

5 Experiências de Uso da Infra-Estrutura

5.1. Objetivo

Este capítulo apresenta algumas experiências de uso da infra-estrutura proposta, onde serão apresentados exemplos de *matchmaking* aplicado em dois contextos. O primeiro deles está relacionado ao domínio de gerenciamento de competências, em que se deseja propor cursos para pessoas, e o segundo está relacionado ao *matching* entre perfis de pessoas.

Também será apresentada uma discussão sobre a capacidade de se estender a infra-estrutura. Nesse sentido, será apresentada uma proposta de como definir o conceito de prioridade nos resultados de *matching* obtidos através da infra-estrutura.

Alguns dos exemplos de *matching* para o domínio de gerenciamento de competências apresentados nesse capítulo utilizam as funções de avaliação de similaridade discutidas e implementadas no capítulo anterior, bem como todo o conjunto restante de implementações realizadas para o domínio em questão. Para os demais exemplos a serem apresentados, serão definidas novas implementações, que serão discutidas de forma não tão detalhada quanto aquelas do Capítulo 4.

5.2. Matchmaking Aplicado ao Domínio de Gerência de Competências

5.2.1. Estrutura dos Conceitos do Domínio

O domínio de gerência de competências pode ser definido, de forma bem resumida, por três conceitos básicos: pessoas, recursos de desenvolvimento e competências. Os recursos de desenvolvimento podem ser cursos, livros, palestras, dentre outros. Porém, para o caso em questão, eles ficarão restritos apenas a cursos.

Basicamente, pessoas possuem competências que foram desenvolvidas por cursos que foram feitos durante suas vidas. Porém, para desempenhar determinadas tarefas, as pessoas, normalmente, precisam desenvolver novas competências, tendo associadas a elas um conjunto de competências requeridas.

Os cursos, em contrapartida, representam os instrumentos utilizados para desenvolver nas pessoas as competências que elas requerem. Porém, para que uma pessoa possa fazer um curso, faz-se necessário que ela atenda aos seus pré-requisitos, que podem ser, por exemplo, outros cursos que devem ser feitos anteriormente ao curso em questão.

A Figura 41 apresenta uma visão desses conceitos e de seus atributos, bem como os relacionamentos existentes entre eles.

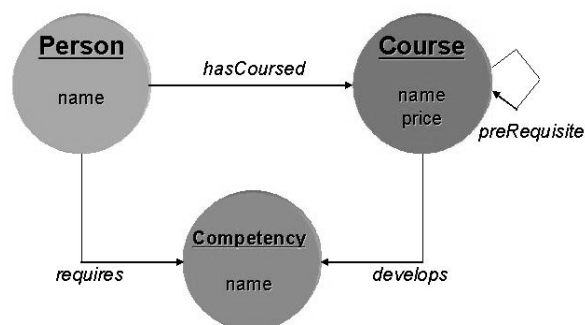


Figura 41 – Estrutura dos conceitos do domínio de gerência de competências

Essa estrutura, para o domínio de gerência de competências, será acessada pela infra-estrutura através da definição de um conjunto de ontologias descritas em RDFS, que podem ser acessadas através das URI's apresentadas na Tabela 1.

Conceito	URI correspondente ao RDFS associado
Person	http://www.lac.inf.puc-rio.br/~ferrao/ontologies/onts-rdfs/person.rdfs
Course	http://www.lac.inf.puc-rio.br/~ferrao/ontologies/onts-rdfs/course.rdfs
Competency	http://www.lac.inf.puc-rio.br/~ferrao/ontologies/onts-rdfs/aulanet.rdfs

Tabela 1 – URI's dos conceitos do domínio de gerência de competências

5.2.2. Definindo Instâncias do Domínio

Para a execução dos exemplos relacionados ao domínio em questão, foi necessário definir um conjunto de instâncias dos conceitos definidos anteriormente (pessoas, cursos e competências). Primeiramente, foi estabelecido um conjunto de instâncias relacionadas a cursos, que estão apresentadas na Tabela 2.

URI relacionada ao RDF associado à instância			
Nome do Curso <i>name property</i>	Preço <i>price property</i>	Comps. desenvolvidas <i>develops property</i>	Pré-requisitos <i>preRequisite property</i>
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course1.rdf			
Course1	\$100,00	Competency1	
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course2.rdf			
Course2	\$150,00	Competency2 Competency4	Course1
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course3.rdf			
Course3	\$150,00	Competency2 Competency3	Course1
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course4.rdf			
Course4	\$300,00	Competency5 Competency6 Competency8 Competency12	Course2 Course5
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course5.rdf			
Course5	\$250,00	Competency5 Competency6 Competency7	Course7
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course6.rdf			
Course6	\$150,00	Competency4 Competency5 Competency6	Course3
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course7.rdf			
Course7	\$400,00	Competency9 Competency10 Competency13	Course4
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course8.rdf			
Course8	\$300,00	Competency8 Competency9 Competency11	Course5
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course9.rdf			
Course9	\$300,00	Competency9 Competency11	Course6
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/courses/course10.rdf			
Course10	\$300,00	Competency9 Competency10 Competency11 Competency12	Course6

Tabela 2 – Instâncias de cursos

Cada curso tem associado a ele um conjunto de competências que ele desenvolve, bem como seu nome e preço. Para facilitar o entendimento da relação de pré-requisitos entre os cursos, a Figura 42 apresenta uma visão simplificada dos valores definidos para essa relação.

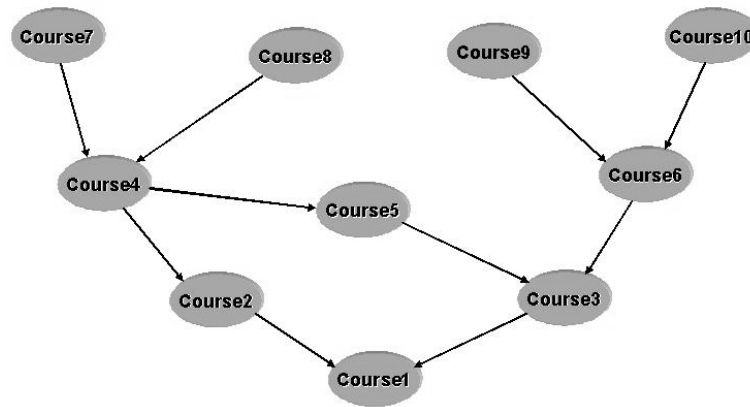


Figura 42 – Relacionamento de pré-requisitos entre as instâncias de cursos

As setas indicam os pré-requisitos de cada curso. O *Course1*, por exemplo, não tem nenhum pré-requisito definido, podendo ser cursado por qualquer pessoa. Em contrapartida, o *Course4* só pode ser cursado por pessoas que já tenham feito os cursos *Course2* e *Course5*, bem como todos os pré-requisitos associados a eles.

Também foi definido um conjunto de instâncias relacionadas ao conceito *Person*. Para cada pessoa foi definido um conjunto de cursos realizados anteriormente, bem como seu nome e o conjunto de competências que ela precisa desenvolver. A Tabela 3 apresenta essas instâncias, bem como os valores definidos, relacionadas a cada uma das pessoas.

URI relacionada ao RDF associado à instância		
Nome da Pessoa <i>name property</i>	Cursos realizados <i>hasCoursed property</i>	Competências requeridas <i>requires property</i>
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/people/luis.rdf		
Luis	Course1	Competency2 Competency5 Competency9 Competency11
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/people/maria.rdf		
Maria	Course1	Competency2 Competency5 Competency8 Competency9 Competency11
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/people/antonio.rdf		
Antônio	Course1 Course2 Course3	Competency5 Competency6 Competency8 Competency9
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/people/tereza.rdf		
Tereza	Course1 Course2 Course3	Competency5 Competency6 Competency8 Competency9 Competency13
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/people/joao.rdf		
João	Course1 Course2	Competency12
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/people/ana.rdf		
Ana	Course1 Course2 Course3 Course5	Competency12
http://www.lac.inf.puc-rio.br/~ferrao/ontologies/rdf/people/cristiano.rdf		
Cristiano		Competency1 Competency2 Competency4 Competency5 Competency6 Competency12

Tabela 3 – Instâncias de pessoas

Para os exemplos que serão apresentados posteriormente, foram definidas 13 competências. Devido à simplicidade de sua estrutura, definida apenas pelo seu nome, elas não serão apresentadas aqui, mas estão disponíveis para apreciação no Anexo A, assim como todas as ontologias dos conceitos apresentados, e suas respectivas instâncias.

5.2.3. Exemplo de Matching do Tipo 1to1

A Figura 43 apresenta uma requisição de *matching* do tipo *1to1* (para maior clareza de conteúdo, as referências completas das URI's das instâncias

foram omitidas). Nesse exemplo, dado um conjunto de pessoas e de cursos representando, respectivamente, o conjunto de demandas e de ofertas, deseje-se definir, para cada pessoa, qual é o melhor curso que deve ser feito por ela, de modo a atender ao seu conjunto de competências requeridas.

```
<?xml version="1.0"?>
<Matchings>
  <Matching type="1to1" model="ONTOLOGY" name="Example_1to1">
    <Problem>
      <Elements>
        <Element id="luis.rdf" />
        <Element id="antonio.rdf" />
        <Element id="maria.rdf" />
        <Element id="joao.rdf" />
        <Element id="tereza.rdf" />
        <Element id="ana.rdf" />
        <Element id="cristiano.rdf" />
      </Elements>
    </Problem>
    <SolutionSpace>
      <Elements>
        <Element id="course1.rdf" />
        <Element id="course2.rdf" />
        <Element id="course3.rdf" />
        <Element id="course4.rdf" />
        <Element id="course5.rdf" />
        <Element id="course6.rdf" />
        <Element id="course7.rdf" />
        <Element id="course8.rdf" />
        <Element id="course9.rdf" />
        <Element id="course10.rdf" />
      </Elements>
    </SolutionSpace>
    <Strategies>
      <BruteForceStrategy name="Brute Force Strategy" maximize="true">
        <DomainKnowledge name="PersonCourseDomainKnowledge" />
      </BruteForceStrategy>
    </Strategies>
  </Matching>
</Matchings>
```

Figura 43 – Requisição de matching 1to1 para o domínio de gerência de competências

A Tabela 4 sumariza os melhores resultados obtidos para cada uma das pessoas especificadas na requisição de *matching*. O valor de similaridade, nesse exemplo, corresponde ao número de competências requeridas pela pessoa que foram atendidas pelo curso proposto na solução do problema de *matchmaking*.

Pessoa	Melhor Solução	Valor Similaridade Melhor Solução	2ª Melhor Solução	3ª Melhor Solução
Luis	Course3	1.0	Course2	-
Antônio	Course6	2.0	Course5	-
Maria	Course3	1.0	Course2	-
João	Course3	0.0	-	-
Tereza	Course6	2.0	Course5	-
Ana	Course4	1.0	-	-
Cristiano	Course1	1.0	-	-

Tabela 4 – Resultados de matching 1to1 para o domínio de gerência de competências

Deve-se perceber que, nem sempre, o curso obtido como solução atende a alguma das competências requeridas pela pessoa. Isso acontece uma vez que o curso que atenderia a alguma dessas competências pode apresentar pré-requisitos ainda não atendidos pela pessoa. Dessa forma, o curso proposto como melhor solução pode equivaler a um desses pré-requisitos, que deve ser cumprido em algum momento pela pessoa.

5.2.4. Exemplo de Matching do Tipo 1toN

A Figura 44 apresenta apenas a parte alterada, com relação à requisição da Figura 43, para o *matching 1toN*, em que se deseja propor seqüências de cursos para cada uma das pessoas definidas no conjunto de demandas do problema de *matchmaking*. A construção dessas seqüências deve levar em consideração os pré-requisitos de cada um dos cursos que a compõem, bem como a não ocorrência de dois ou mais cursos que desenvolvam as mesmas competências na pessoa.

```
<?xml version="1.0"?>
<Matchings>
  <Matching type="1toN" model="ONTOLOGY" name="Example_1toN">
    ...
    <Strategies>
      <TabuSearchStrategy name="Tabu Search Strategy" maximize="true">
        <DomainKnowledge name="PersonCourseDomainKnowledge" />
      </TabuSearchStrategy>
    </Strategies>
  </Matching>
</Matchings>
```

Figura 44 – Requisição de matching 1toN para o domínio de gerência de competências

Como pode ser visto na requisição de *matching 1toN*, as principais mudanças com relação ao exemplo anterior dizem respeito ao tipo de *matching* a ser executado (*type="1toN"*) e à estratégia de *matching* a ser utilizada para obter os resultados, baseada no algoritmo de busca *Tabu*.

A Tabela 5 apresenta os melhores resultados obtidos a partir da execução da requisição de *matching* desse exemplo. Os valores de similaridades obtidos representam, ordenadamente, o número de competências atendidas pela seqüência de cursos, o custo total para fazer todos os cursos da seqüência e o número de competências atendidas pelo primeiro curso a ser realizado pela pessoa.

Pessoa	Melhor Solução	Valor Similaridade Melhor Solução	2ª Melhor Solução	3ª Melhor Solução
Luis	Course3 Course6 Course9	4.0 -600.0 1.0	Course2 Course3 Course6 Course9	Course2 Course3 Course6
Antônio	Course5 Course8	4.0 -550.0 2.0	Course5	-
Maria	Course3 Course5 Course8	5.0 -700.0 1.0	Course3 Course6 Course5 Course8	Course2 Course6 Course9 Course5 Course8
João	Course3 Course6 Course10	0.799 -600.0 -0.1	-	-
Tereza	Course5 Course4 Course7	5.0 -950.0 2.0	Course5 Course6 Course4 Course7	Course5 Course8 Course6 Course4 Course7
Ana	Course4	1.0 -300.0 1.0	-	-
Cristiano	Course1 Course3 Course6 Course10	6.0 -700.0 1.0	Course1 Course2 Course3 Course6 Course10	Course1 Course2 Course3 Course6

Tabela 5 – Resultados de *matching 1toN* para o domínio de gerência de competências

5.2.5. Exemplo de Matching do Tipo NtoN

Durante o Capítulo 4 não foi apresentada nenhuma implementação de função de avaliação de similaridade para o caso *NtoN*. O Anexo B apresenta uma possível implementação das funções *evalNto1* e *evalNtoN* para o domínio de gerenciamento de competências.

Como pode ser visto na implementação dessas funções, o valor de similaridade retornado pela função *evalNto1* corresponde à soma dos valores de similaridade obtidos pela aplicação da função *eval1to1* entre cada uma das pessoas que compõem o conjunto de demandas e um curso em particular do conjunto de ofertas.

A função *evalNto1* teve sua implementação definida pois ela constitui a base para a implementação da função *evalNtoN*. O valor de similaridade obtido por essa última função corresponde à aplicação da função *evalNto1* entre as pessoas que compõem o conjunto de demandas e cada um dos cursos que compõem a seqüência de cursos sendo proposta como solução para o problema de *matchmaking*.

A Figura 45 apresenta um exemplo de requisição de *matching* para o caso *NtoN*, em que se deseja propor uma seqüência de cursos para um conjunto de pessoas. Pode-se verificar que apenas duas pessoas compõem o conjunto de demandas. Isso se explica pela maneira como as funções de similaridade discutidas anteriormente foram implementadas. Como estamos obtendo um curso ou uma seqüência de cursos para um conjunto inteiro de pessoas, haverá situações em que não será possível obter uma solução. Para o exemplo em questão, buscou-se definir um caso em que seria possível obter soluções, uma vez que as duas pessoas que compõem o conjunto de demandas necessitam atender algumas competências em comum.

```

<?xml version="1.0"?>
<Matchings>
  <Matching type="NtoN" model="ONTOLOGY" name="Example_NtoN">
    <Problem>
      <Elements>
        <Element id="antonio.rdf" />
        <Element id="tereza.rdf" />
      </Elements>
    </Problem>
    <SolutionSpace>
      <Elements>
        <Element id="course1.rdf" />
        <Element id="course2.rdf" />
        <Element id="course3.rdf" />
        <Element id="course4.rdf" />
        <Element id="course5.rdf" />
        <Element id="course6.rdf" />
        <Element id="course7.rdf" />
        <Element id="course8.rdf" />
        <Element id="course9.rdf" />
        <Element id="course10.rdf" />
      </Elements>
    </SolutionSpace>
    <Strategies>
      <TabuSearchStrategy name="Tabu Search Strategy" maximize="true">
        <DomainKnowledge name="PersonCourseDomainKnowledge" />
      </TabuSearchStrategy>
    </Strategies>
  </Matching>
</Matchings>

```

Figura 45 – Requisição de matching NtoN para o domínio de gerência de competências

A Tabela 6 resume os resultados obtidos para a requisição de *matching* anterior. O valor de similaridade obtido diz respeito, ordenadamente, à soma do número de competências atendidas para todas as pessoas do conjunto, ao custo a ser pago por cada uma das pessoas para fazer os cursos e ao número de competências atendidas pelo primeiro curso que compõem a solução de *matchmaking*.

Pessoas	Melhor Solução	Valor Similaridade Melhor Solução	2ª Melhor Solução	3ª Melhor Solução
Antônio	Course5	8.0	Course5	-
Tereza	Course8	-550.0 4.0		

Tabela 6 – Resultados de matching NtoN para o domínio de gerência de competências

5.3. Matching Entre Perfis de Pessoas

5.3.1. Visão Geral

Dado um grupo de pessoas, o problema de selecionar aquela, ou aquelas, que estão mais relacionadas a uma pessoa qualquer perfaz um exemplo bastante recorrente de problema de *matchmaking*. Contudo, essa seleção pode ser feita levando em consideração diferentes aspectos. Por exemplo, se o objetivo da escolha for encontrar o melhor parceiro para acompanhar outra pessoa num evento qualquer, parâmetros relacionados às preferências e às afinidades pessoais desses indivíduos devem ter muita relevância. Porém, se o objetivo final for realizar alguma tarefa acadêmica ou profissional, os parâmetros a serem avaliados muito provavelmente serão outros.

Os exemplos apresentados a seguir não têm por objetivo apresentar critérios de seleção complexos, tais como os apresentados em [Cali, 2004] e [Machado, 2004]. Em contrapartida, o objetivo é demonstrar a aplicabilidade da infra-estrutura num segundo contexto. Esses exemplos definem como executar o *matching* entre perfis de pessoas utilizando a estrutura e as instâncias definidas para o conceito *Person*.

Para tanto, foram construídas novas implementações das classes Domain Knowledge e Evaluator, sendo definidas duas funções de similaridades, para os casos de *matching 1to1* e *1toN*. Para essas funções, o critério de similaridade corresponde ao quanto as pessoas precisam aprender, ou seja, o grau de similaridade entre duas pessoas é maior proporcionalmente ao número de competências requeridas que elas possuem em comum. Essas implementações correspondem às classes *PersonPersonDomainKnowledge* e *PersonPersonEvaluator*, estando seus códigos-fonte apresentados no Anexo C.

5.3.2. Exemplo de Matching do Tipo 1to1

A Figura 46 apresenta um exemplo de requisição para o caso de *matching 1to1*. Nesse exemplo, dados dois conjuntos de pessoas, para cada pessoa do

conjunto de demandas, deseja-se definir qual a pessoa mais similar a ela no conjunto de ofertas.

```
<?xml version="1.0"?>
<Matchings>
  <Matching type="1to1" model="ONTOLOGY" name="PersonPerson_1to1">
    <Problem>
      <Elements>
        <Element id="luis.rdf" />
        <Element id="antonio.rdf" />
        <Element id="maria.rdf" />
        <Element id="joao.rdf" />
        <Element id="tereza.rdf" />
        <Element id="ana.rdf" />
        <Element id="cristiano.rdf" />
      </Elements>
    </Problem>
    <SolutionSpace>
      <Elements>
        <Element id="luis.rdf" />
        <Element id="antonio.rdf" />
        <Element id="maria.rdf" />
        <Element id="joao.rdf" />
        <Element id="tereza.rdf" />
        <Element id="ana.rdf" />
        <Element id="cristiano.rdf" />
      </Elements>
    </SolutionSpace>
    <Strategies>
      <BruteForceStrategy name="FindPartner" maximize="true">
        <DomainKnowledge name="PersonPersonDomainKnowledge" />
      </BruteForceStrategy>
    </Strategies>
  </Matching>
</Matchings>
```

Figura 46 – Requisição de matching entre perfis de pessoas para o caso 1to1

A Tabela 7 sumariza os resultados obtidos para esse caso de *matching*. Novamente, o valor de similaridade corresponde ao número de competências requeridas que a demanda e a oferta possuem em comum.

Pessoa	Melhor Solução	Valor Similaridade Melhor Solução	2ª Melhor Solução	3ª Melhor Solução
Luis	Maria	4.0	Cristiano	Tereza
Antônio	Tereza	4.0	Maria	Cristiano
Maria	Luis	4.0	Tereza	Antonio
João	Cristiano	1.0	Ana	-
Tereza	Antônio	4.0	Maria	Cristiano
Ana	Cristiano	1.0	João	-
Cristiano	Tereza	2.0	Maria	Antônio

Tabela 7 – Resultados do matching 1to1 entre perfis de pessoas

5.3.3. Exemplo de Matching do Tipo 1toN

A Figura 47 apresenta apenas a parte alterada, com relação à requisição da Figura 46, para o caso de *matching 1toN*, cujo objetivo é obter o grupo de pessoas mais similar a uma pessoa em particular.

```
<?xml version="1.0"?>
<Matchings>
  <Matching type="1toN" model="ONTOLOGY" name="PersonPerson_1toN">
    ...
    <Strategies>
      <TabuSearchStrategy name="FindPartners" maximize="true">
        <DomainKnowledge name="PersonPersonDomainKnowledge" />
      </TabuSearchStrategy>
    </Strategies>
  </Matching>
</Matchings>
```

Figura 47 – Requisição de matching entre perfis de pessoas para o caso 1toN

A Tabela 8 apresenta os resultados obtidos com a execução do *matching* definido na requisição anterior. O valor de similaridade obtido corresponde à soma de todas as competências requeridas entre a demanda e cada uma das pessoas que compõem o conjunto correspondente à oferta.

Pessoa	Melhor Solução	Valor Similaridade Melhor Solução	2ª Melhor Solução	3ª Melhor Solução
Luis	Maria Antônio Tereza Cristiano	10.0	-	-
Antônio	Tereza Maria Luis Cristiano	11.0	-	-
Maria	Luis Antônio Tereza Cristiano	12.0	-	-
João	Ana Cristiano	2.0	-	-
Tereza	Antônio Maria Luis Cristiano	11.0	-	-
Ana	João Cristiano	2.0	-	-
Cristiano	Luis Antônio Maria Tereza João Ana	10.0	-	-

Tabela 8 – Resultados do matching 1toN entre perfis de pessoas

5.4. Estendendo a Infra-Estrutura Proposta

5.4.1. Visão Geral

O quanto a infra-estrutura proposta é flexível? Será necessário definir novos pontos de extensão para acomodar outros aspectos relacionados a certos problemas de *matchmaking*? Essas e outras questões podem ser alvo de questionamento por parte de pessoas que necessitem utilizar a infra-estrutura proposta para resolver um problema de *matchmaking* em particular.

Devido à riqueza de complexidade que, normalmente, cerca a classe de problemas relacionada a *matchmaking*, pensar que todos os aspectos relacionados a esses problemas foram levados em consideração na definição e na construção da infra-estrutura é ser bastante otimista. Porém, nem sempre será necessário criar novos pontos de flexibilização para acomodar determinados aspectos de um dado problema a ser resolvido, podendo isso ser feito através da adaptação dos pontos de flexibilização existentes.

O objetivo dessa seção é apresentar uma discussão acerca dessa capacidade de adaptação da infra-estrutura, sendo definida para tanto uma proposta de inclusão do conceito de prioridade nos resultados de *matchmaking*, até então não contemplado pelos pontos de flexibilização existentes. Como exemplo do se deseja alcançar, suponha que o objetivo da execução do *matching* seja propor cursos para pessoas. Porém, deseja-se dar prioridade para escolhas de cursos que ensinem matérias relacionadas à Engenharia de Software. Dessa forma, se dois resultados de *matching* possuírem o mesmo grau de similaridade, porém um deles ensinar Engenharia de Software, esse será mais bem classificado que o outro curso em questão.

É claro que é possível implementar uma função de similaridade que possua esse comportamento. Porém, parece mais claro separar esse conceito de prioridade em algo que possa ser reaproveitado para outros problemas de *matchmaking*. A Figura 48 apresenta uma proposta de implementação do conceito de prioridade na definição de funções de avaliação de similaridade.

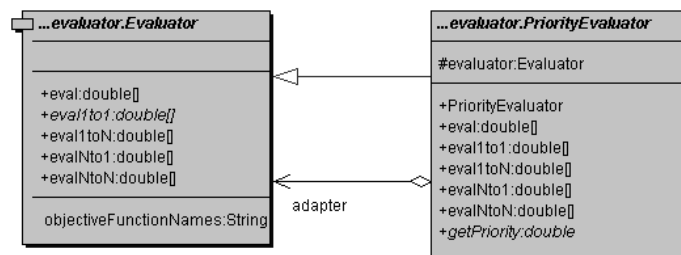


Figura 48 – Diagrama de classes

Como pode ser visto, o componente Evaluator foi estendido, sendo definido um novo componente chamado PriorityEvaluator. Ele funciona como um adaptador (*Adapter pattern*) para outras implementações de Evaluator, definindo sobre os valores das funções de avaliação o conceito de prioridade.

Basicamente, esse novo componente disponibiliza um método abstrato *getPriority* que deve ser implementado com a lógica de negócio relacionada ao conceito de prioridade. Para o exemplo discutido anteriormente, uma implementação desse método poderia verificar se algum dos cursos propostos na solução de *matching* ensina Engenharia de Software, atribuindo nesse caso um valor adicional a essa solução em particular.

O Anexo D apresenta a implementação do componente PriorityEvaluator. Como pode ser visto, o método *eval* é redefinido, sendo seu comportamento definido da seguinte forma: num primeiro passo, o método *eval* do componente adaptado é chamado, sendo o valor de similaridade obtido. O passo seguinte é chamar o método *getPriority*, que avaliará a solução de *matching* de acordo com a lógica de negócio implementada, sendo obtido um novo valor de similaridade, que contempla o conceito de prioridade. Por fim, os dois valores de similaridade obtidos serão unidos, sendo que aquele relacionado à prioridade ocupará a primeira dimensão do valor de similaridade resultante.

5.4.2. Definindo Prioridade no Domínio de Gerência de Competências

Nessa seção será apresentado um exemplo do uso do componente PriorityEvaluator para definir prioridade nos resultados de *matching* obtidos para

o domínio de gerenciamento de competências. Para tanto, foi definida uma extensão desse componente para o domínio em questão, em que o conceito de prioridade nas soluções de *matching* corresponde à existência do curso *Course5* nessas soluções, ou seja, aquelas que apresentem esse curso terão prioridade sobre as demais soluções obtidas. O Anexo D descreve a implementação dessa extensão, cujo nome é *PersonCoursePriorityEvaluator*.

A Figura 49 apresenta parte da requisição de *matching* para o caso *1toN*, que utiliza o avaliador de similaridade *PersonCoursePriorityEvaluator*. O que muda com relação ao exemplo apresentado na Figura 44 é a utilização do domínio de conhecimento construído com o avaliador de similaridade citado anteriormente.

```
<?xml version="1.0"?>
<Matchings>
  <Matching type="1toN" model="ONTOLOGY" name="Example_1toN_Priority">
    ...
    <Strategies>
      <TabuSearchStrategy name="Tabu Search Strategy" maximize="true">
        <DomainKnowledge name="PersonCourseDomainKnowledgeWithPriority" />
      </TabuSearchStrategy>
    </Strategies>
  </Matching>
</Matchings>
```

Figura 49 – Requisição de matching entre perfis de pessoas para o caso 1toN

A Tabela 9 apresenta os resultados obtidos com a execução do *matching* definido na requisição anterior. Como pode ser visto pela comparação com os resultados da Tabela 5 (para o caso *1toN* sem prioridade), as soluções obtidas tendem a possuir o curso *Course5*. Apenas para os casos em não seja possível a escolha desse curso é que os resultados se apresentam iguais entre as tabelas.

Pessoa	Melhor Solução	Valor Similaridade Melhor Solução	2ª Melhor Solução	3ª Melhor Solução
Luis	Course3 Course5 Course8	6.0 4.0 -700.0 1.0	Course2 Course3 Course5 Course8	Course2 Course3 Course5
Antônio	Course5 Course8	6.0 4.0 -550.0 2.0	Course5	-
Maria	Course3 Course5 Course8	7.5 5.0 -700.0 1.0	Course2 Course3 Course5 Course8	Course2 Course3 Course5
João	Course3 Course6 Course10 Course5	1.05 0.69 -850.0 -0.1	Course3 Course6 Course10	-
Tereza	Course5 Course4 Course7	7.5 5.0 -950.0 2.0	Course5 Course6 Course4 Course7	Course5 Course8 Course6 Course4 Course7
Ana	Course4	1.0 1.0 -300.0 1.0	-	-
Cristiano	Course1 Course3 Course2 Course5 Course4	9.0 6.0 -950.0 1.0	Course1 Course2 Course3 Course5	Course1 Course2

Tabela 9 – Resultados do matching 1toN com prioridade