica – ETEP, da diretoria Acadêmica de gestão e Informática, e propusemos que fosse feito levantamento das principais situações que demandam atenção, bem como a classificação destas situações que se baseasse numa maior ou menor necessidade de atenção por parte dos técnicos da ETEP. Como resultado deste levantamento, foram classificadas um total de 26 demandas.

Estas demandas, foram distribuídas em cinco grupos, ou classes, de forma que possibilitou a determinação de um peso em ordem de relevância, de acordo com a avaliação da especialista pedagógica, onde o maior peso determina maior importância e necessidade de atenção.

Seguiu-se no processo de desenvolvimento deste trabalho, definir um método de cálculo de similaridade, ou dissimilaridade, entre os casos, de forma a tornar os processos de indexação e recuperação efetivamente funcionais. Foi decidido a utilização da Distância Euclidiana para calcular as distância entre os casos, onde calculamos a distância para cada atributo que representa uma demanda entre os casos.

$$E(x, y) = \sqrt{\sum_{a=1}^n (x_a - y_a)^2}$$

Equação 1

A Equação 1 descreve o cálculo euclidiano para medidas de dissimilaridade, onde  $E(x, y)$  é a distância e  $x$  e  $y$  são dois vetores com  $n$  atributos numéricos.

A Distância Euclidiana é uma medida de dissimilaridade, que representa um segmento de reta com a menor distância entre dois pontos, sendo escolhida para este trabalho por ser aplicável em espaços multidimensionais, com valores discretos ou contínuos. Os vetores a serem utilizados nos cálculos deste trabalho são o conjunto de demandas de cada caso registrado na base de casos, os atributos corresponde às demandas discriminadas no levantamento e classificação descrito anteriormente.

A Base de Casos de referência, tem um total de 53 casos, todos bem definidos, completos e originados de experiências do mundo real. Para o levantamento dos



mesmos, contamos com o apoio de profissionais especialistas em orientação pedagógica, para realizar o desenvolvimento da lista de demandas e principais soluções aplicadas aos casos, os quais tiveram conhecimento durante o primeiro período letivo do ano 2014, no Instituto Federal de Educação, Ciência e Tecnologia do RN – IFRN.

Na definição da Base de Casos, cada demanda está representada como um atributo do caso, sendo que, entre os casos elencados, alguns possuem mais de um atributo, porém, dados os limites da nossa fonte de informações, nem todos os atributos classificados estão contemplados na Base de Casos.

Estes atributos selecionados pela especialista, podem ser avaliados de forma condicional a estarem relacionados com o caso, e foram valorados numericamente com 0 (ausente) e 1 (presente). Ou seja, caso o atributo exista ele recebe o valor 1 (um), se não existir, o valor 0 (zero). Os atributos que não forem respondidos ou que não são de ciência do orientador pedagógico recebem igualmente o valor 0.

Outros atributos também foram selecionados, de forma a dar uma maior especificidade aos casos e que servirão para outros propósitos, como a geração de relatórios e fichas de acompanhamento pedagógico etc. São atributos multivalorados que receberam código para seus valores, de acordo com a forma de classificação.

Por exemplo, o atributo SEXO, recebe valor 0 para masculino e valor 1 para feminino. Dados com valores escalares absolutos, ou com valores textuais, como por exemplo, o percentual de rendimento acadêmico, ou a quantidade de salários da renda familiar, recebem seu valor nominal.

Essa distribuição de atributos é importante caso deseje-se utilizar o processo de mapeamento, onde serão aplicadas funções para calcular a distancia (dissimilaridade) entre os casos. Não deve se confundir esta função de mapeamento com o modelo de programação MapReduce, proposto por Dean e Ghemawat (2004), que está intrinsecamente ligado ao processamento e geração de dados massivos, sendo sua principal utilização em grandes massas de dados e processamento em clusters de computadores.

MapReduce foi inspirado por estruturas similares no mundo da linguagem funcional, onde *map* é uma função que é aplicada em cada membro de uma coleção e *reduce* é uma função aplicada através da coleção (Tiwari 2011).

É possível aplicar MapReduce na Base de Casos, e o Sistema Gerenciador de Banco de Dados – SGBD – MongoDB tem suporte nativo a esta técnica, porém, já que

nossa base de dados é significativamente limitada, esta é uma recomendação para trabalhos futuros que queiram estender a aplicação deste trabalho ou aplica-lo a uma base de casos de amplitude massiva.

#### **1.4. Estrutura do trabalho**

Este trabalho está estruturado em 4 capítulos, incluindo esta introdução.

O capítulo 2, trata da fundamentação teórica do Raciocínio Baseado em Casos, cada um dos elementos, ou fases, que o compõe. Falamos também a respeito da similaridade e os problemas que envolvem técnicas de agregação. Finalizamos a fundamentação teórica explicando sobre as principais tecnologias e métodos empregados.

O capítulo 3 trata do desenvolvimento do trabalho, apresentamos a arquitetura do sistema desenvolvido e os resultados analíticos e conjecturais levantados como conclusões da aplicação do sistema. São discutidas também algumas particularidades sobre a base de casos e suas características.

No capítulo 4, concluímos este trabalho abordando o conhecimento adquirido e algumas recomendações para esta matéria que podem ser aplicados para trabalhos futuros.

## 2. FUNDAMENTAÇÃO TEÓRICA

Sistemas baseados em conhecimento são sistemas de Inteligência Artificial que raciocinam sobre suas possíveis ações no mundo fazendo uso de dois mecanismos principais: 1) Base de conhecimento e 2) Mecanismo de inferência (Abel e Castilho 1996).

As principais diferenças entre estes e os sistemas convencionais está na organização dos dados, na separação do conhecimento e método de solução. Sistemas baseados em conhecimento utilizam soluções de raciocínio que podem ser baseados em regras, em modelos, em casos etc.

### 2.1. Raciocínio Baseado em Casos

Wangenheim *et al* (2013) afirmam que dá-se o nome de Raciocínio Baseado em Casos (RBC), à técnica de Inteligência Artificial, como um conjunto de atividades que auxilia na resolução de problemas, propondo soluções incontestavelmente utilizadas e documentadas, ao recuperar e adaptar experiências passadas – chamadas casos – armazenadas em uma Base de Casos. Um novo caso é resolvido com base na adaptação de solução de casos similares já conhecidos (Wangenheim *et al* 2013).

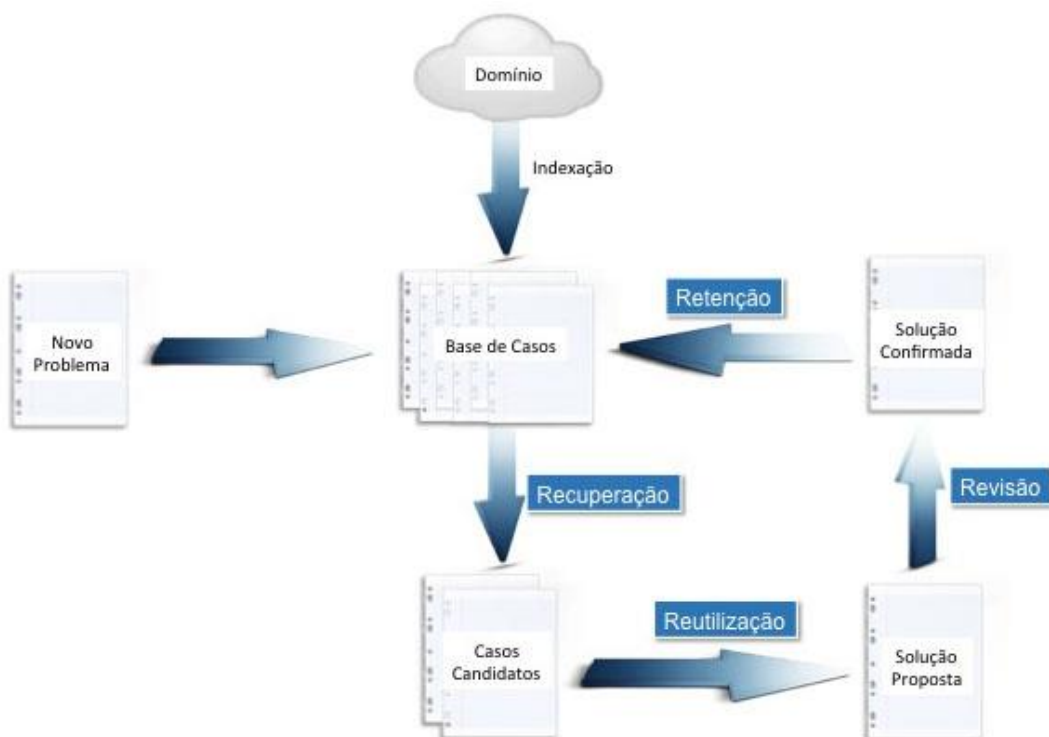


Figura 1. Ciclo do Raciocínio Baseado em Casos

Segundo Watson (1997), RBC pode ser resumido como um processo cíclico que compreende os quatro R's:

- Recuperação dos casos mais similares;
- Reutilização para solucionar o caso;
- Revisão da solução proposta;
- Retenção (aprendizado) da nova solução como parte de um novo caso.

Para tanto é necessário que exista uma base de casos e um mecanismo de inferência que através de similaridade apresente soluções para o caso apresentado fazendo uso, quando necessário, de adaptação dos casos existentes ao novo que foi apresentado. A Figura 1 descreve o processo cíclico que compõe as fases do RBC.

### **2.1.1. Caso**

“Caso é uma unidade contextualizada de conhecimento que representa experiência” (Watson 1997), e deve conter: o caso que descreve o estado das coisas quando ocorreu o caso e a solução que foi estabelecida para o caso. Para Abel e Castilho (1996) e Watson (1997), um caso diferencia-se das outras experiências armazenadas contribuindo com uma melhor solução para o problema apresentado; um caso não é genérico, pois suas informações são contextualizadas a uma determinada experiência; um caso é uma instância concreta de uma experiência.

Como primeiro passo na construção da base de casos, está a aquisição e representação dos casos. No quesito aquisição, podemos encontrar diversos cenários (Abel e Castilho 1996) que podem incluir:

- Os casos não estão disponíveis em uma fonte externa;
- Os casos estão disponíveis parcialmente em uma fonte externa;
- Os casos estão disponíveis e contém erros;
- Os casos estão disponíveis e estão corretos.

Depois de coletados os casos, deve-se definir a forma como serão representados. Não existe um consenso a respeito de que tipo de dados ou quais informações devem estar contidas em um caso, mas devemos ter em mente dois aspectos significativos destas informações: a facilidade de aquisição da informação e a funcionalidade desta.

Um caso ideal pode ser estruturado de forma a conter os seguintes dados de forma estruturada: uma descrição dos aspectos relevantes do caso – neste trabalho, as demandas que são objeto de atenção por parte da ETEP; o contexto no qual o caso está inserido – pode ser um texto descritivo ou palavras-chave que associam o caso com um contexto específico; a descrição da solução associada ao caso – neste trabalho, descrevemos solução como o conjunto de encaminhamentos dados a cada atendimento pedagógico inserido na base de casos; e a avaliação da solução empregada. Na Figura 2, descrevemos a composição de um caso ideal com demandas e encaminhamentos.

Se, para efeito de exemplo, pegarmos o domínio do sistema proposto, que é o aconselhamento pedagógico de estudantes, um caso ideal poderia ser descrito como um documento onde existem registros com informações sobre cada estudante: nome, matrícula, turma, período etc. Estes seriam dados de contextualização.

Os aspectos relevantes do caso, geralmente nomeados de feição ou atributos, seriam as demandas de atenção, ou problemas que este aluno possa estar passando ou gerando para outrem. Ou seja, se o aluno está tendo problemas de relacionamento com outros alunos e/ou professores, se existe algum conflito familiar ou afetivo, se algum problema de ordem psicológica, econômica ou social está interferindo na aprendizagem e no convívio do aluno etc.

A descrição das soluções empregadas incluiriam todas as ações tomadas pela orientação pedagógica, incluindo encaminhamentos, acompanhamento etc. Encaminhamentos podem ser: encaminhar efetivamente o aluno ao atendimento médico ou psicológico da escola, fazer visita à residência do aluno, elaborar rotinas de estudo, chamar os pais ou responsáveis para conversar etc.

A avaliação da solução empregada poderia ser a descrição, ou consideração por parte da orientação pedagógica, sobre os resultados obtidos com a solução descrita e como estas contribuíram para o caso anterior, de forma a embasar a decisão de retomar esta como solução para o novo caso-problema.

### **2.1.2. Base de Casos**

A estrutura de dados composta de casos é chamada Base de Casos, pode ser uma base de dados em forma de documentos, pares de chave-valor, objetos, predicados, redes semânticas, etc. O mais utilizado é o caso representado como um

objeto, ou um frame, embora seja comum combinar um ou mais formalismos entre os citados (Wangenheim *et al* 2013).

Na estrutura de dados orientado a documentos, uma notação usada amplamente e que será usada na solução proposta é a JSON (Javascript Object Notation – Notação de Objetos Javascript), que permite descrição de objetos como documentos, agrupados em conjuntos de chave-valor, onde o valor pode ser um documento ou objeto agregado e subsequentemente. O Formato de Comunicação de Dados JSON é descrito pela RFC 4627 (Crockford 2006), como sendo independente de linguagem, leve, baseado em texto e sua origem deriva dos padrões da linguagem de programação ECMAScript.

Este trabalho utiliza a notação JSON, e sistema de banco de dados orientado a documentos para representação de casos. Retomando o exemplo do item anterior, poderíamos escrever o documento que representa um exemplo de caso ideal, utilizando a notação JSON, da maneira descrita na Figura 2.

```
{
  nome: {type: String, required: true},
  sexo: Number,
  telefone: {type: String},
  email: {type: String},
  matricula: {type: String, required: true},
  curso: {type: String, required: true},
  periodo: {type: Number, required: true},
  data: {type: String, required: true},
  observacoes: {type: String},
  demandas: {
    desequilibrioPsicologico: Number,
    orientacaoSecular: Number,
    orientacaoPedagogica: Number,
    muitasFaltas: Number,
    problemaSocioEconomico: Number,
    atrasosConstantes: Number,
    bullying: Number,
    desmotivacaoRendimento: Number,
    desmotivacaoCurso: Number,
    dificuldadeAprendizagem: Number,
    problemaDisciplinarLeve: Number,
    problemaDisciplinarMedio: Number,
    problemaDisciplinarGrave: Number,
    problemaComportamento: Number,
    conflitoFamiliar: Number,
    problemaRelacionamentoPai: Number,
    problemaRelacionamentoCasa: Number,
    problemaRelacionamentoAluno: Number,
    problemaRelacionamentoProfessor: Number,
    situacaoAbuso: Number,
    situacaoExclusao: Number,
    situacaoTimidez: Number,
    separacaoPais: Number
  },
}
```

```

encaminhamentos: {
  psicologiaEscolar: Number,
  servicoSocial: Number,
  setorMedico: Number,
  chamarPais: Number,
  chamarProfessores: Number,
  acionarCoordenacao: Number,
  advertencia: Number,
  repreensao: Number,
  suspensaoEscolar: Number,
  trancarPeriodoCompulsorio: Number,
  trancarPeriodoVoluntario: Number,
  visitaDomiciliar: Number,
  atendimentoDomiciliar: Number,
  rotinaEstudo: Number,
  centroAprendizagem: Number,
  mudarTurno: Number,
  conversar: Number
},
contexto: {type: String, required: true},
atendidoPor: {type: String, required: true},
classe: {type: String, required: true}
}

```

Figura 2. Descrição de um caso ideal utilizando a notação JSON

### 2.1.3. Indexação

A indexação é um processo importante e decisivo dentro do RBC, assim como numa base de dados convencional, porém não necessariamente, todas as informações relevantes deverão ser indexadas. As informações indexadas servirão para acelerar o processo de recuperação enquanto que informações não-indexadas podem servir de contextualização para a decisão de reutilização do caso ou em outros aspectos do sistema de informação e não somente na recuperação.

Devemos ter em mente que os casos serão indexados de forma automatizada e que existem cada dia mais métodos de indexação. Indexação por medidas ou especificações, baseadas em diferenças, por similaridade e métodos indutivos de aprendizado são alguns exemplos que podemos utilizar na indexação. Este trabalho utilizará o mesmo algoritmo do k-Vizinho mais Próximo (neste trabalho abreviado para kNN, do inglês *k-nearest neighbor*) na indexação e na recuperação de casos.

Como identificar quais são as informações que deverão indexar os casos? Não existe uma resposta simples de se obter, porém devemos nos ater a algumas características que índices devem possuir: ser preditivo; estar direcionado ao caso ao qual será aplicado; ser abstrato o suficiente para que através de extrapolação possa ser usado; ser concreto o suficiente para ser reconhecido no futuro (Wangenheim *et al* 2013).

#### **2.1.4. Recuperação**

A recuperação deve utilizar um algoritmo que verifique os clusters que tem maior similaridade com o caso-problema apresentado e permita otimizar a reutilização e revisão do caso. Essa recuperação pode se dar como: uma busca por proximidade, em processos como o proposto por este trabalho; busca hierárquica, utilizando processos semelhantes ao algoritmo ID3, ou derivados deste, conforme descrito por Nagy et al (2013); existem ainda busca por simulação paralela, busca serial etc.

Ao contrário de um banco de dados relacional, uma base de casos requer que a recuperação dos casos se dê de forma avaliada, envolvendo heurísticas e julgamentos sobre a relevância do dado recuperado. Entre os algoritmos mais relevantes e divulgados na recuperação de casos estão: Recuperação de Padrões, Algoritmo de Indução e Algoritmo de Vizinhança ou k-Vizinhos mais Próximos.

#### **2.1.5. Reutilização**

Raciocínio Baseado em Casos aplica-se na solução de problemas pela possibilidade de que, caso haja conhecimento do domínio aplicado ou das constantes existentes neste, pode-se aplicar uma técnica interpretativa através da qual podemos reutilizar uma solução conhecida de um caso semelhante, ou adaptar uma ou mais soluções conhecidas para propor a solução para o caso apresentado. Existem duas etapas no processo de reutilização: Adaptação e Justificação e Crítica (Abel e Castilho 1996).

Adaptação ocorre quando a solução, ou conjunto de soluções propostas, não correspondem especificamente às ações que serão tomadas, porém são feitos ajustes na solução para que a mesma sirva ao problema dado. Já na justificação e crítica é necessária uma busca no domínio por uma combinação dos atributos que porventura possam causar problemas na solução, e corrigir estas distorções.

Retomando-se o exemplo do domínio educacional, e supondo-se que uma das soluções propostas seja o encaminhamento para orientação psicológica com o psicólogo da escola. Agora imagina-se que a escola em questão não possua um profissional disponível, deverá então, ser feita adaptação na solução para que, o aluno, seja encaminhado a um psicólogo que fará o acompanhamento externo ao ambiente escolar.



Qualquer que seja, este trabalho conta com a interação do usuário que irá determinar e irá realizar os ajustes, críticas ou adaptações nas situações que julgar necessárias.

### **2.1.6. Revisão e Retenção**

Revisão é o processo de julgar quão boa uma solução é. Neste passo a solução deverá ter sido experimentada no mundo real e recebe um feedback sobre o que ocorreu e se foi correspondente ao esperado, ou se gerou alguma situação nova como resultado. Neste caso, será de suma importância receber o feedback sobre a anomalia com descrição detalhada dos fatores e como prevenir o surgimento de novas anomalias.

Essas informações podem ser: explicação das diferenças, justificativas diferentes, comparação de possibilidades alternativas e avaliação. Se o conjunto formado pelo problema proposto mais a solução avaliada resultar em um caso com características distintas, este novo caso é retido e indexado à Base de Casos.

## **2.2. Similaridade e o problema de agregação**

Uma questão generalizada enfrentada por pesquisadores das mais diversas áreas é como organizar os dados observados em estruturas manipuláveis, isto é, como desenvolver taxonomia. Em vistas de resolver este problema, diversas técnicas e algoritmos vem sendo propostos. Ao conjunto destas técnicas, dá-se o nome de Análise Cluster, ou *clustering* (agrupamento), que faz menção a característica comum de agrupamento das características ou variáveis.

Agrupamentos são encontrados nas mais diversas situações no nosso dia-a-dia. Por exemplo, no ambiente escolar, existem os diversos grupos de estudantes separados por sala, ou em um restaurante, pode-se separar as pessoas em grupos pelas mesas que compartilham.

Análise *cluster* é uma ferramenta de análise exploratória de dados que pega diferentes objetos e os separa em grupos, de forma que o grau de associação (similaridade) entre dois objetos é máxima se eles pertencem ao mesmo grupo e mínima caso contrário. Dessa forma, análise *cluster*, pode ser usada para descobrir estruturas de dados sem recorrer a uma explicação ou interpretação.

Importante observar que a similaridade entre os membros das classes será cada vez menor quanto maior o nível de agregação. Um exemplo importante de *clustering*, é

a classificação que os biólogos fazem dos homens separando-os em grupos (clusters) diversos, onde homem faz parte dos primatas, mamíferos, e animais, por exemplo. Homem tem similaridade maior com chimpanzés (primatas) do que com jacarés (animais).

A taxonomia aplicada neste trabalho baseou-se nas demandas que compõe os casos, e nos grupos (classes) que podem ser construídos a partir destas informações. O resultado foi a separação em 5 classes de atributos, separados por aspectos do mundo real e a influência na vida acadêmica dos alunos.

Tabela 1. Descriminação de classes e seus respectivos pesos, em ordem de relevância.

<b>Classe</b>	<b>Peso</b>
Dificuldade cognitiva/psicopedagógica	5
Conflitos psicológicos	4
Problemas de relacionamento/comportamento no âmbito escolar	3
Problemas disciplinares	2
Dificuldade de relacionamento afetivo/familiar	1

Determinou-se também o peso das classes, de 1 a 5, de forma que as demandas que necessitam de atenção maior por parte do profissional pedagogo esteja contida nas classes de peso maior, enquanto que demandas consideradas menos relevantes pertencem a classes que possuem peso menor. A Tabela 1 relaciona as classes e os respectivos pesos.

Na Tabela 2, estão descritas todas as demandas que foram elencadas, no total de 26, que serão os atributos calculados na busca por similaridade entre os casos, relacionando estes atributos com os pesos que receberam de acordo com a análise pedagógica, pela classificação anteriormente determinada.

Tabela 2. Distribuição de demandas nas respectivas classes e pesos, conforme análise da especialista psicopedagógica por ordem alfabética.

<b>Demandas</b>	<b>Classe</b>	<b>Peso</b>
Conflito familiar	E	1
Pais em separação	E	1
Problema de relacionamento com a mãe	E	1
Problema de relacionamento com o pai	E	1
Problema de relacionamento em casa	E	1
Problema de ordem socioeconômica do aluno ou da família	E	1
Problema de ordem disciplinar grave	D	2
Problema de ordem disciplinar leve	D	2

Problema de ordem disciplinar médio	D	2
Problema de comportamento	C	3
Problema de relacionamento aluno x aluno	C	3
Problema de relacionamento professor x aluno	C	3
Bullying	B	4
Conflito com relação a opção sexual	B	4
Conflito devido a situação relacional ou afetiva	B	4
Desmotivação pela opção de curso	B	4
Desmotivação por baixo rendimento	B	4
Necessidade de orientação secular (pessoal/não acadêmica)	B	4
Situação de abuso (moral, sexual etc.)	B	4
Situação de exclusão em sala	B	4
Situação relacionada à timidez	B	4
Atrasos constantes	A	5
Desequilíbrio ou problema de ordem psicológica	A	5
Dificuldade de aprendizagem em disciplina	A	5
Muitas faltas	A	5
Necessidade de orientação pedagógica	A	5

### 2.3. Tecnologias e métodos empregados

Este trabalho, propõe uma aplicação web, desenvolvida com a linguagem de marcação HTML5, juntamente com a linguagem de programação Javascript para a construção da interface com o usuário e a lógica de negócio.

A Web pode ser descrita como uma plataforma que utiliza um conjunto de técnicas e tecnologias, entre eles a internet e o hipertexto, para criar uma rede de informações de amplo alcance. As informações podem estar em formato de texto, áudio, vídeo, imagens etc., e podem ser acessados por um programa de computador ou dispositivo móvel conectado à rede.

HTML5 é uma nova versão de HTML (*Hypertext Markup Language*, ou em português, Linguagem de Marcação de Hipertextos), que é comumente utilizada para criar marcações e estruturar documentos, além de permitir o desenvolvimento de forma a maximizar a semântica dos mesmos. Assim, sistemas baseados na web podem se beneficiar da estrutura de construção dos documentos sendo capazes de “entender” o significado de cada uma das suas partes. (Ferreira 2013)

No dia 16 de setembro de 2014, o grupo de trabalho em HTML, ligado ao World Wide Web Consortium – W3C, que é o consórcio das principais empresas de tecnologias ligadas à web, e que são responsáveis pela padronização das tecnologias para esta plataforma, publicou a CFR-HTML5.

CFR é a sigla em inglês de *Call for Review*, que pode ser traduzido como uma chamada para revisão pela comunidade. Esta publicação é uma especificação da linguagem HTML, e foi publicada para que a comunidade de desenvolvedores e as empresas que lidam com a web possam participar do processo de desenvolvimento através de comentários à mesma.

A linguagem Javascript, na sua especificação mais recente, é uma linguagem de programação leve, suficientemente complexa, com recursos de orientação a objetos, não-tipada, que permite sua aplicação tanto compilada em servidores e máquinas virtuais, quanto como aplicações clientes, sendo interpretadas em navegadores de internet comuns.

Flanagan (2004), afirma que o núcleo geral da linguagem foi incorporado no Netscape, Internet Explorer e outros navegadores web, e aprimorado para programação web com adição de objetos. Javascript inclui ainda mecanismos de herança orientado a objeto, laços e instruções com sintaxe que lembram outras linguagens, como C, Java, Perl entre outras.

A linguagem Javascript é suportada seguindo os padrões publicados pela ECMA International, e sua mais recente especificação é a ECMA-262 (ECMA International 2011), na versão 5.1, publicada em junho de 2011.

### **2.3.1. k-Vizinhos mais Próximos (kNN)**

O sistema proposto por este trabalho faz uso do algoritmo intitulado Algoritmo dos k-Vizinhos mais Próximos (*k-nearest neighbor*, no original em inglês). O fluxograma simplificado do algoritmo é descrito na Figura 3.

Em princípio, o usuário fornece um conjunto de descrições do problema atual. Então aplica-se uma função de cálculo para a medida de similaridade escolhida à todas as instâncias de conhecimento contidas na base de casos. Em seguida, ordena-se os casos por ordem de similaridade, retornando um ou mais casos, até o limite estipulado por k, entre os mais próximos (Wangenheim *et al* 2013).

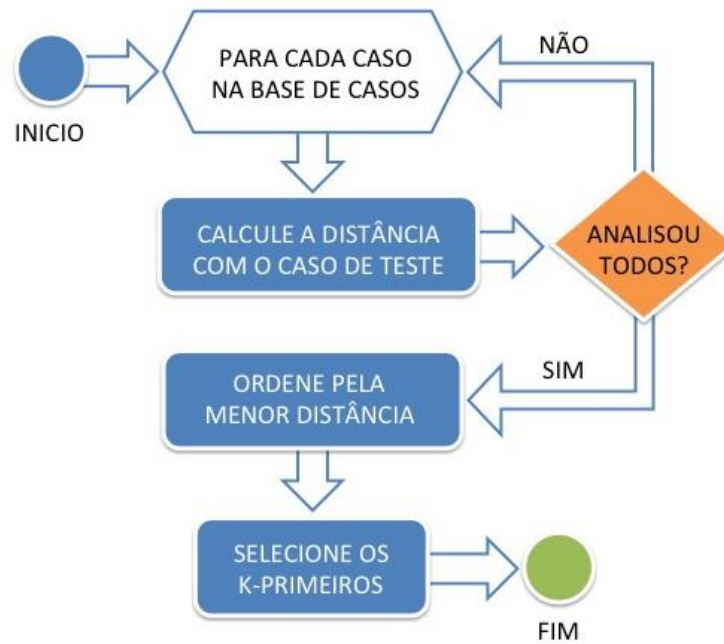


Figura 3. Fluxograma simplificado do algoritmo kNN

Uma desvantagem clara dessa abordagem é o alto valor computacional do algoritmo, pois é necessário comparar os dados de testes de cada novo caso apresentado com todos os outros presentes na base de casos, recalculando as distâncias cada vez que é utilizado.

As principais características que levam a utilizar-se este algoritmo se encontra na sua facilidade de implementação e na possibilidade de adaptação para diferentes entradas com múltiplas dimensões, inclusive.

Uma etapa natural deste algoritmo é a classificação da entidade testada, inferindo a que grupo o conjunto de dados de teste pertence com base na classe dominante à qual pertencem os vizinhos mais próximos. Neste trabalho, esta etapa do algoritmo é descartada, em vista de que optou-se por agrupar os casos de acordo com a classe a qual pertence a demanda de maior peso presente no caso.

A Figura 4 exhibe uma implementação de parte deste algoritmo, com ênfase no cálculo euclidiano de distâncias.

O primeiro passo, é definir o peso de cada entrada, conforme classificação da Tabela 2 na seção 2.2. Em seguida, percorre-se os vetores de entrada, ou seja, o caso-problema (caso) e um caso presente na Base de Casos (vizinho), multiplicando cada atributo pelo peso correspondente. Segue-se então a soma dos quadrados das diferenças de todos os atributos e por fim divide-se pelo intervalo de demandas e faz-se o cálculo da raiz quadrada, retornando seu valor que, conforme o cálculo euclidiano

normalizado, é a distância entre os casos. Repete-se o cálculo para cada entrada registrada na Base de Casos.

```
// Realiza o cálculo euclidiano para distâncias
var calcularDistancias = function (caso, vizinho) {
  var pesos = {
    atrasosConstantes: 5,
    desequilibrioPsicologico: 5,
    dificuldadeAprendizagem: 5,
    muitasFaltas: 5,
    orientacaoPedagogica: 5,
    bulling: 4,
    conflitoOpcaoSexual: 4,
    conflitoRelacionalAfetivo: 4,
    desmotivacaoCurso: 4,
    desmotivacaoRendimento: 4,
    orientacaoSecular: 4,
    situacaoAbuso: 4,
    situacaoExclusao: 4,
    situacaoTimidez: 4,
    problemaComportamento: 3,
    problemaRelacionamentoAluno: 3,
    problemaRelacionamentoProfessor: 3,
    problemaDisciplinarGrave: 2,
    problemaDisciplinarLeve: 2,
    problemaDisciplinarMedio: 2,
    conflitoFamiliar: 1,
    separacaoPais: 1,
    problemaRelacionamentoMae: 1,
    problemaRelacionamentoPai: 1,
    problemaRelacionamentoCasa: 1,
    problemaSocioEconomico: 1
  };
  // insere o peso nas demandas
  var c = [];
  var v = [];
  for (demanda in caso) {
    c[demanda] = caso[demanda] * pesos[demanda];
    v[demanda] = (vizinho[demanda] != undefined) ? (vizinho[demanda] *
pesos[demanda]) : 0;
  }
  // calcula a soma dos quadrados das diferenças
  var soma = 0;
  for (demanda in c) {
    soma += Math.pow(c[demanda] - v[demanda], 2);
  }
  return Math.sqrt(soma / 26); // retorna a distância normalizada
}
```

Figura 4. Implementação do Cálculo de distância dos Casos na linguagem Javascript

Uma outra particularidade deste trabalho é o fato de que todos os casos deverão ser armazenados, já que deverão ser registradas todas as atividades de atendimento pedagógico atendidas pela ETEP. Para tanto, utilizaremos uma segunda coleção de registro de casos, esta apenas para fins estatísticos e administrativos, sendo que estes aspectos da aplicação não serão exploradas no presente trabalho.

### 3. DESENVOLVIMENTO DO PROJETO

#### 3.1. Arquitetura do Sistema

A arquitetura de um sistema pode ser definida como a descrição da organização dos diversos componentes do sistema e como estes, e o sistema, interagem entre si e com o ambiente no qual estão inseridos. A arquitetura selecionada para o desenvolvimento deste trabalho é o de cliente-servidor, comum ao ambiente web.



Figura 5. Modelo arquitetural de aplicação cliente-servidor.

Na arquitetura cliente-servidor, conforme descrito na Figura 5, o usuário através da aplicação cliente realiza uma requisição e envia os dados ao servidor, que processa e responde ao cliente com as informações (Mendes 2002). Os dados são persistidos na base de casos e podem ser processados pelo servidor.

Esse modelo arquitetural é especialmente entendido como um modelo em camadas, onde permite-se compreender o papel de cada camada do sistema de forma isolada.

O sistema proposto é compreendido em três camadas: 1) interface com usuários – é a camada que os usuários utilizam para interagir com o sistema; 2) camada de negócio – presente no lado servidor, corresponde a lógica de domínio da aplicação; 3) camada de persistência – nesta camada os dados da aplicação são guardados em um repositório ou banco de dados.

### 3.1.1. Interface com Usuários

A camada mais superficial da aplicação é a interface com o usuário. A interface deste trabalho foi desenvolvida usando um conjunto de documentos e formulários escritos utilizando HTML5 e algumas bibliotecas de script Javascript que podem ser interpretadas pelo navegador de internet.

Entre as bibliotecas utilizadas, destaca-se AngularJS, por ser uma biblioteca escrita em Javascript que tem seu foco na manipulação de interface através de uma tecnologia nomeada *two-way data binding*. Esta tecnologia permite que a interface seja alterada conforme a lógica do sistema muda, sem a necessidade de recarregar a aplicação para visualizar as novas informações, assim como, também reflete as ações do usuário na lógica do sistema em tempo de execução (Google Inc. 2010).

Ou seja, se relacionarmos os campos de um formulário a um objeto que pode ser manipulado com AngularJS, a cada interação que o usuário tenha no preenchimento deste, o objeto é modificado simultaneamente, permitindo, por exemplo, que os dados preenchidos possam ser verificados mesmo antes do usuário decidir encerrar a interação.

Com o uso desta tecnologia, o modelo passa a ser o recurso único da verdade (SSOT – *Single Source of Truth*, em inglês). SSOT refere-se a prática de estruturar a informação de maneira que exista apenas uma fonte de informação daquele dado, quaisquer outras formas de acesso aos dados se dá por referência apenas (Google Inc. 2010). A Figura 6 ilustra o funcionamento do *two-way data binding*.

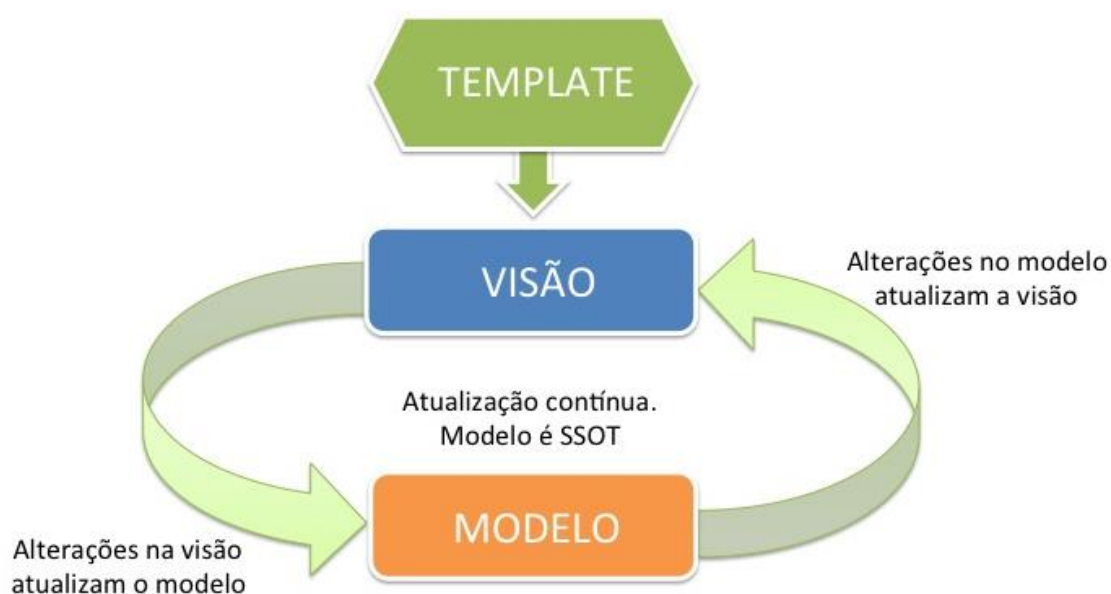


Figura 6. Esquema de *Two-way data binding* na biblioteca AngularJS

Adaptado de Google Inc, 2010.



### 3.1.2. Servidor da Aplicação

A segunda camada da aplicação fica alocada no servidor, responsável por receber as solicitações da interface com o usuário e responder de acordo com o resultado do processamento da requisição. Esta camada intermediária é executada, em um ambiente de servidor desenvolvido para este fim, através da tecnologia usada para processar Javascript no lado do servidor chamada NodeJS.

NodeJS é uma plataforma de processamento que utiliza a tecnologia direcionada a eventos com o modelo de I/O não-bloqueante, leve e eficiente. Geralmente utilizada em aplicações de tempo real, com intensa transmissão de dados e que funcionam como sistema distribuído. I/O é a sigla, em inglês, para o processo de entrada e saída de informações no servidor (Joyent Inc 2014).

Dizer que o modelo de entrada e saída é não-bloqueante significa que o servidor, quando receber uma requisição irá processá-la e em algum momento ele poderá responder aquela requisição, porém, para tanto, não deixará de receber e responder outras requisições que porventura cheguem até o mesmo.

Uma das características que permite o NodeJS ser não-bloqueante é que este é *single-thread*, ou seja, a aplicação terá somente uma instância de cada processo, sendo, no entanto, possível a criação de clusters, como no caso da biblioteca Mongoose, uma biblioteca escrita em Javascript para facilitar o trabalho do desenvolvedor que deseja conectar a aplicação com o SGBD MongoDB (LearnBoost 2010).

Outra particularidade é a forma como se dá orientação a eventos do NodeJS. Enquanto que nos navegadores de internet os eventos aos quais o Javascript estão relacionados serem, por exemplo, o clique do mouse, o pressionar de uma tecla etc., o servidor de aplicação NodeJS não responderá a estas ações. Os eventos a que estão relacionadas as atividades dele são, entre outros, *connect* (conexão com banco de dados), *open*, *read*, *close* (operações em arquivos em geral), *data* (como em transmissão de dados) etc.

A tecnologia de *callback* do Javascript tem um papel importante em todas as etapas de processamento do NodeJS, já que este é um item crucial para o gerenciador de eventos do servidor. É através dela que podemos, mesmo com a limitação de ser *single-thread*, realizar operações paralelas assíncronas de forma não-bloqueante (Joyent Inc 2014).

O mecanismo utilizado para o gerenciamento destes eventos é o *Event-Loop*. O *event-loop*, na verdade, é um ciclo infinito que percorre, a cada iteração, toda a sua lista de eventos e verifica se algum evento foi emitido. Quando identificado um evento, ele o executa e o encaminha para a fila de executados. O poder deste mecanismo de processo está na possibilidade de, enquanto um evento é processado, podermos definir funcionalidades que serão disparadas, e descrever a lógica destes. A Figura 7 exemplifica a execução do mecanismo de *Event-Loop* do NodeJS.



Figura 7. Mecanismo Event-Loop do NodeJS

Adaptado de Joyent Inc., 2014

### 3.1.3. Persistência dos Dados

A terceira e última camada, é a camada de persistência dos dados. Foi utilizado um SGBD (Sistema Gerenciador de Bancos de Dados) que nos permitisse armazenar os casos de forma otimizada e mantivesse a mesma estrutura de documentos utilizada nas camadas superiores da aplicação, fazendo assim, com que os dados sofressem o mínimo de alterações possíveis e que somente as transformações necessárias fossem realizadas.

Como um fator importante ao RBC, a base de casos ser persistida em um banco de dados que mantém sua estrutura na forma de coleção de documentos, e ainda permite a utilização nativa de algoritmos de classificação, MapReduce etc, nos auxiliando a obter resultados de forma nativa, tende a ser altamente eficiente.

Foi utilizado, para desenvolvimento do projeto objeto deste trabalho, um SGBD NoSQL, orientado a documentos, de código aberto e uso livre, de nome MongoDB (MongoDB Inc. 2014).

O termo noSQL é usado amplamente para descrever estruturas de bancos de dados que vão além da estrutura relacional, e que podem incluir conceitos de tuplas, grafos, pares de chave-valor, árvore, documento etc. Sadalage e Fowler (2013) afirmam que as características comuns dos bancos de dados NoSQL são:

- não utilizam modelo relacional;
- tem uma boa execução em clusters;
- seu código é aberto;
- são criados para propriedades na web do século XXI;
- não tem esquema (*schemeless*).

O MongoDB, utiliza também Javascript como linguagem nativa para suas rotinas e aplicações, além de armazenar os dados, em um formato semelhante à notação de objetos Javascript (JSON), só que em binário ao invés de texto pleno. A Figura 8 ilustra a comparação entre um documento representado em texto pleno e sua representação em binário utilizando a notação Javascript.

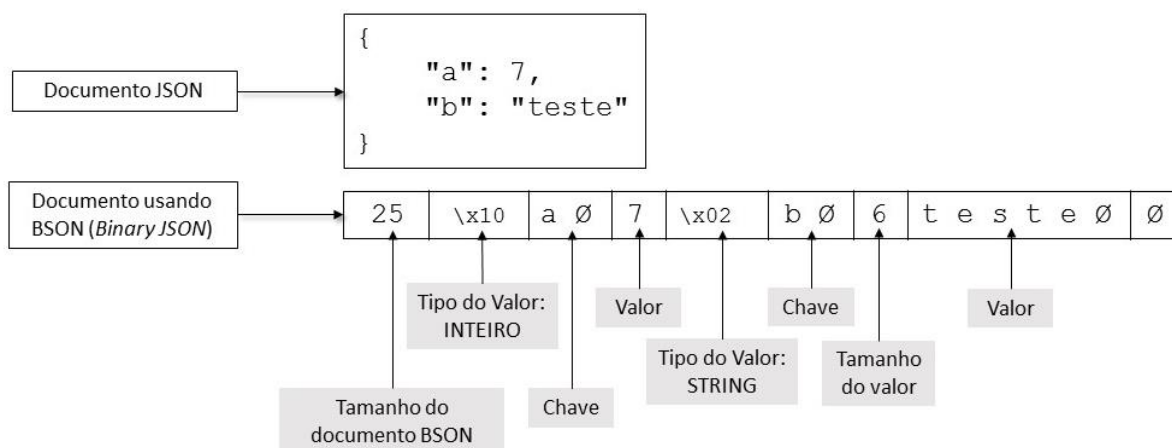


Figura 8. Comparação da representação de um mesmo documento usando JSON em texto pleno e na sua representação em binário, BSON.

Adaptado de Especificação BSON

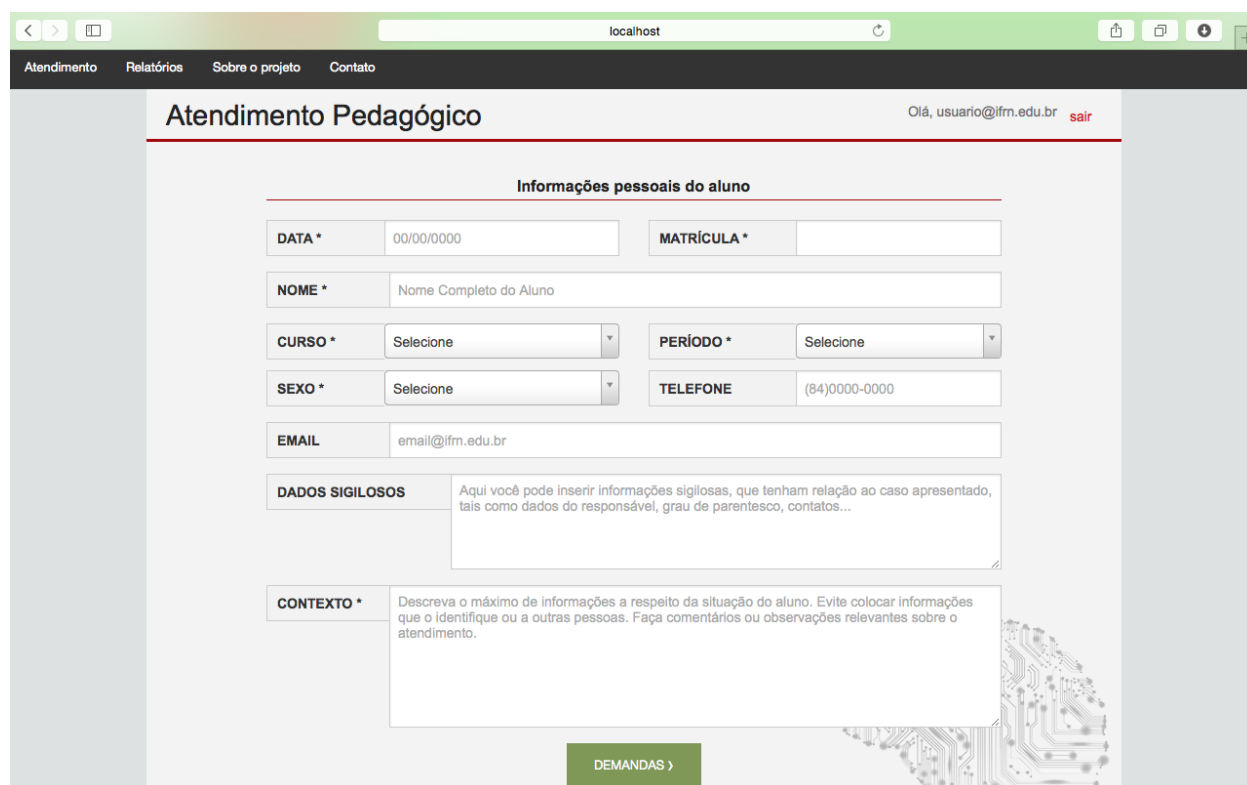
### 3.2. Resultados

O principal resultado, produto deste trabalho, foi o desenvolvimento de um sistema inteligente de apoio à decisão, que tem como público alvo de usuários, aconselheiros pedagógicos e técnicos de ETEP's. O sistema foi desenvolvido com o

conjunto de tecnologias descritas neste documento, e está disponibilizado como software de código aberto, sob a licença MIT. O Anexo II – LICENSE, é o texto integral da licença sob a qual está disponibilizado o software.

Como meio de disponibilização, e visando a possibilidade de futuras contribuições da comunidade, escolhemos criar um repositório público de código usando a plataforma web, na comunidade de desenvolvedores GitHub.com. O endereço URL de acesso público é <https://github.com/embits/educase>. Esta plataforma, além de permitir a publicação, possui uma ferramenta de versionamento de código, que se mostrou bastante útil durante o período de desenvolvimento.

A Figura 9 representa uma das telas do sistema, mais especificamente, o cadastro de informações de contextualização e discriminação do caso com informações sobre o aluno, curso, informações de contato etc.



The screenshot shows a web browser window with the URL 'localhost'. The page title is 'Atendimento Pedagógico'. The user is logged in as 'Olá, usuario@ifm.edu.br' with a 'sair' link. The main content area is a form titled 'Informações pessoais do aluno'. The form fields are:

DATA *	00/00/0000	MATRÍCULA *	
NOME *	Nome Completo do Aluno		
CURSO *	Selecione	PERÍODO *	Selecione
SEXO *	Selecione	TELEFONE	(84)0000-0000
EMAIL	email@ifm.edu.br		
DADOS SIGILOSOS	Aqui você pode inserir informações sigilosas, que tenham relação ao caso apresentado, tais como dados do responsável, grau de parentesco, contatos...		
CONTEXTO *	Descreva o máximo de informações a respeito da situação do aluno. Evite colocar informações que o identifique ou a outras pessoas. Faça comentários ou observações relevantes sobre o atendimento.		

At the bottom of the form, there is a green button labeled 'DEMANDAS >'.

Figura 9. Interface com o usuário do sistema. Passo 1 – Informações pessoais do aluno

Através de um formulário interativo, é realizado o cadastro do atendimento pedagógico em 3 passos, sendo que estes passos incluem as etapas de recuperação, reutilização, revisão e retenção do RBC a partir das informações fornecidas e da consulta na base de dados, conforme descrito a seguir.

No primeiro passo, são fornecidas as informações pessoais do aluno, como nome, matrícula, curso, sexo, data do atendimento, entre outras. Nesta tela também

deve ser fornecido uma descrição do contexto no qual o caso está inserido, descrevendo locais, pessoas e situações, que poderão servir inclusive em um momento posterior, quando este novo caso já fizer parte da base de casos e vier a ser sugerido como uma das prováveis soluções.

No segundo passo, o usuário deverá elencar as demandas, dentre a lista de 26 demandas elencadas pela equipe da ETEP e descrita na Tabela 2 supracitada. Após seleciona as demandas que estão relacionadas a este novo caso, o usuário deverá clicar no botão “Encaminhamentos”, de cor verde, ilustrado na Figura 10 abaixo, que realizará uma requisição assíncrona ao servidor informando as demandas que foram selecionadas. O servidor por sua vez calculará, dentre todos os casos registrados na Base de Casos, os vizinhos mais próximos com base no cálculo euclidiano de distância e irá responder a solicitação sugerindo as soluções dos casos mais similares. Este é o primeiro passo do processo de Raciocínio Baseado em Casos, a recuperação.

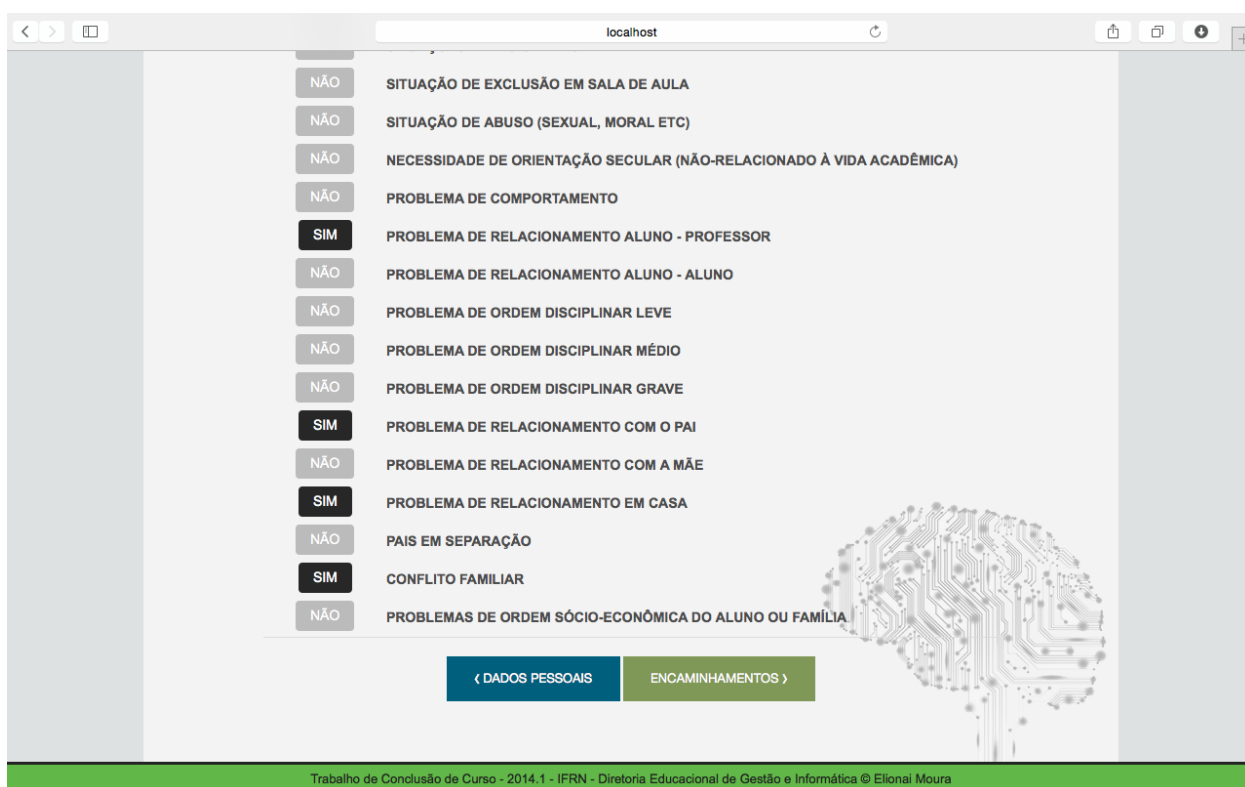


Figura 10. Passo 2 – Seleção de demandas do novo caso.

Quando o servidor responde à solicitação, o usuário poderá optar por aplicar a solução que melhor julgar conveniente à situação do novo caso, sendo esta a etapa de reutilização, ou ainda verificar uma ou mais soluções sugeridas e adaptar ao cenário do novo caso. A tela de sugestões de solução também permite que seja visualizado o

contexto no qual aquelas sugestões estão relacionadas. Esta é a etapa de revisão do RBC.

Por último, é feito o cadastro do caso novo, sendo enviada nova requisição ao servidor, de forma que este deverá calcular se o conjunto formado das demandas e dos encaminhamentos (soluções) apresentados forma uma nova unidade de conhecimento, a qual deverá ser inserida à base de casos, encerrando o ciclo RBC com a fase de retenção, ou aprendizado.

O processo de cadastro de um novo caso, no último passo, possui algumas particularidades que merecem uma maior atenção, inclusive, porque aqui são realizadas algumas etapas importantes que irão influenciar não somente nossa base de dados, mas também os registros gerais dos atendimentos pedagógicos e os problemas enfrentados pela ETEP do IFRN, conforme apontado pela pedagoga entrevistada.

Quando o servidor recebe a requisição para persistir o registro de um caso, a primeira ação que este deve tomar é a de classificar o novo caso. Como exposto no capítulo onde falamos sobre o algoritmo kNN, este é comumente utilizado para classificação, através da proposição de classes com base na classe dominante entre os vizinhos mais próximo ao caso dado.

O kNN foi utilizado neste trabalho para fins de indexação e para recuperação de casos na Base de Casos, através do uso do cálculo de distância e da seleção dos vizinhos mais próximos, porém na fase de classificação, utilizou-se o critério de classificar conforme a classe do atributo mais significativo presente no caso-problema, visto que, de acordo com a metodologia de enfrentamento de problemáticas que a equipe técnico-pedagógica utiliza, classificar os casos com base não em seus vizinhos mais próximos, mas sim com base na classe mais significativa entre as demandas que o compõe, facilitará a produção de dados estatísticos de forma a elencar mais evidentemente os problemas que necessitam de maior atenção.

Isto se dá pelo fato de que, muitas vezes, apesar de a classificação sugerida pelo kNN indicar determinado agrupamento, como a classificação foi inicialmente realizada em cima de cada uma das demandas especificamente, ocorre de serem sugeridas, às vezes, classificar o novo caso dentro de um grupo, ou classe, na qual este não possui nenhuma demanda relacionada.

Identificou-se que este fenômeno ocorre devido a classe dominante estar presente em casos, que possui uma variedade de demandas, que influencia aqueles

casos que, de outra forma, estariam agrupados de maneira distinta à sugerida pelo algoritmo do vizinho mais próximo.

Outro ponto importante de abordar, na persistência da informação, tem a ver com a produção de conhecimento *versus* a necessidade de cumular informações dos atendimentos pedagógicos. A Base de Casos deve ser constituída de informações únicas. Cada caso ali presente, deve representar um conjunto isolado de demandas e soluções, de maneira que, ao ser solicitada a persistência de um caso que não produza um exemplar único e realmente novo de conhecimento a ser agregado, este deverá ser descartado, de forma a manter a integridade e a performance da técnica de RBC. Por outro lado, a atividade da ETEP necessita que todos os atendimentos sejam registrados, de forma a criar um banco de informações que reflitam as situações do mundo real. A solução encontrada para que ambas condições sejam satisfeitas foi a utilização de duas coleções distintas no SGBD. Uma será nossa base de casos, onde as solicitações de recuperação, reutilização, revisão e retenção serão cumpridas e calculadas com base nos princípios da Inteligência Artificial, com unidades distintas de conhecimento, da forma descrita anteriormente.

A outra coleção, irá persistir todas as solicitações de retenção de casos, mesmos aqueles que não gerem uma nova unidade de conhecimento para o RBC, mas que porém, registra e representa o conjunto global de atendimentos realizados. Esta segunda coleção irá auxiliar, entre outras coisas, na produção de relatórios, como demonstrado na Figura 11, que ilustra o Relatório de Demandas mais Atendidas. Este tipo de relatório será um ferramental importante para a ETEP realizar atividades de prevenção e cuidados básico, tornando as informações colhidas durante os atendimentos “mais palpáveis e mais úteis”.



Figura 11. Relatório de Demandas mais Atendidas.

Por não ser, todavia, o cerne deste trabalho, a geração de relatórios e gráficos, mas sim, a construção de um sistema inteligente aplicado ao aconselhamento pedagógico, estas ferramentas deverão ser construídas em atividades posteriores, inclusive podendo ser explorada por outro trabalho, que tenham um enfoque mais estatístico ou voltado ao desenvolvimento deste tipo.



## 4. CONSIDERAÇÕES FINAIS

O desenvolvimento de Sistemas Inteligentes usando RBC tem demonstrado que, apesar do esforço na construção de uma Base de Casos suficientemente grande, e com a definição de algoritmos e técnicas de indexação e recuperação cada dia mais eficazes, um problema recorrente que ainda deve ser solucionado é o da aplicabilidade de soluções que se baseiam em quão similar são os casos entre si.

Isso se deve ao fator similaridade, referindo-se ao conhecimento, de forma a trazer um conjunto de experiências passadas, que possam ser utilizadas na solução do problema atual. Dado que, na maioria das vezes, a quantidade de casos é insuficiente ou inadequada para refletir a solução dos problemas, uma abordagem que tem uma grande chance de ser mais eficiente, ou que traga soluções mais adequadas, seria a utilização de algoritmos que proponham as soluções mais úteis, ao invés de sugerir soluções com base nos casos mais similares.

### 4.1. CONCLUSÃO

Aprendizado de máquina, inteligência artificial, modelos de raciocínio são temas extensos e que carecem ainda de bastante atenção por parte da comunidade acadêmica. São parte de um campo promissor de profundas transformações sociais, econômicas e culturais (Whitby 2004). Redes neurais, *big-data*, modelos de memória, entre outros, são cada dia mais utilizados em soluções comerciais e não somente no ramo científico (Nagy et al 2013).

É importante, porém, chamar a atenção para o fato que diversas considerações humanas nas decisões tomadas estão longe de serem simuladas por computador ou quaisquer sistemas digitais.

As soluções e análises de casos podem ser cada vez mais detalhados ou generalizados conforme seja a necessidade dos usuários de um sistema especialista, mas o fator humano ainda é o mais importante, por isso se diz que este é um sistema de apoio à decisão, e a necessidade de um especialista pedagogo, orientador acadêmico ou outro, será sempre necessário.

## 4.2. SUGESTÕES E RECOMENDAÇÕES

Existem diversas considerações a serem refletidas nesta matéria, caso se deseje aprofundar no estudo e desenvolvimentos de sistemas utilizando a técnica de Raciocínio Baseado em Casos para aprendizado de máquina, como por exemplo: o modelo de memória de casos, a granularidade da informação a ser apresentada, que casos devem ser representados, que estrutura pode ser mais facilmente adaptada a interface humano-computador, os casos de aquisição, representação, indexação, recuperação e os modelos de aprendizado concorrentes, entre outros, são temas que sugerimos serem explorados em trabalhos posteriores.

Entre as principais possibilidades de ampliação deste trabalho em desenvolvimentos futuros, apontamos como imprescindível a concentração de esforços em buscar um mecanismo de inferência que possibilite a recuperação de casos mais úteis a situação-problema apresentada, ou ainda, o desenvolvimento de um mecanismo de avaliação dos casos mais recorrentes, ou das soluções que são mais reutilizadas, e um estudo a respeito de como estão estruturados e elencando-os como sendo possíveis candidatos de soluções influenciadoras mesmo em casos onde a similaridade seja não tão significativa.

Possivelmente, realizar uma implantação em outras diretorias acadêmicas, ou mesmo em outros campi e instituições educacionais, permitindo uma avaliação sistematizada da solução apresentada neste trabalho e ampliando o número de casos da Base de Casos. Existe ainda a necessidade de integração desta com outras ferramentas, como por exemplo, um sistema de mineração de dados educacionais, que forneça informações em maior número, gerando casos simulados. Geração de uma plataforma que exiba relatórios de utilização, e gere os gráficos sugeridos na conclusão deste trabalho também serão ferramentas úteis e são sugeridas como desdobramentos do presente trabalho.

## REFERÊNCIAS

ABEL, Mara; CASTILHO, José Mauro Volkmer. **Um Estudo Sobre Raciocínio Baseado em Casos** – 1996. UFRS – Porto Alegre, RS.

Crockford, D. **RFC 4627** – 2006. The Internet Society – Acessado em: 23/9/2014. Disponível em: <http://www.ietf.org/rfc/rfc4627.txt>.

FERREIRA, Silvio. **Guia Prático de HTML5** – 2013. Editora Universo dos Livros – São Paulo, SP.

FLANAGAN, David. **Javascript: O guia definitivo** – Tradução de Edson Furmankiewicz – 2004. Editora Bookman – Porto Alegre, PR.

JEFFREY, Dean; GUEMAWAT, Sanjay. **MapReduce: Simplified Data Processing on Large Clusters** – 2004 – Google Inc. Acessado em 14/9/2014. Disponível em: <http://static.googleusercontent.com/media/research.google.com/pt-BR/archive/mapreduce-osdi04.pdf>.

MENDES, Antônio. **Arquitetura de Software: desenvolvimento orientado para arquitetura** 2002 – Editora Campus. Rio de Janeiro – RJ.

NAGY, Heba Mohammed; ALY, Walid Mohamed; HEGAZY, Osama Fathy. **An Educational Data Mining System for Advising Higher Education Students** – 2013. International Journal of Computer, Information Science and Engineering, Vol:7, N:10, 2013. World Academy of Science, Engineering and Technology.

SADALAGE, Pramod J; FOWLER, Martin. **NoSQL Essencial, Um guia conciso para o mundo emergente da persistência poliglota** – 2013. Novatec Editora – São Paulo, SP.

TIWARY, Shashank. **Professional NoSQL, a hands-on guide to leveraging NoSQL databases** – 2011. John Wiley & Sons.

WANGENHEIM, Christiane Gresse von; WANGENHEIM, Aldo von e RATEKE, Thiago. **Raciocínio Baseado em Casos com Softwares Livres e Aplicativos Móveis** – 2ª edição - 2013. Bookess Editora – Florianópolis, SC.

WATSON, Ian Duncan. **Applying Case-Based Reasoning** – 1997. Morgan Kauffman Publishers – San Francisco, CA.

WHITBY, Blay. **Inteligência Artificial – Um guia para iniciantes** - 2004. Editora Madras – São Paulo, SP.

\_\_\_\_\_. **AngularJS** – 2010. Google Inc. – Acessado em: 14/8/2014. Disponível em: <http://docs.angularjs.org>

\_\_\_\_\_. **BSON Specs** – 2014. Creative Commons (sem direitos autorais). – Acessado em: 6/8/2014. Disponível em: <http://bsonspec.org/spec.html>

\_\_\_\_\_. **MongoDB** – 2014. MongoDB Inc. – Acessado em: 14/8/2014. Disponível em: <http://www.mongodb.com/what-is-mongodb>

\_\_\_\_\_. **Mongoose** – 2010. LearnBoost. – Acessado em: 14/8/2014. Disponível em: <http://mongoosejs.com/docs/guide.html>

\_\_\_\_\_. **NodeJS** – 2014. Joyent Inc. – Acessado em: 10/6/2014. Disponível em: <http://nodejs.org/documentation>

\_\_\_\_\_. **Standard ECMA-262** – 2011. ECMA International – Acessado em: 23/9/2014. Disponível em: <http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf>

## GLOSSÁRIO

**Algoritmo:** é um conjunto de instruções, ou regras, para a resolução de um problema. Pode ser descrito como um processo computacional, finito, baseado em um conjunto de regras, que executa uma tarefa específica.

**Callback:** Tecnologia implementada por algumas linguagens de programação, que possibilita a passagem de uma referência a um código executável, como parâmetro de outro código executável. Também conhecida como função de retorno.

**Camada de Negócios:** diz-se do conjunto das partes de um sistema, onde estão contidas todas as regras necessárias ao funcionamento das atividades principais do sistema, podendo incluir descrição dos modelos de dados, níveis de acesso, controladores de entrada e saída etc.

**Clustering:** Termo em inglês, significando o conjunto de técnicas de agrupamento e classificação das informações.

**ETEP:** Equipe Técnico-Pedagógica. Uma equipe composta por dois tipos de profissionais: Técnicos Administrativo de Nível Superior Técnico em Assuntos Educacionais e Técnicos Administrativo de Nível Superior Pedagogo. Suas atribuições são descritas em um documento institucional.

**Evento:** é toda mudança de estado, que pode desencadear uma ou mais atividades, ou representar a interação do usuário, ou algum dispositivo externo com o sistema.

**Licença MIT:** Declaração de licenciamento de software que garante acesso livre ao código, permitindo derivação e alteração do mesmo, enquanto garante o direitos do autor ou autores do código fonte.

**No-SQL:** Tecnologia de bancos de dados que utilizam esquemas diversos do tradicional modelo relacional.

**Objeto:** é um modelo de dados, que visa representar uma entidade do mundo real, abstraindo as características irrelevantes ao sistema, e mantendo as funcionalidades e papéis mais significativos.

**Persistência:** Diz-se da atividade ou dos mecanismos utilizados para realizar a durabilidade da informação através do seu armazenamento, possibilitando o acesso posteriormente.

Schemeless: Termo em inglês, que define um Sistema Gerenciador de Banco de Dados que é independente de esquemas, ou seja, a estrutura de dados numa mesma coleção é livre de padrões, podendo inclusive, coexistirem dados completamente distintos em estrutura na mesma coleção.

Script: conjunto de instruções bem-formadas, escrito em linguagem de programação, que pode ser interpretado ou compilado por um dispositivo ou sistema que entenda as instruções expressas.

# ANEXO I – FICHA DE ATENDIMENTO PEDAGÓGICO



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE

## FICHA DE ATENDIMENTO PEDAGÓGICO

DIATINF

FOTO

Aluno (a): \_\_\_\_\_

Matrícula: \_\_\_\_\_ Curso: \_\_\_\_\_

Fone: \_\_\_\_\_ E-mail: \_\_\_\_\_

Data: \_\_\_/\_\_\_/\_\_\_

RESUMO DO ATENDIMENTO

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

\_\_\_\_\_  
Assinatura do Aluno ou Responsável

\_\_\_\_\_  
Assinatura da Pedagogia

Data: \_\_\_/\_\_\_/\_\_\_

RESUMO DO ATENDIMENTO

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

\_\_\_\_\_  
Assinatura do Aluno ou Responsável

\_\_\_\_\_  
Assinatura da Pedagogia

## **ANEXO II - LICENSE**

The MIT License (MIT)

Copyright (c) 2014 Elionai Moura

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.