

**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**PROTEIN LOCATOR: UM MÉTODO PARA  
CONSOLIDAÇÃO DE RESULTADOS NA IDENTIFICAÇÃO  
DE PROTEÍNAS**

**HIGOR DE SOUZA RODRIGUES**

**ORIENTADOR: WAGNER FONTES**

**DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA**

**PUBLICAÇÃO: 349/2008**

**BRASÍLIA / DF: JULHO/2008**



**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**PROTEIN LOCATOR: UM MÉTODO PARA  
CONSOLIDAÇÃO DE RESULTADOS NA IDENTIFICAÇÃO  
DE PROTEÍNAS**

**HIGOR DE SOUZA RODRIGUES**

DISSERTAÇÃO DE MESTRADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:

---

**WAGNER FONTES, Doutor, UnB  
(ORIENTADOR)**

---

**RICARDO STACIARINI PUTTINI, Doutor, UnB  
(EXAMINADOR INTERNO)**

---

**MARIA EMÍLIA MACHADO TELLES WALTER, Doutora, UnB  
(EXAMINADORA EXTERNO)**

**DATA: BRASÍLIA/DF, 30 DE JULHO DE 2008.**



## FICHA CATALOGRÁFICA

RODRIGUES, HIGOR DE SOUZA  
PROTEIN LOCATOR: UM MÉTODO PARA CONSOLIDAÇÃO DE RESULTADOS NA IDENTIFICAÇÃO DE PROTEÍNAS [Distrito Federal] 2008.

xix, 212p., 210 X 297 mm (ENE/FT/UnB, Mestre, Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia, 2008).

Departamento de Engenharia Elétrica.

1. Bioinformática  
3. Proteômica

2. Proteínas  
4. Protein Locator

I. ENE/FT/UnB.

II. Título (Série)

## REFERÊNCIA BIBLIOGRÁFICA

RODRIGUES, H. S. (2008). PROTEIN LOCATOR: UM MÉTODO PARA CONSOLIDAÇÃO DE RESULTADOS NA IDENTIFICAÇÃO DE PROTEÍNAS. Dissertação de Mestrado em Engenharia Elétrica, Publicação 349/2008, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 212p.

## CESSÃO DE DIREITOS

AUTOR: HIGOR DE SOUZA RODRIGUES

TÍTULO: PROTEIN LOCATOR: UM MÉTODO PARA CONSOLIDAÇÃO DE RESULTADOS NA IDENTIFICAÇÃO DE PROTEÍNAS.

GRAU: Mestre ANO: 2008

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta dissertação de mestrado pode ser reproduzida sem a autorização por escrito do autor.

---

Higor de Souza Rodrigues  
Rua 20 Norte, Lote 06 Apto. 1201 – Águas Claras  
CEP 71915-750 – Taguatinga – DF - Brasil



## **AGRADECIMENTOS**

Ao amigo Wagner Fontes que sempre acreditou em minha capacidade e me apoiou nos momentos difíceis dessa jornada.

Aos amigos da PGR, principalmente ao Vinícius e Lucas, que me ajudaram em tudo o que foi possível no decorrer desse trabalho. E também participaram de momentos de descontração nas horas difíceis.

Ao meu irmão, Renan, e ao Evandro por toda a ajuda técnica que me deram durante este projeto, até mesmo quando eles também estavam atrasados com seus trabalhos.

Aos professores Adson, Anderson e Puttini, pelo apoio no processo de aceitação neste programa de mestrado.

Ao pessoal do Laboratório de Bioquímica que sempre que possível me ajudaram nos desafios da Bioquímica.

A todos os professores que tive ao longo do curso, pela contribuição em minha formação acadêmica.

Agradecimento especial para minha esposa Marina, meus pais, Alexandrina e José Carlos, e meus irmãos, Alice e Renan, pelo apoio incondicional em todos os momentos.

Agradeço a Jah por tudo em todos os momentos da minha vida.

Dedico este trabalho a todas as pessoas que possam se beneficiar com os resultados das novas pesquisas de identificação de medicamentos e métodos de curas.

## RESUMO

**Protein Locator: um método para consolidação de resultados na identificação de proteínas**

**Autor: Higor de Souza Rodrigues**

**Orientador: Wagner Fontes**

**Programa de Pós-graduação em Engenharia Elétrica**

**Brasília, julho de 2008**

Um dos papéis mais importantes da Bioinformática proteômica pode ser descrito como o tratamento do conjunto de dados gerado a partir do sequenciamento de proteínas, construindo de forma eficaz e organizada, informações inteligíveis para os pesquisadores dessa área. Existem diversos bancos de dados de seqüências, como o EMBL, o SwissProt e o UniProt, bem como diferentes programas para realizar buscas por similaridades nestes bancos de dados, como o Mascot, o Fasta, o Blast e AACompIdent. O objetivo deste estudo foi construir um sistema inédito que apresente de maneira probabilística a similaridade entre proteínas que constituem os bancos de dados pré-existent e os dados experimentais fornecidos pelos pesquisadores. A partir da inserção dos dados, o sistema, chamado Protein Locator, busca as seqüências similares nos programas já existentes, e utiliza o algoritmo QFAST de combinação de p-valores e também o algoritmo PLscore, uma nova versão do QFAST proposto por este estudo, para a combinação de todos os resultados obtidos. Os algoritmos realizam a combinação das probabilidades dos resultados fornecidos pelos programas de identificação e o Protein Locator apresenta ao usuário os valores originais de cada programa e o valor consolidado pela combinação dos resultados, sendo formado pelo identificador da proteína e a probabilidade de erro do match.

Para a validação do método de combinação de resultados e do algoritmo PLscore, foram realizadas pesquisas de identificação de 18 conjuntos de dados de experimentos teóricos com proteínas que simularam seu sequenciamento, análise de composição de aminoácidos e obtenção da lista de massa de peptídeos. Em 9 desses experimentos, foram incluídos desvios laboratoriais e nos outros 9 foram utilizadas as informações completas. Em 14 dos 18 resultados, a combinação dos dados possibilitou o aumento na acurácia do resultado; em 4 casos, não houve mudanças nas conclusões das pesquisas e em nenhum caso houve piora dos resultados. O tempo entre o armazenamento de informações das pesquisas e a espera pelos resultados combinados foi de aproximadamente 30 minutos, bastante inferior ao tempo medido para se realizar um experimento semelhante de forma manual, cerca de 3 horas.

## **ABSTRACT**

**Protein Locator: um método para consolidação de resultados na identificação de proteínas**

**Author: Higor de Souza Rodrigues**

**Supervisor: Wagner Fontes**

**Programa de Pós-graduação em Engenharia Elétrica**

**Brasília, July 2008**

The analysis of protein sequencing data is one of the most important roles of proteomic bioinformatics. In addition, bioinformatics organizes data in an optimized way to be used by researches in this area. There are some protein databases, such as EMBL, SwissProt and Uniprot with software to search for sequencing similarities such as Mascot, Fasta, Blast and AACompIdent. The aim of this study was to create a new system to calculate statistical similarity degree between proteins described in databases and experimental data. The system, called Protein Locator, compares experimental data with sequences through the preexisting software and uses both the QFAST p-value combination algorithm and the PLscore algorithm (a new version of QFAST proposed by this study) to combine results. The algorithms combine probability between the results from the sequences search software and Protein Locator shows the original p-values from each software, the p-value obtained from results combination, and also the protein identifier and the probability of match.

To evaluate the results combination method and the PLscore algorithm, we have used 18 data collections from theoretical experiments in which protein sequencing, analysis of amino acids composition and peptides mass were simulated. In 9 of these experiments, we have included the laboratory error and in the other 9 we have used the complete data. In 14 out the 18 results, data combination method increased accuracy; in the other 4, results were equivalent to those found without combination. Combination of results and protein identification required 30 minutes from laboratory data insertion while manual search would usually require approximating 3 hours.

## ÍNDICE

<b>1.</b>	<b>INTRODUÇÃO .....</b>	<b>17</b>
<b>1.1.</b>	<b>CARACTERIZAÇÃO DO PROBLEMA .....</b>	<b>17</b>
<b>1.2.</b>	<b>OBJETIVOS .....</b>	<b>18</b>
<b>1.3.</b>	<b>ORGANIZAÇÃO DO TRABALHO .....</b>	<b>21</b>
<b>2.</b>	<b>CONCEITOS BÁSICOS EM PROTEÔMICA .....</b>	<b>23</b>
<b>2.1.1.</b>	<b>Proteômica .....</b>	<b>23</b>
<b>2.1.2.</b>	<b>Bioinformática.....</b>	<b>28</b>
<b>2.2.</b>	<b>TÉCNICAS DE IDENTIFICAÇÃO DE PROTEÍNAS .....</b>	<b>30</b>
<b>2.3.</b>	<b>PROGRAMAS UTILIZADOS PARA IDENTIFICAÇÃO DE PROTEÍNAS .....</b>	<b>33</b>
<b>3.</b>	<b>CONCEITOS BÁSICOS EM COMPUTAÇÃO .....</b>	<b>36</b>
<b>3.1.</b>	<b>BANCOS DE DADOS.....</b>	<b>36</b>
<b>3.1.1.</b>	<b>Principais formas de armazenamento de dados proteômicos e genômicos .....</b>	<b>37</b>
<b>3.1.2.</b>	<b>Bancos de dados de proteínas .....</b>	<b>38</b>
<b>3.2.</b>	<b>SERVIDORES WEB .....</b>	<b>40</b>
<b>3.3.</b>	<b>LINGUAGEM PHP .....</b>	<b>41</b>
<b>3.4.</b>	<b>ALGORITMO QFAST.....</b>	<b>43</b>
<b>4.</b>	<b>REVISÃO BIBLIOGRÁFICA .....</b>	<b>46</b>
<b>5.</b>	<b>METODOLOGIA .....</b>	<b>52</b>
<b>5.1.</b>	<b>METODOLOGIA DE DESENVOLVIMENTO DO SISTEMA.....</b>	<b>52</b>

5.1.1. Visão geral .....	52
5.1.2. Desenvolvimento do software .....	53
5.2. ADICIONANDO SERVIÇOS AO PROGRAMA.....	59
5.3. UTILIZAÇÃO DO SISTEMA .....	61
6. RESULTADOS E DISCUSSÕES .....	67
6.1. AMBIENTE DE TESTE .....	67
6.2. METODOLOGIA DE TESTE.....	67
6.3. DESCRIÇÃO DAS PROTEÍNAS UTILIZADAS .....	68
6.4. RESULTADOS DOS TESTES DE IDENTIFICAÇÃO .....	72
7. CONCLUSÕES E RECOMENDAÇÕES .....	90
8. REFERÊNCIAS BIBLIOGRÁFICAS .....	92
A. DOCUMENTAÇÃO DO SOFTWARE E CASOS DE USO.....	98
B. – DOCUMENTAÇÃO DO BANCO DE DADOS.....	155
C. MANUAL DO ADMINISTRADOR.....	209

## ÍNDICE DE TABELAS

Tabela 2-1 Aminoácidos e seus códigos de uma e três letras .....	27
Tabela 2-2 Exemplo de lista de massas.....	32
Tabela 5-1 Tabela de priorização das atividades .....	54
Tabela 6-1 Resultados da busca com dados completos da proteína P33956 .....	73
Tabela 6-2 Resultados da busca com dados parciais da proteína P33956 .....	74
Tabela 6-3 Resultados comparativos dos métodos QFAST e Fisher .....	75
Tabela 6-4 Resultados da busca com dados completos da proteína Q7A781 .....	75
Tabela 6-5 Resultados da busca com dados parciais da proteína Q7A781 .....	76
Tabela 6-6 Resultados da busca com dados completos da proteína P80674 .....	77
Tabela 6-7 Resultados da busca com dados parciais da proteína P80674 .....	78
Tabela 6-8 Resultados da busca com dados completos da proteína P01024 .....	79
Tabela 6-9 Resultados da busca com dados parciais da proteína P01024 .....	80
Tabela 6-10 Resultados da busca com dados completos da proteína A6WMJ7 .....	81
Tabela 6-11 Resultados da busca com dados parciais da proteína A6WMJ7 .....	81
Tabela 6-12 Resultados da busca com dados completos da proteína Q8Z937 .....	82
Tabela 6-13 Resultados da busca com dados parciais da proteína Q8Z937 .....	83
Tabela 6-14 Resultados da busca com dados completos da proteína O46903 .....	84
Tabela 6-15 Resultados da busca com dados completos da proteína O46903 .....	85
Tabela 6-16 Resultados da busca com dados completos da proteína Q8K019 .....	86
Tabela 6-17 Resultados da busca com dados parciais da proteína Q8K019 .....	87
Tabela 6-18 Resultados da busca com dados completos da proteína A1BAN4 .....	88
Tabela 6-19 Resultados da busca com dados parciais da proteína A1BAN4 .....	89

## ÍNDICE DE FIGURAS

Figura 1-1 Visão geral do sistema.....	19
Figura 1-2 Diagrama de atividades da identificação de proteínas .....	20
Figura 1-3 Diagrama de atividades da identificação por meio do sistema Protein Locator.....	21
Figura 2-1 Estrutura de dupla hélice do DNA.....	24
Figura 2-2 Processo de transcrição dos genes em RNA .....	25
Figura 2-3 Processo de tradução de RNA para proteína. ....	26
Figura 2-4 Representação do aminoácido “Aspartato” .....	27
Figura 2-5 Exemplo Eletroforese 2-D .....	31
Figura 2-6 Etapa durante o seqüenciamento por degradação de Edman .....	32
Figura 2-7 Etapa durante a análise da composição de aminoácidos .....	33
Figura 3-1 Exemplo de seqüência em formato FASTA.....	38
Figura 3-2 Utilização de servidores web no mundo. ....	41
Figura 3-3 Utilização do PHP nos servidores ao redor do mundo.....	43
Figura 3-4 Equação para combinação de p-valores .....	45
Figura 3-5 Algoritmo QFAST.....	45
Figura 5-1 Criação de novo usuário .....	61
Figura 5-2 Tela de <i>login</i> de usuário .....	62
Figura 5-3 Visualização das pesquisas do usuário .....	62
Figura 5-4 Criação de uma pesquisa .....	63
Figura 5-5 Possíveis próximas etapas .....	63
Figura 5-6 Adicionar composição de aminoácidos .....	64
Figura 5-7 Adicionar informações de <i>fingerprint</i> .....	64
Figura 5-8 Adicionar informações de seqüência de proteína.....	65
Figura 5-9 Visualizar informações detalhadas .....	65
Figura 5-10 Sucesso na submissão de pesquisa .....	66
Figura 5-11 Resultados consolidados.....	66

## ÍNDICE DE ABREVIATURAS UTILIZADAS

2D – BIDIMENSIONAL

BLAST – *BASIC LOCAL ALIGNMENT SEARCH TOOL*

DNA – ACIDO DESOXIRRIBONUCLEÍCO

FASTA – *FAST ALIGNMENT SEARCH TOOL*

GUI – INTERFACE GRÁFICA DO USUÁRIO (*GRAPHICAL USER INTERFACE*)

IUPAC – UNIÃO INTERNACIONAL DE QUÍMICA PURA E APLICADA (*INTERNATIONAL UNION OF PURE AND APPLIED CHEMISTRY*)

LC – CROMATOGRAFIA LÍQUIDA (*LIQUID CROMATROGRAPHY*)

MS – ESPECTROMETRIA DE MASSA (*MASS SPECTROMETRY*)

MW – MASSA MOLECULAR (*MOLECULAR WEIGHT*)

pH - POTENCIAL HIDROGENIÔNICO

PI – PONTO ISOELÉTRICO

PL – PROTEIN LOCATOR

PMF – *PEPTIDE MASS FINGERPRINT* (LISTA DE MASSAS DE PEPTIDEOS)

RNA – ACIDO RIBONUCLEÍCO

UC – CASO DE USO

UCD – DIAGRAMA DE CASO DE USO



# 1. INTRODUÇÃO

## 1.1. CARACTERIZAÇÃO DO PROBLEMA

Uma característica bioquímica fundamental comum a todos os organismos é o uso de DNA (ácido desoxirribonucléico) para armazenar informações genéticas. Watson e Crick propuseram, em 1953, a estrutura do DNA, composta por um arranjo tridimensional de dois filamentos [1]. Os filamentos são polímeros lineares constituídos por quatro tipos diferentes de monômeros (nucleotídeos contendo as seguintes bases nitrogenadas): adenina (A), citosina (C), guanina (G) e timina (T). O pareamento específico dessas bases na dupla hélice (as ligações são sempre estabelecidas entre C-G e A-T) possibilita determinar a seqüência dos monômeros no filamento pareado. Essa característica é fundamental para a conservação da informação genética durante a reprodução celular, pois cada um dos filamentos, após uma separação entre eles, pode servir de base para a construção de seu novo par.

A seqüência dessas bases é a forma de armazenamento da informação genética. Ela determina a seqüência das moléculas de ácido ribonucléico (RNA), por um processo conhecido como transcrição, que, por fim, determina a seqüência de aminoácidos das proteínas produzidas nos organismos, por meio do processo de tradução. Esses processos serão mais detalhados no capítulo 2 desta dissertação.

O conhecimento da seqüência de aminoácidos de uma proteína é importante por diversos motivos. Primeiro, para elucidar seu mecanismo de ação. Proteínas com novas funcionalidades podem ser geradas pela alteração de seqüências de proteínas conhecidas. Segundo, porque a seqüência de aminoácidos é um dos determinantes da estrutura tridimensional da proteína, por meio das interações entre eles. Terceiro, a determinação da seqüência faz parte dos estudos de patologia molecular. As alterações de seqüência podem produzir função anormal de proteínas e causar doenças, sendo que algumas fatais, como a anemia falciforme e a fibrose cística, que podem ser resultado da alteração de apenas um aminoácido dentro de uma proteína. Por fim, a seqüência de uma proteína revela informações sobre sua história evolutiva, pois as proteínas que se assemelham umas às outras em sua seqüência têm um ancestral em comum [2].

Para se identificar proteínas com segurança no resultado, pode ser necessário utilizar mais de um programa de identificação e, para aumentar ainda mais a confiança, utilizar diferentes técnicas de identificação na mesma pesquisa. Segundo as recomendações da editoria da revista *Molecular & Celular Proteomics*, Steven Carr e colaboradores [3], para que uma publicação seja aceita nesta revista, é necessário realizar uma série de procedimentos durante a pesquisa, inclusive, identificar a proteína utilizando mais de um programa.

Para que o cientista utilize diferentes programas, é necessário que ele verifique as condições de submissão de pesquisas em cada programa que desejar utilizar, acesse a página *web* do programa, preencha o formulário com as informações, submeta e aguarde o resultado. A página de resultados possui uma série de informações, sendo necessário estabelecer um padrão para aceitação do resultado. Após essa primeira identificação, o cientista precisa realizar o mesmo procedimento para os demais programas que deseje utilizar.

Os resultados de cada programa são apresentados em páginas *web*. Para que o cientista armazene-os, é necessário que seja estabelecido um método de armazenamento de dados. Após obter todos os resultados necessários, cabe ainda, ao cientista, realizar uma análise estatística dos resultados para definir a real proteína identificada.

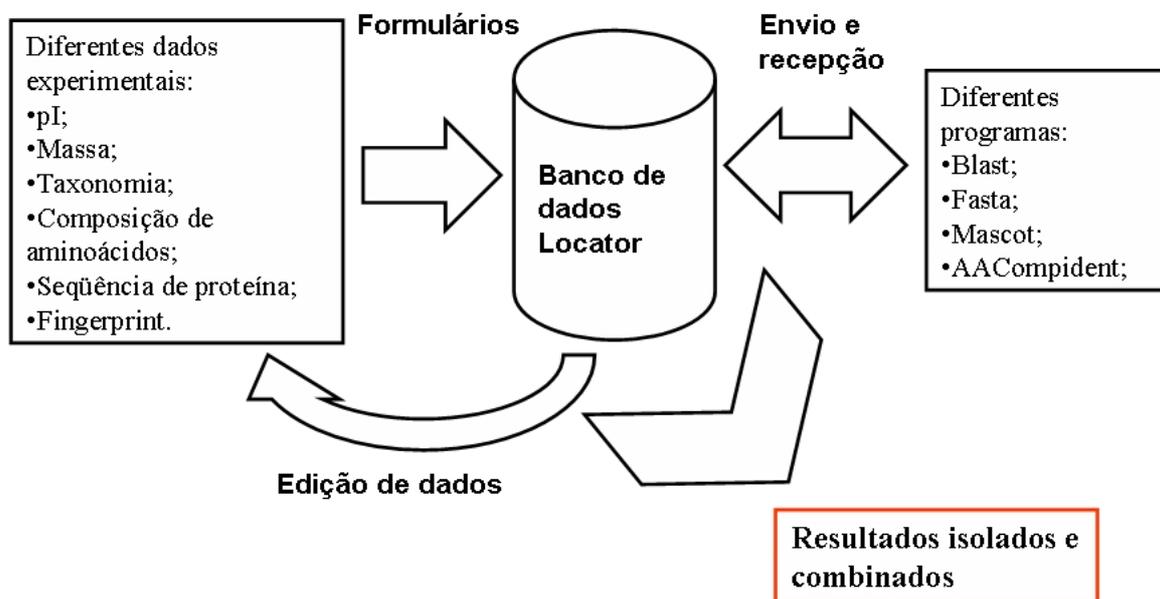
## 1.2. OBJETIVOS

O objetivo deste projeto é aumentar a probabilidade de acerto na identificação de proteínas, de acordo com o constatado por González e colaboradores [4], por meio da combinação dos resultados de diferentes programas de identificação de proteínas.

Para tanto, deverá ser produzido um sistema que gerencie as informações das pesquisas do cientista e permita a consolidação dos resultados por meio da combinação dos resultados obtidos por diferentes programas de identificação de proteínas, abordagem atualmente conhecida como *proteomics pipeline* [5].

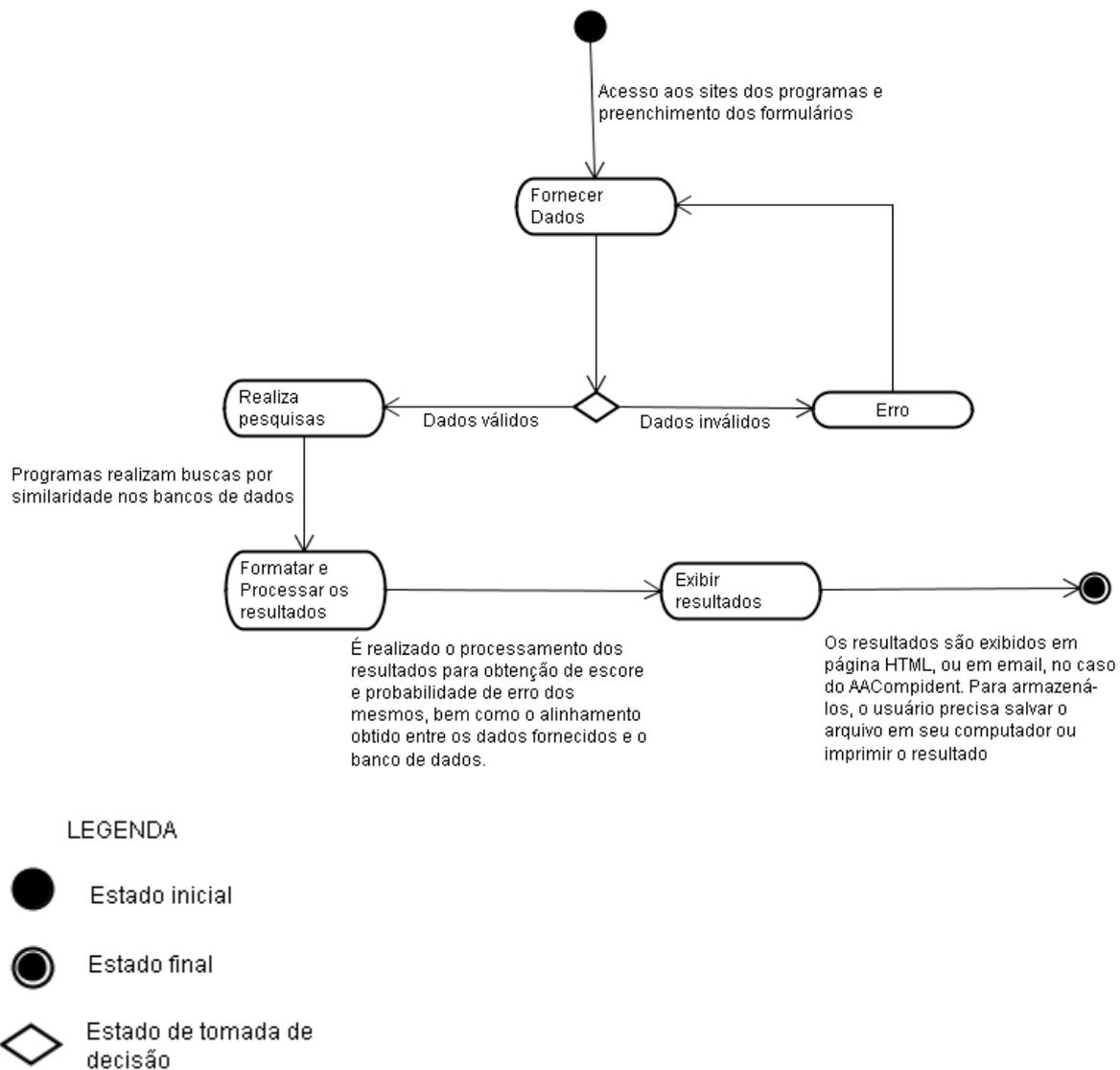
Esta iniciativa é pioneira, uma vez que os experimentos de proteômica realizados atualmente utilizam, de forma manual, mais de um programa de identificação apenas para comprovar o resultado do primeiro programa utilizado, sem que os mesmos sejam combinados.

A visão geral deste sistema é da seguinte forma:



**Figura 1-1 Visão geral do sistema**

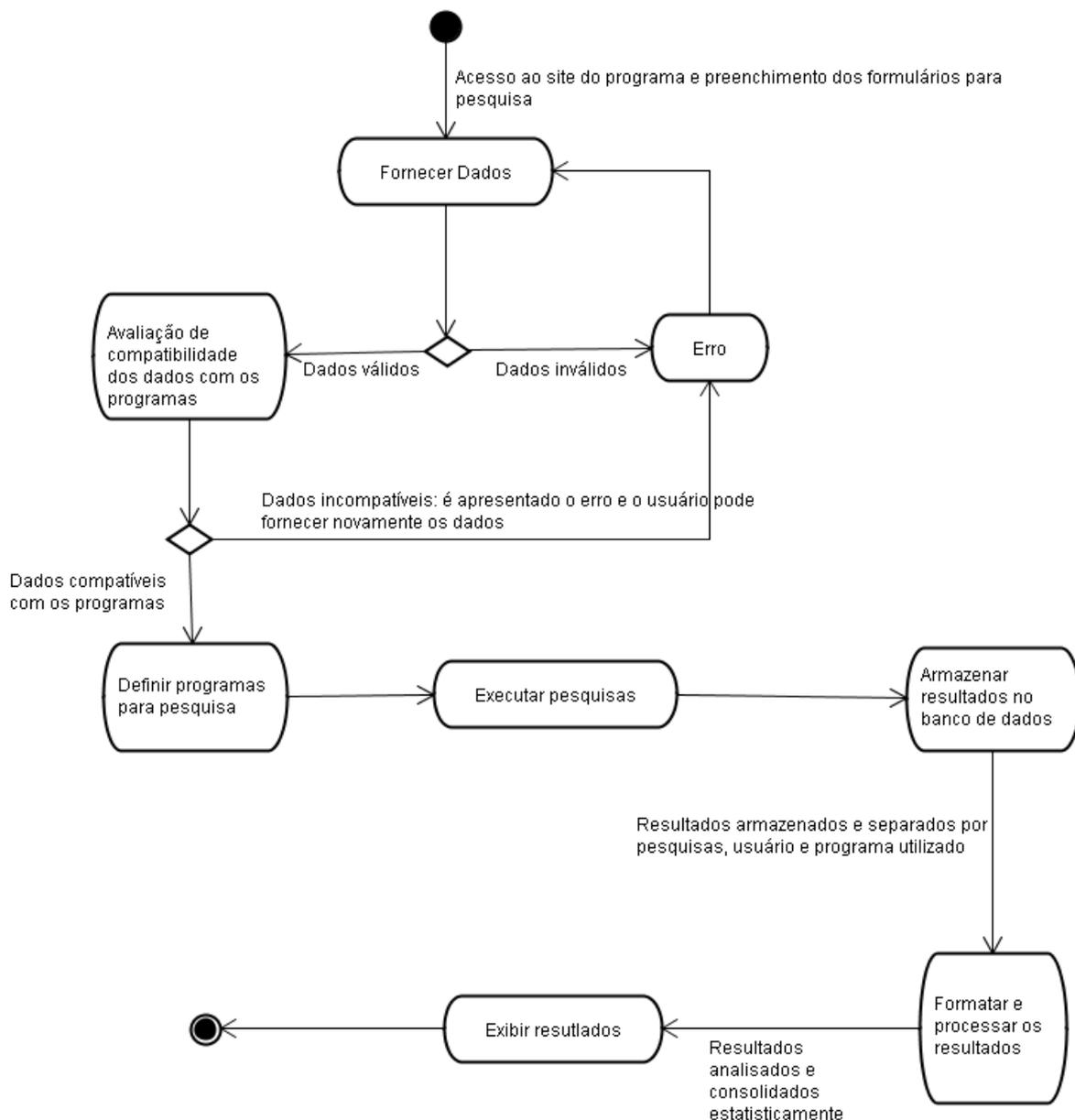
O foco do projeto é a facilitação e o aprimoramento das buscas para identificação de proteínas, realizadas por profissionais de laboratórios de pesquisas em bioquímica. Atualmente, os experimentos realizados para identificação de proteínas seguem o seguinte fluxo de atividades:



**Figura 1-2 Diagrama de atividades da identificação de proteínas**

Na realização da pesquisa, os cientistas devem seguir o diagrama acima para cada um dos programas que desejam utilizar na identificação de proteínas. Frequentemente, é utilizado apenas um programa de identificação ou o segundo programa é utilizado apenas para confirmar o resultado do primeiro.

O projeto objetiva construir um sistema que possibilite a utilização, de forma automática, de várias ferramentas de identificação, simultaneamente, para a mesma pesquisa, realizando o armazenamento dos resultados originais e a consolidação estatística dos mesmos, facilitando a tomada de decisão por parte do cientista. A utilização do sistema segue o seguinte fluxo de atividades:



**Figura 1-3 Diagrama de atividades da identificação por meio do sistema Protein Locator**

### 1.3. ORGANIZAÇÃO DO TRABALHO

Este capítulo aborda a contextualização, objetivos do projeto e visão geral. No capítulo 2, são apresentados alguns conceitos básicos em proteômica e, no capítulo 3, são apresentados alguns conceitos básicos em computação.

No capítulo 4, é realizada a revisão bibliográfica, com citações de referências para os programas de identificação de proteínas abordados no projeto e outros programas com

funcionalidades que são englobadas pelo projeto, apesar de não serem alvo dos algoritmos deste trabalho.

O capítulo 5 apresenta a metodologia de desenvolvimento de *software* utilizada no projeto, bem como as funcionalidades do sistema e algumas figuras ilustrativas das mesmas.

O capítulo 6 é crucial, pois apresenta os testes realizados, os resultados e a análise dos mesmos, que comprovam o alcance dos objetivos propostos e a forma como isso pôde ser medido. Esta etapa requereu amplas discussões entre os membros do projeto e cientistas do laboratório de bioquímica, visando apresentar dados realmente relevantes para a avaliação do sistema.

No capítulo 7, o foco é a conclusão das análises realizadas no projeto e a indicação de trabalhos futuros que poderão melhorar ainda mais o sistema.

Os apêndices desta dissertação estão bastante ricos em descrição do sistema. O **Apêndice A** apresenta a especificação funcional, abordando todos os casos de uso, regras de negócio e os principais cenários do sistema. O **Apêndice B** especifica o banco de dados desenvolvido neste projeto, detalhando as entidades (tabelas do banco de dados) e seus relacionamentos. Por fim, o **Apêndice C** procura tornar possível a administração do sistema por usuários capacitados, incluindo as instruções para instalação do *software*, para desenvolvimento de novas funcionalidades e a estrutura de arquivos utilizados pelo sistema.

## 2. CONCEITOS BÁSICOS EM PROTEÔMICA

### 2.1.1. Proteômica

O proteoma é o conjunto das proteínas expressas pelo genoma de um organismo, grupo de células ou secreção, em uma determinada situação fisiológica [6]. Proteômica é o estudo das variações quantitativas dos níveis de expressão das proteínas e suas modificações pós-traducionais (o proteoma não é conservado em todas as células do organismo) [7]. As suas aplicações são freqüentemente utilizadas na descoberta de novas drogas, diagnósticos e terapias para tratamento de doenças [8]. A palavra proteômica é formada pela mistura de “proteins” e “genomics” e foi criada pelo professor Marc Wilkins [9] no início dos anos 90. Nos anos 50 já era feito o seqüenciamento de aminoácidos por meio da Degradação de Edman e os primeiros programas de computador para auxílio na interpretação de resultados do seqüenciamento apareceram, permitindo o início da identificação das proteínas que viriam a ser aplicados futuramente nos estudos dos proteomas [10].

Algumas das perspectivas de aplicações da proteômica compreendem estudos farmacêuticos de novas drogas que têm como alvo proteínas identificadas. A validação dos alvos de drogas identificados, estudos de toxicologia *in-vitro* e *in-vivo* e estudos dos efeitos colaterais podem ser melhorados com ajuda da proteômica [11].

A hipótese de Watson e Crick [1] só foi realmente comprovada nos anos 90, com a determinação de seqüências genômicas completas de centenas de organismos diferentes, desde microorganismos simples a animais mais complexos. Estes seqüenciamentos foram realizados por pesquisas em projetos de genomas.

O genoma é a lista completa das bases nucleotídicas que compõem genes ou regiões intergênicas, que, por sua vez, compõem regiões de um filamento de DNA. O proteoma é a representação funcional do genoma, abrangendo todos os tipos, funções e interações de proteínas de um organismo.

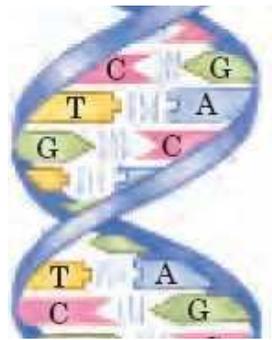
As proteínas são moléculas grandes e complexas, indispensáveis às funções vitais. Elas estão envolvidas nos mais diversos processos biológicos, desde a movimentação (ex: actina e miosina, proteínas associadas à contração muscular), percepção do ambiente (ex: diversos mecanismos fotossensíveis em animais são dependentes de proteínas) até os

mecanismos de defesa contra infecções (ex: anticorpos, os quais são proteínas) e de ataque (ex: diversas toxinas de microorganismos são de natureza protéica) [2].

Cada proteína é formada, originalmente, como uma seqüência de aminoácidos, cuja identificação e ordem são preditas, em parte, pelos genes, de acordo com a seqüência de bases presentes no DNA.

O DNA é um polímero linear constituído por quatro tipos de bases nucleotídicas: adenina (A), citosina (C), guanina (G) e timina (T), que se organizam numa dupla hélice formada por dois filamentos de bases entrelaçadas. A seqüência de bases ao longo do filamento atua como uma forma de armazenar a informação genética.

A figura abaixo ilustra a dupla hélice do DNA, em que as bases nucleotídicas estão pareadas: C – G e T – A.



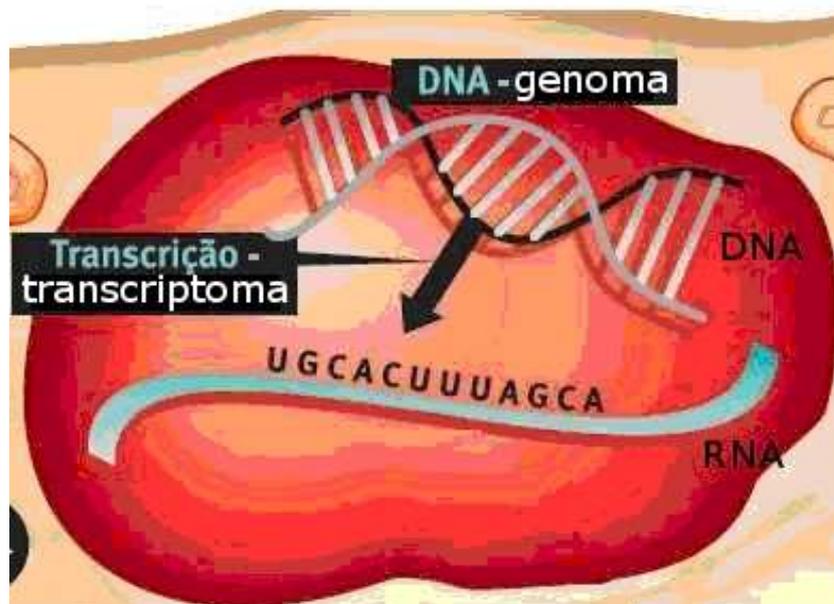
**Figura 2-1 Estrutura de dupla hélice do DNA** Fonte: Lehninger Biochemistry 4ª edição 2005, página 30

A seqüência de DNA determina a seqüência das moléculas de RNA (ácido ribonucléico) e as seqüências de RNA, são traduzidas em cadeias lineares de proteínas, num processo que será descrito detalhadamente em seguida.

A codificação de cada um dos aminoácidos das proteínas é realizada pela expressão de um conjunto, chamado de códon, com 3 bases ao longo do filamento do RNA (derivado do filamento de DNA específico). Esta relação existente entre a seqüência de DNA e a seqüência codificada da proteína é chamada de código genético. Apenas uma pequena parte do material genético codifica as proteínas, cerca de 3% do genoma humano. Ao restante do DNA cabem importantes funções de regular a expressão de genes específicos (que, por conseguinte, produzem proteínas específicas) em tipos celulares e condições fisiológicas particulares, sendo este mecanismo conhecido como expressão gênica. Apesar de praticamente todas as células conterem o mesmo material genético, tipos celulares diferem consideravelmente

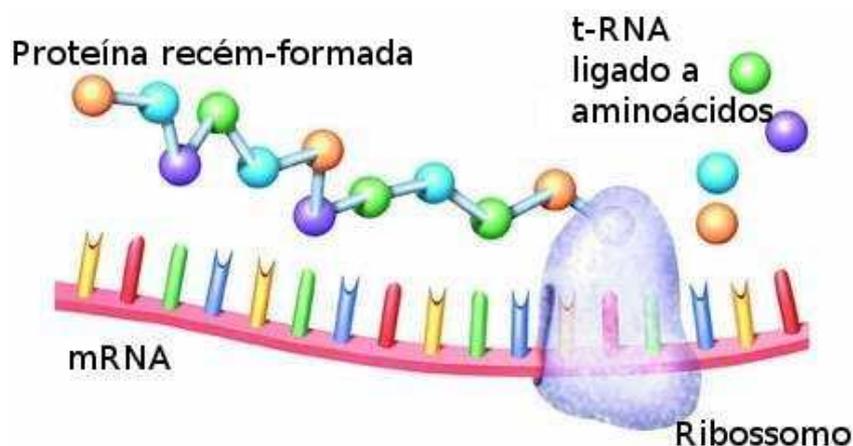
quanto às proteínas que produzem, ou seja, existem diferenças na expressão gênica entre as células. A expressão é regulada pela presença de moléculas sinalizadoras (hormônios, citocinas, etc.) junto às células.

No processo de transcrição, as seqüências lineares de genes são transcritas em moléculas lineares de ácido ribonucléico, com a seqüência complementar de ribonucleotídeos: no caso do RNA, a complementação é feita entre C-G e A-U (a timina é substituída por uracila no RNA). As moléculas transcritas de RNA podem ser de três tipos: RNA mensageiro (mRNA), RNA ribossômico (rRNA) e RNA transportador (tRNA). Os três tipos participam da síntese de proteínas, porém é o RNA mensageiro quem codifica a seqüência da proteína a ser produzida. Na figura abaixo, pode-se observar o processo de transcrição, em que uma seqüência de DNA é transcrita em uma seqüência de RNA:



**Figura 2-2 Processo de transcrição dos genes em RNA** Fonte: Proteoma, Ciência hoje página 22 (com adaptações)

No processo de tradução, cada códon, conjunto de 3 bases ao longo do filamento de mRNA, codifica um aminoácido específico dentre 20 possibilidades apresentadas na tabela 2-1, por meio de uma ligação entre o tRNA e o mRNA. A seguir, é apresentada uma ilustração do processo de tradução (que ocorre com maior freqüência no ambiente do ribossomo celular), em que uma seqüência de RNA é traduzida em uma seqüência de proteína.



**Figura 2-3 Processo de tradução de RNA para proteína.** Fonte: [www.dorlingkindersley-uk.co.uk/](http://www.dorlingkindersley-uk.co.uk/) (com modificações)

As cadeias lineares de proteínas, formadas a partir da tradução do RNA, se enovelam formando estruturas tridimensionais e, após o enovelamento, podem se ligar a outras proteínas por meio de fortes interações.

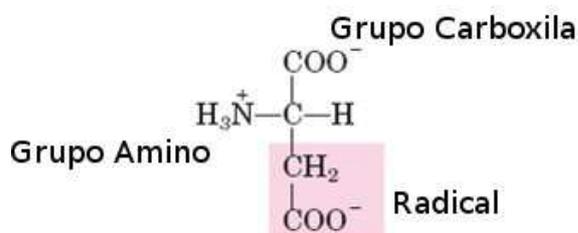
A estrutura primária da proteína é caracterizada por uma seqüência de aminoácidos que ligados formam cadeias peptídicas e essa seqüência de aminoácidos é um dos fatores que determina a estrutura tridimensional da proteína, por meio das interações entre eles. Além da estrutura primária, existe a secundária, em que as cadeias peptídicas podem se dobrar em estruturas regulares, a terciária, em que proteínas hidrossolúveis se enovelam em estruturas compactas com interior apolar e a estrutura quaternária, em que cadeias peptídicas se associam em estruturas de múltiplas subunidades.

Os aminoácidos são as unidades básicas das proteínas. Cada um deles é constituído de um Carbono central ligado a um grupamento amina ( $\text{NH}_3^+$ ), uma carboxila ( $\text{COO}^-$ ), um átomo de hidrogênio (H) e um radical (R), sendo este o que diferencia um aminoácido de outro. As vinte diferentes cadeias R encontradas freqüentemente em proteínas variam em tamanho, forma, carga, capacidade de formação de pontes de hidrogênio, caráter hidrofóbico e reatividade química. A fim de unificar a representação simplificada dos aminoácidos, facilitando os desenvolvedores de sistemas, a IUPAC (*International Union of Pure and Applied Chemistry*) [12] criou uma tabela contendo a lista com os aminoácidos representados por um código de uma ou três letras, dependendo da aplicação desenvolvida. A seguir, a tabela com esta representação:

**Tabela 2-1 Aminoácidos e seus códigos de uma e três letras**

Código de uma letra	Código de três letras	Nome do aminoácido
A	Ala	Alanina
R	Arg	Arginina
N	Asn	Asparagina
D	Asp	Ácido Aspártico
C	Cys	Cisteína
Q	Gln	Glutamina
E	Glu	Ácido Glutâmico
G	Gly	Glicina
H	His	Histidina
I	Ile	Isoleucina
L	Leu	Leucina
K	Lys	Lisina
M	Met	Metionina
F	Phe	Fenilalanina
P	Pro	Prolina
S	Ser	Serina
T	Thr	Treonina
W	Trp	Triptofano
Y	Tir	Tirosina
V	Val	Valina
B	Asx	Ácido Aspártico ou Asparagina
Z	Glx	Ácido Glutâmico ou Glutamina
X	Xaa	Qualquer Aminoácido

Cada um dos aminoácidos possui uma estrutura diferenciada e propriedades específicas, como ponto isoelétrico, peso molecular e carga. A seguir, uma ilustração de uma molécula de aminoácido, com o grupo amino e a carboxila e o radical marcado em tom de rosa.



**Figura 2-4 Representação do aminoácido “Aspartato”** Fonte: Lehninger Biochemistry 4ª edição 2005, página 10 (com modificações)

O ponto isoelétrico (pI) corresponde ao pH em que uma molécula apresenta carga elétrica líquida igual a zero, ou seja, há equilíbrio entre as cargas positivas e negativas na

molécula [13]. O pI de uma molécula pode afetar sua solubilidade em água e a capacidade de interagir com outros compostos dependendo do meio em que esteja [14]. A análise de proteínas feita por eletroforese bidimensional (2D-PAGE) utiliza as propriedades elétricas da amostra, separando as proteínas em um gradiente de pH em uma de suas dimensões. Outra característica importante, utilizada para separação de proteínas, é a massa molecular (MW – *molecular weight*), que é a soma das massas de todos os elementos da molécula em questão. Alguns aminoácidos apresentam-se, em certas condições fisiológicas, com carga elétrica positiva, outros, com carga negativa e ainda existem os eletricamente neutros. A interação entre cadeias de cargas opostas são chamadas de pontes salinas, existindo nas proteínas aproximadamente a cada 30 resíduos de aminoácidos [15].

A seqüência, a composição de aminoácidos, bem como a massa molecular de proteínas encontradas em organismos não interligados evolutivamente é bastante diferente. Por outro lado, proteínas com a mesma atividade em organismos evolutivamente próximos freqüentemente apresentam elevado grau de similaridade. Dessa forma, cada tipo de organismo produz proteínas que podem nos fornecer características para identificá-los e determinar o grau de semelhança entre organismos ou mesmo entre moléculas [2].

A identificação de proteínas também é uma importante fonte de informação para a área médica. Um dos exemplos reside no estudo de doenças genéticas, que podem ser causadas por uma proteína mutante, a qual contém uma seqüência ou uma composição de aminoácidos diferentes da proteína normal, que deveriam ocupar o lugar originalmente.

Projetos de análise de proteomas têm crescido juntamente com o término de seqüenciamentos completos de genomas. Projetos de proteomas revelam quais genes são expressos nas células na forma de proteínas e, experimentos mais aprofundados, podem fornecer informações sobre diferentes formas de expressão dos genes em proteínas. A plenitude do seqüenciamento de genomas permite a análise de diferentes proteomas [16].

### **2.1.2. Bioinformática**

Existem várias definições na literatura para esta ciência. Uma definição bem aceita é a de Luscombe e colaboradores [17], que define a Bioinformática como uma união entre biologia e informática envolvendo tecnologias computacionais de armazenamento de dados, manipulação e distribuição de informações relacionadas a macromoléculas como DNA, RNA e proteínas [18]. O papel da Bioinformática nos projetos de análise de proteomas envolve o

armazenamento e a manipulação de grande quantidade de informações, que incluem imagens de géis bidimensionais, cromatogramas, espectros de massa e a disponibilização de informações de proteínas já identificadas, tais como sua massa, pI, composição e seqüência de aminoácidos, até a determinação e exibição de estruturas 3-D para visualização de proteínas.

Assim como temos os estudos biológicos *in-vivo*, realizados em organismos vivos e os estudos *in-vitro* em meios artificiais, a Bioinformática pode ser considerado o estudo da biologia molecular *in-silico*, realizado por microprocessadores. O que diferencia a Bioinformática da biologia computacional é a sua limitação à análise de estruturas, seqüência e funções de genes e genomas e seus correspondentes protéicos (proteínas traduzidas e proteomas) [18].

Para a distribuição de informações de genomas e proteomas, é indispensável a aplicação da Bioinformática, pois é esta a ciência responsável pelo armazenamento das informações em bancos de dados e disponibilização desses para consultas pela internet. Os avanços das pesquisas são favorecidos pela maior distribuição dos dados, em bancos públicos e por meio de ferramentas de busca e análise de resultados.

O grande foco das análises em Bioinformática é viabilizar o processamento e a compreensão de dados, gerados em grande volume por experimentos de genômica e proteômica, e viabilizar a interpretação desses dados a fim de levar à melhor compreensão dos sistemas vivos e suas funções celulares. As funções celulares sempre envolvem a participação de proteínas, cuja característica estrutural e funcional provém de suas seqüências de aminoácidos. As análises desempenhadas pelas ferramentas computacionais têm aplicação no desenvolvimento de uma base de conhecimento para novas drogas, análises de DNA e biotecnologia em geral, como para a agricultura.

Dessa forma, os objetivos da Bioinformática são: desenvolvimento de ferramentas computacionais e bancos de dados e a aplicação destes na geração de conhecimento biológico para melhor entender os sistemas vivos. As ferramentas computacionais incluem programas para análise de seqüenciamento, de estruturas e de funcionalidades de moléculas biológicas. [18]

Os avanços da Bioinformática possibilitaram: a transformação de bancos de dados primários de proteínas, que se apresentam como arquivos de texto puro, para bancos de dados secundários, que são estruturados e com acesso livre; a criação de ferramentas *web* para

acesso às informações dos bancos de dados de proteínas; a criação de diversas ferramentas para localizar seqüências de proteínas por suas diferentes características; a evolução dos equipamentos para espectrometria de massa e das ferramentas para análise de géis 2D (técnicas que serão abordadas posteriormente nesta dissertação).

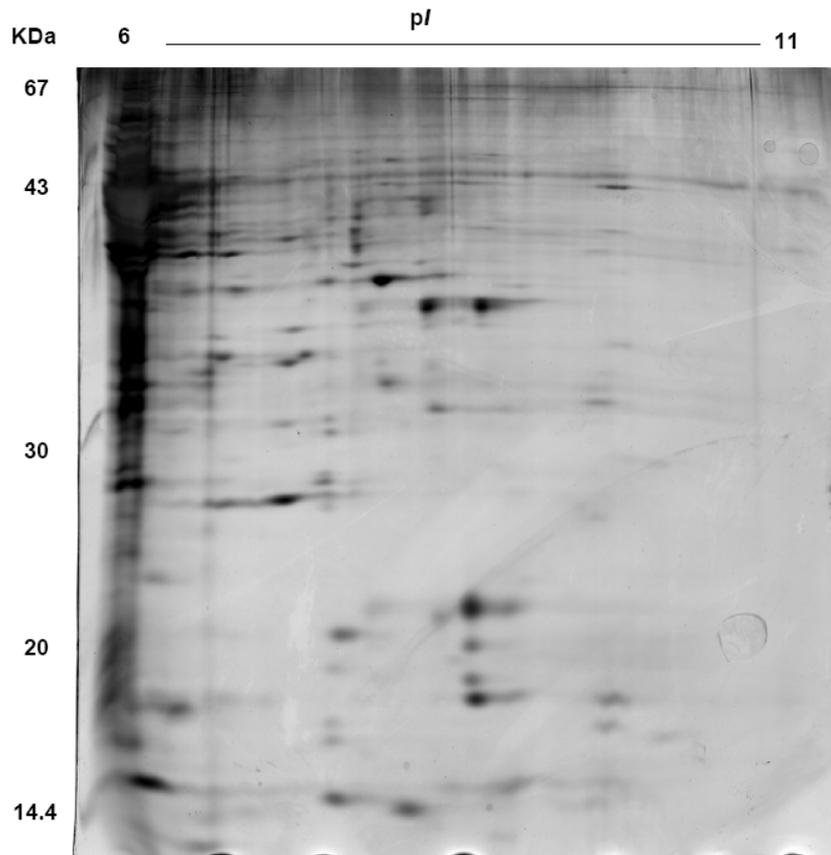
Nas análises de amostras de proteínas realizadas atualmente, são obtidas informações de diferentes características, algumas genéricas, como massa e pI. Outras, bastante específicas, como massas de conjuntos de peptídeos, composição e seqüência de aminoácidos e características dos reagentes utilizados nas pesquisas e da estrutura da proteína.

Os programas de identificação de proteínas por análise de suas características são muito específicos e recebem como insumos apenas determinados tipos de dados, normalmente referentes a apenas uma técnica de identificação. Diante dessa limitação, o desafio proposto para este projeto foi a elaboração de um sistema completo, que abordasse as informações obtidas das diferentes técnicas de identificação, analisasse as possíveis ferramentas disponíveis e consolidasse os resultados apresentados por essas ferramentas.

## **2.2. TÉCNICAS DE IDENTIFICAÇÃO DE PROTEÍNAS**

Atualmente são utilizadas diferentes técnicas de detecção e identificação de proteínas, cada uma observando determinadas características, isoladamente, da amostra analisada. Neste projeto, enfocamos três técnicas, descritas abaixo.

Antes da aplicação de uma técnica de identificação de proteínas, é necessário realizar a separação prévia de uma proteína presente em uma amostra, uma vez que a maioria das amostras é formada por misturas de proteínas. Para isso, podem ser utilizadas as técnicas de eletroforese 2D [8] (separação de proteínas de uma amostra por pI e, em seguida, por massa molecular) ou cromatografia (método de separação física em que os componentes passam por uma distribuição seletiva, promovendo a separação deles) [19]. Após a separação das proteínas da amostra, uma delas (ou uma mistura com poucos componentes) é selecionada, de acordo com o interesse da pesquisa. A seguir, um exemplo de gel-2D, apresentando características de pI e massa da amostra:



**Figura 2-5 Exemplo Eletroforese 2-D** Fonte: Dissertação de mestrado de Adriana Magalhães [20]

A primeira das técnicas aplicadas à identificação é a lista de massas dos peptídeos (PMF). Nessa técnica, as partículas de uma amostra são ionizadas e essas partículas carregadas são organizadas de acordo com suas massas [21]. Todo espectrômetro de massa é formado por três partes principais: fonte iônica, analisador de massas e um detector [22]. Estes equipamentos produzem um espectro dos peptídeos que constituem a proteína presente na amostra utilizada, cujos picos indicam a razão massa/carga, geralmente com resolução suficiente para permitir a diferenciação entre isótopos e entre formas multiplamente carregadas da mesma amostra. A diferença entre massas e a distância entre picos possibilitam a identificação de aminoácidos.

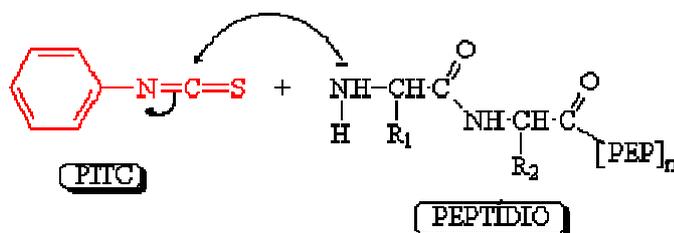
O conjunto das massas moleculares dos peptídeos, identificados pelo espectrômetro de massa, constitui a impressão digital da proteína (PMF – *Peptide Mass Fingerprint*). As informações de PMF podem ser utilizadas para identificar proteínas em bancos de dados, identificar falhas no processo de transcrição e também as modificações pós-traducionais [23].

Os instrumentos atuais são capazes de obter espectros de massa com precisão de 0.01Da ou melhores, porém, na identificação de proteínas, os erros são inevitáveis, podendo ser reduzidos. Existem inúmeras fontes de erros em experimentos laboratoriais, desde a forma de manipulação da amostra até o estado de conservação dos equipamentos utilizados. Também contribuem como fonte de erro a crescente quantidade de informações depositadas nos bancos de seqüências e, em muitos casos, sua inexatidão. Em um ambiente desse tipo, quanto maior a quantidade de informações utilizadas para identificação (fornecidas como fonte de busca), menor a chance de falha [24].

**Tabela 2-2 Exemplo de lista de massas**

M/z	Intensidade
718.398022	195.170000
1193.569765	288.430000
1204.616805	256.310000
1234.679047	563.930000
1320.567813	905.550000
1362.677240	1160.970000
1434.735636	977.560000
1440.677187	228.580000

A segunda técnica de identificação abordada por este projeto é o seqüenciamento da proteína, identificando os aminoácidos que a compõem e a ordem em que se encontram. A análise da composição dos peptídeos, determinando a sua seqüência, pode ser realizada pela técnica de Degradação de Edman, que pode ser feita de forma manual ou automática, utilizando equipamentos adequados [25]. Nesta técnica, é utilizada a derivatização N-terminal com PITC seguida por uma clivagem ácida que realiza a remoção dos aminoácidos por meio de interações químicas com a parte N-Terminal dos peptídeos. Os aminoácidos são removidos um a um, classificados por cromatografia e a seqüência linear pode ser determinada [2]. A seguir, um exemplo de reação durante a Degradação de Edman:



**Figura 2-6 Etapa durante o seqüenciamento por degradação de Edman** Fonte: [www.unb.br/cbsp/](http://www.unb.br/cbsp/) [26]



1) INTERFACE *WEB* (em que o usuário insere as informações que deseja utilizar para busca);

2) ALGORITMO DE PROCESSAMENTO (algoritmo característico do software de identificação, em que é feito o processamento das informações inseridas pelo usuário para realização da busca pelos resultados);

3) BUSCA NO BANCO DE DADOS SELECIONADO (nesta etapa é realizada a busca nos banco de dados selecionado pelo usuário ou pré-definido pelo software);

4) ANÁLISE ESTATÍSTICA DOS RESULTADOS (nesta etapa o software faz a análise estatística dos resultados encontrados, preparando-os para exibi-los ao usuário);

5) EXIBIÇÃO DOS RESULTADOS (esta etapa pode ser feita enviando os resultados por *e-mail* ao usuário ou exibindo-os na INTERFACE *WEB*).

Os programas que realizam identificação por *fingerprint* (neste projeto foi utilizado o Mascot) permitem ao usuário a busca de proteínas em bancos de dados desde que sejam fornecidos dados como: lista de massas de peptídeos (nas formas monoisotópica ou média), enzima utilizada para clivagem, modificações pós-traducionais e o banco de dados para busca, dentre outros. A implementação do algoritmo de cálculo do escore no Mascot incorpora o algoritmo de Mowse, descrito em [28], em que realiza a digestão teórica da seqüência das proteínas do banco e utiliza a lista de massas desta digestão para comparação com os dados do usuário. No programa Mascot, o escore é apresentado na forma  $-10 \cdot \text{Log}$  (escore de Mowse), sendo que quanto maior o escore Mascot, maior a chance de acerto no resultado. Outro programa que permite a identificação por meio desta técnica é o Phenyx. Este software gerencia pesquisas do usuário, analisa resultados e formata gráficos. Possui mais funcionalidades do que o Mascot, porém o acesso completo ao Phenyx é restrito aos usuários que compraram a licença de uso.

Os programas que realizam a identificação por comparação da seqüência da proteína (neste projeto foram utilizados o Blast e o Fasta) permitem ao usuário a identificação de proteínas com base em informações de seqüência, seja completa ou parcial, a partir das seguintes características da amostra: seqüência de aminoácidos (completa ou parcial), geralmente no formato FASTA e matriz utilizada no experimento. O Blast possui algumas características específicas, como: utilização da taxonomia completa do organismo e filtro por

regiões menos complexas. Para que o algoritmo do Blast seja utilizado, é necessária a existência de uma seqüência a ser pesquisada e um banco de seqüências para pesquisas. São feitas buscas por subseqüências da pesquisa semelhantes a subseqüências do banco de dados. Para o estabelecimento do score de alinhamento de seqüência, o Blast utiliza uma aproximação heurística dos resultados. Já o Fasta, permite a escolha do banco de dados de seqüência onde se deseja realizar a busca, seu algoritmo é dinâmico e rigoroso, apresentando as seqüências similares exatas.

O programa utilizado para identificação de proteínas por sua composição de aminoácidos, AACompident [29], permite a utilização de diversas constelações (conjunto de aminoácidos que podem estar presentes na composição), sendo que a constelação utilizada neste projeto é a constelação livre (permite utilizar todos os aminoácidos mais os duvidosos, como o ASX – arginina ou asparagina). Além da composição percentual dos aminoácidos, é possível informar os valores de pI e de massa molecular e, também, utilizar dados de taxonomia, uma proteína de calibração e sua composição de aminoácidos, o banco de dados a ser utilizado para pesquisa. O score calculado por este programa indica o grau de diferença entre a composição pesquisada e a composição das proteínas no banco de dados. O score é calculado, para cada uma das proteínas encontradas no banco, pela soma do quadrado da diferença entre a composição percentual de todos os aminoácidos da proteína pesquisada e das proteínas no banco de dados. Os resultados são ordenados do menor score (melhor resultado) para o maior (pior resultado).

Os programas Mascot, Blast, Fasta e AACompident foram escolhidos para utilização neste projeto por serem de uso consagrado entre os pesquisadores, vastamente descritos na literatura, terem acesso livre e, alguns deles (Blast e Fasta), apresentarem versões para execução local do programa também de uso livre.

Outra ferramenta para se localizar proteínas é o programa Multident [30] que possibilita a localização em bancos de dados de proteínas por meio de informações de ponto isoelétrico, peso molecular, composição de aminoácidos, taxonomia e outras informações mais gerais das amostras. Este programa foi utilizado para escolher as proteínas empregadas nos testes deste projeto.

## 3. CONCEITOS BÁSICOS EM COMPUTAÇÃO

### 3.1. BANCOS DE DADOS

Os primeiros bancos de dados surgiram para organizar, armazenar e disponibilizar dados, que consistiam em arquivos de papel ou arquivos texto de computadores acessados por diferentes aplicativos. Esses bancos ou bases de dados, após muitas evoluções, permitiram que conjuntos de registros fossem armazenados de forma estruturada, a facilitar a obtenção e reorganização dos mesmos e a produção de informação útil por meio dos sistemas de gerenciamento de bancos de dados.

Um sistema de gerenciamento de bancos de dados (SGBD) é um conjunto de programas de gerenciamento que acessa informações inter-relacionadas. Seu objetivo é permitir o armazenamento e a recuperação de conjuntos de dados [31]. Os SGBDs evitam que os dados sejam guardados em sistemas de armazenamento de arquivos, ou seja, arquivos texto espalhados e acessados por diferentes aplicativos. Essa forma aleatória de acesso às informações pode resultar em redundância e inconsistência de dados, dificuldade no acesso aos dados, isolamento de dados, anomalias de acesso concorrente, problemas graves de segurança e integridade.

A visão abstrata dos dados - em que os detalhes de como e onde os arquivos com os dados armazenados são omitidos - é possível com a utilização de SGBD. Esta visão pode ser acessada de três formas: nível físico, em que informações de baixo nível são descritas detalhadamente; nível conceitual, descreve quais dados estão armazenados no banco de dados e suas relações; nível de visões, que descreve apenas parte do banco de dados, pois muitos usuários não estão interessados no banco completo.

Um modelo de dados é a estrutura que descreve os dados, seus relacionamentos, a semântica e as restrições de consistência. Para o universo deste projeto, serão analisados os modelos lógicos relacionais (Modelo Entidade-Relacionamento) e os bancos de dados baseados em arquivos (*Flat File*). Os modelos relacionais, com sua origem na década de 70, foram um sucesso em razão de sua estrutura simples e uniforme (um banco relacional é composto por um conjunto de relações, com fundamentação teórica bastante sólida na matemática). Segundo E.F.Codd [32], este modelo descreve um banco de dados como uma coleção de relacionamentos entre valores que respeitam requisitos básicos de existência.

### 3.1.1. Principais formas de armazenamento de dados proteômicos e genômicos

#### Modelos de dados relacionais

O modelo de dados mais utilizado para armazenamento de dados genômicos e proteômicos é o relacional. Este modelo é baseado em princípios matemáticos, dentre os quais se destaca a Teoria de Conjuntos (os elementos se relacionam com os conjuntos da forma pertence ou não pertence). Nesse modelo, todos os dados são representados como relações matemáticas, possuindo dois possíveis predicados, verdadeiro ou falso. A linguagem padrão utilizada em bancos relacionais é a SQL (*Structured Query Language*).

A linguagem SQL foi desenvolvida na década de 70 pela *IBM* por um projeto que visava demonstrar a viabilidade da implementação do modelo relacional proposto por E.F. Codd [32]. Esta linguagem é vastamente utilizada por sua simplicidade e facilidade de uso. O padrão SQL foi determinado pela *American National Standards Institute* (ANSI) em 1986, pela norma ANSI SQL 87, e posteriormente pela *International Organization for Standardization* (ISO) com a norma ISO/IEC 9075. Atualmente, esta linguagem permite a utilização de expressões regulares, execuções de comandos recursivos e gatilhos na execução de consultas, inserções, remoções e atualizações de informações.

Neste modelo de dados, relacional, as entidades (ou tabelas de um banco de dados) são caracterizadas por um nome e seus atributos, comumente tratados por colunas. Essa estrutura armazena os dados do banco. Muitas vezes, para que os dados armazenados na base se transformem em informação útil, é necessário que sejam estabelecidos relacionamentos entre os atributos de diferentes dados, entre atributos de um mesmo dado ou entre atributos de um dado com algum valor externo de comparação.

Um relacionamento é caracterizado por uma associação entre atributos de diferentes entidades. A estrutura lógica deste modelo é expressa pelo Diagrama Entidade-Relacionamento. O diagrama E-R deste projeto é apresentado no **Apêndice B – Modelo de dados**.

#### Bases de dados em arquivos de texto

Esses bancos de dados mantêm suas informações registradas em arquivos puros de texto (*flat file*), um registro por linha. Nestes arquivos, os atributos do registro são separados por espaço em branco, ou vírgulas (gerando arquivos CSV) ou algum outro caractere delimitador definido. Nestes bancos de dados, não há relacionamentos, pois se trata de um simples arquivo de texto puro.

O formato freqüentemente utilizado para seqüências de proteínas é o FASTA. Este formato é padrão, devido à simplicidade da exibição da seqüência que qualquer programa de identificação de proteínas é capaz de processar. Ele pode ser iniciado com o nome da proteína, precedido de “>” e, em uma nova linha, a seqüência dos aminoácidos no formato de uma letra, segundo a IUPAC, em texto puro:

```
>gi|18203677|sp|Q9ZGE9|BCHN
MERVERENGCFHTFCPIASVAWLHRKIKDSFFLIVGTHTCAHFIQTALDVMVYAHSRFGFAVLEESDLVS
ASPTEELGKVVQVQVDEWHPKVIFVLSTCSVDILKMDLEVSCDKLSTRFGFPVLPASTSGIDRSFTQGED
AVLHALLPFVFPKEAPAVEPVVEKKPRWFSFGKESEKEKAEPARNLVLIGAVTDSTIQQLQWELKQLGLPK
VDVFPDGDIRKMPVINEQTVVVPLQPYLNDTLATIRRERRAKVLSTVFPVIGPDGTARFLEAICLEFGLDT
SRIKEKEAQAWRDLEPQLQILRGKKIMFLGDNLLELPLARFLTSCDVQVVEAGTPYIHSKDLQQELELLK
ERDVRIVESPDTFKQLQRMQEYKPDLLVAGLGICNPLEAMGFTTAWSI EFTFAQIHGFVNAIDLKLF TK
PLLKRQALMEHGWAEGWLE
```

### Figura 3-1 Exemplo de seqüência em formato FASTA

#### 3.1.2. Bancos de dados de proteínas

Existem bancos de dados biológicos implementados de acordo com os dois modelos de bancos apresentados anteriormente: modelo relacional e bases de dados em arquivos de texto. A utilização de arquivos texto é justificada pela ausência de administradores de banco de dados em alguns projetos, ou para permitir a compatibilidade com programas previamente desenvolvidos.

De acordo com seu conteúdo, os bancos de dados biológicos podem ser divididos em: primários, secundários e especializados [18]:

**Bancos primários** – contém os dados biológicos originais. São abastecidos de informações pela comunidade científica. Contêm uma quantidade mínima de anotações. Nos anos 80, os próprios cientistas inseriam os dados de suas pesquisas diretamente nos bancos. Atualmente, as publicações em jornais científicos exigem a prévia inserção das informações em um dos bancos, para garantir sua livre distribuição. Também é importante notar que atualmente os bancos de dados têm a inserção e anotação dos dados depositados feitas por

moderadores, de forma a validar e padronizar seu conteúdo, evitando os problemas decorrentes da descentralização que ocorria anteriormente.

Alguns dos grandes bancos de dados de seqüências atualmente utilizados nas pesquisas proteômicas são: Uniprot (este banco caracteriza-se como primário e secundário, por reunir diferentes bancos), NCBI e EMBL. Esses bancos são disponíveis, gratuitamente, na Internet, têm colaboração mútua e trocam informações diariamente. Porém, a forma com que os três exibem suas informações é diferente entre si.

Em outra via, as estruturas tridimensionais de macromoléculas são disponibilizadas em um banco que contém a grande maioria dos dados disponíveis, o PDB. Este banco é baseado em arquivo texto (flat file), contendo as coordenadas atômicas das macromoléculas (tanto proteínas quanto DNA).

**Bancos secundários** – para tornar as informações dos bancos primários utilizáveis para pesquisas, é necessário realizar um reprocessamento. Os bancos secundários armazenam estes dados reprocessados. A quantidade de informações resultantes varia bastante entre os bancos secundários de seqüência disponíveis. Alguns mantêm apenas informações das traduções de DNA, outros oferecem anotações e informações de alto nível sobre funções e estrutura da seqüência.

O banco de dados secundário de seqüência de proteínas, chamado Swiss-Prot oferece um elevado nível de anotações sobre aspectos importantes das proteínas armazenadas, tais como: descrição de suas funções, modificações pós-traducionais, estrutura e variantes. Suas informações são derivadas do EMBL. Recentemente, foram reunidas as informações do Swiss-Prot, TrEMBL e PIR, criando o banco UniProt, com uma cobertura enorme de informações de seqüências [33]. Por conter tanto dados primários, oriundos do EMBL, quanto dados secundários, do Swiss-Prot, o UniProt pode ser classificado em ambas categorias, banco de dados primário e secundário.

**Bancos especializados** – normalmente são bancos criados para pesquisas específicas. Suas seqüências são basicamente derivadas de bancos primários, porém com um elevado grau de anotações, podendo haver, inclusive, novas seqüências, uma vez que os cientistas envolvidos são dedicados a assuntos específicos [18].

Uma barreira constante nas tentativas de união de projetos de bancos de dados, principalmente os especializados, é a incompatibilidade de formatos, uma vez que eles podem ser arquivos texto, relacionais ou orientados a objetos. Uma saída utilizada, ultimamente, tem sido a aplicação de linguagens unificadas, como o XML, para a exibição de informações. Porém, nem todos os bancos disponibilizam esse tipo de resultado.

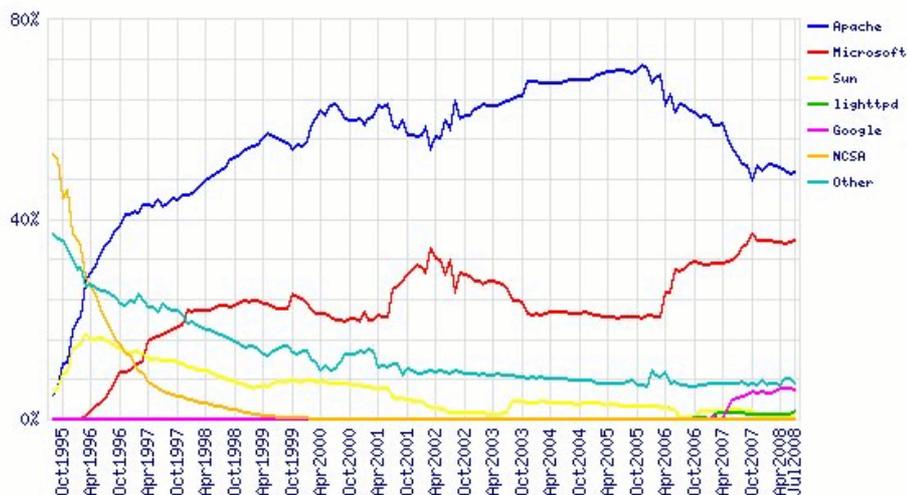
### **3.2. SERVIDORES *WEB***

Diversos bancos de dados de seqüências de proteínas são acessíveis por interfaces *Web*, bem como o sistema proposto por este projeto. Para que sejam viáveis, tais interfaces demandam a implementação de um servidor *web*.

Os servidores *web* utilizam o protocolo HTTP, disponibilizando conteúdo para as estações clientes. O protocolo HTTP é do tipo *request/response*, em que o cliente envia a consulta para o servidor, no formato adequado (contendo o protocolo, sua versão, mensagem MIME contendo informações de identificação do cliente e, se necessário, informações de autenticação do cliente). Este envia a resposta com uma mensagem contendo o protocolo e sua versão e aviso de sucesso ou falha na requisição, seguido de mensagem do tipo MIME com informações do servidor [34].

Esse aplicativo deve apresentar respostas bastante rápidas às requisições, desenvolver multitarefas, apresentar respostas aos possíveis erros e ser capaz de processar diferentes formatos de arquivo. Estas são algumas das características mais desejadas em um servidor *web* [35].

Em análises realizadas pelo instituto Netcraft [36], o Apache é apontado como servidor *web* mais utilizado em aplicações por todo o mundo, conforme pode ser visto na figura abaixo:



**Figura 3-2 Utilização de servidores web no mundo.** Fonte: [http://news.netcraft.com/archives/web\\_server\\_survey.html](http://news.netcraft.com/archives/web_server_survey.html)

Isto é devido à grande quantidade de características favoráveis, argumento defendido por Ben Laurie e Peter Laurie [35]. Algumas das vantagens do Apache, que foram decisivas na escolha como servidor para este projeto, são:

**Administração** – apresenta interface GUI para administração em sistemas *Windows*, e seus os comandos de administração (iniciar, parar, reiniciar) são facilmente executados via linha de comando em sistemas *Linux*;

**Portabilidade** – existem versões de Apache para várias opções de sistemas operacionais, incluindo toda a família *UNIX*, *Windows* e *MacOS*;

**Estabilidade** – por ter seu código aberto ao público, qualquer falha que afete as funcionalidades do sistema é rapidamente corrigida em uma nova versão, o que torna o sistema mais confiável;

**Suporte** – amplamente divulgado em livros e na internet, existem vários locais para buscas de ajuda.

### 3.3. LINGUAGEM PHP

O PHP é uma linguagem de *script*, com múltiplas funcionalidades, que é utilizada, principalmente, para desenvolvimento de páginas *web*, podendo ser embutida no código HTML, mesclada com outras linguagens, como *JavaScript* e XML e integrada a banco de dados por meio de funções que executam SQL [37].

A sintaxe do PHP é semelhante ao C e também oferece suporte a utilização de orientação a objetos. A sua documentação é totalmente disponível no site oficial do PHP ([www.php.net](http://www.php.net)), sendo, também, distribuída livremente em inúmeras fontes na internet.

Como o *script* pode ser embutido no HTML, não há necessidade de se executar o código PHP como um *script* CGI. Por esta característica, é possível fazer as correções do *script* com auxílio das mensagens de erro personalizadas, em vez da mensagem de erro geral disponibilizada nas aplicações CGI (“*500 Internal Server Error*”), bem como torna mais segura a administração dos *scripts* que não precisam de licença para execução no sistema operacional, que é exigido para aplicações CGI.

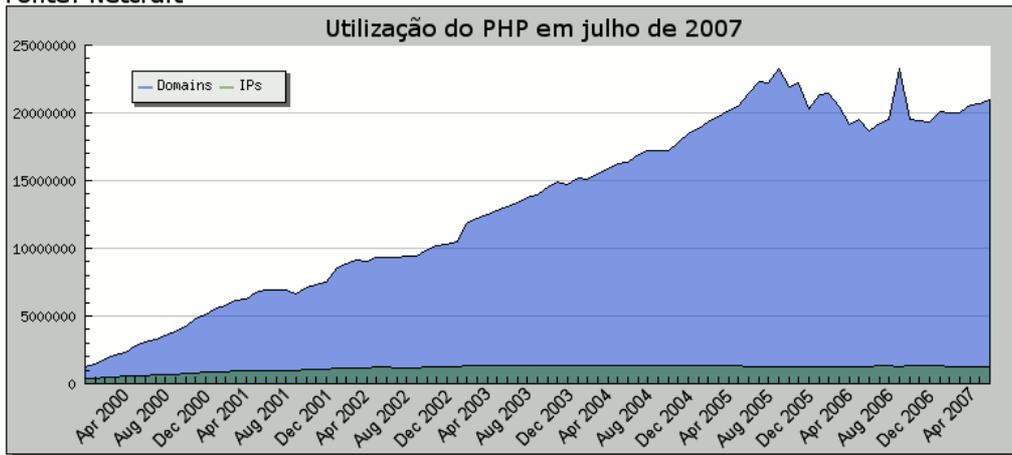
A função do PHP é tornar as páginas *web* dinâmicas, pois ele constrói o código HTML em cada um dos acessos ao *script*. Esta construção pode estar ligada a informações dinâmicas, como data e hora, informações de bancos de dados, informações randômicas, criptografadas, geração de imagens, manipulação de arquivos, entre outros conteúdos, que podem enriquecer uma página HTML.

Dois pontos fortes da linguagem, amplamente utilizados neste projeto, são a forma de se trabalhar com formulários e a comunicação facilitada com bancos de dados MySQL [38]. Além disso, pode ser totalmente embutida no HTML e não necessita ser executada como CGI. Por esses motivos, a linguagem PHP foi escolhida para ser utilizada no projeto.

A extensão do PHP para alguns comandos do Perl proporciona o tratamento de expressões regulares, favorece a recepção e o processamento das páginas de resultados, recebidas durante a execução do programa proposto neste projeto. Outros fatores importantes para a adoção do PHP foram a vasta documentação, disponível na Internet, para esta linguagem, a familiaridade da equipe com *scripts* PHP e a grande quantidade de servidores na *Internet* que utilizam esta linguagem, conforme pode ser visto abaixo:

PHP: 20.917.850 domínios e 1.224.183 endereços IP

Fonte: Netcraft



**Figura 3-3 Utilização do PHP nos servidores ao redor do mundo.** Fonte: [www.php.net/usage.php](http://www.php.net/usage.php) (com modificações)

Além do PHP, também foi necessário utilizar *Javascript*, principalmente nos trechos em que a melhor prática é a execução na máquina do usuário. Exemplos de uso do *Javascript* são o menu do site, as funções de confirmação de alguma atitude, como a confirmação antes da exclusão de um conjunto de dados, e o cálculo do resumo *hash* da senha do usuário.

### 3.4. ALGORITMO QFAST

Após a identificação de proteínas, efetuada por diferentes programas acessados pelo sistema, é necessário realizar uma análise estatística dos resultados obtidos para que seja feita a combinação desses resultados e apresentação de um valor consolidado para o usuário.

Foram avaliados dois métodos estatísticos de combinação de valores, o método de Fisher [39] e o algoritmo QFAST [40]. Ambos os métodos utilizam p-valores para a combinação de resultados. P-valor é um dado estatístico, percentual, de confiança no resultado obtido. O p-valor representa, portanto, a chance de que o resultado tenha sido obtido dentro do conjunto de resultados possíveis, ou seja, a probabilidade de que a amostra tenha sido retirada do espaço amostral considerado. Em um exemplo voltado à busca de proteínas em bancos de dados, se uma determinada seqüência for submetida à busca e for identificada uma proteína no banco com seqüência similar, com um determinado escore, o p-valor representará a probabilidade de um alinhamento aleatório ocorrer, com o escore em questão, ou melhor [41].

Os programas de identificação de proteínas utilizados pelo sistema Protein Locator apresentam em seus resultados o campo “e-valor (*expect value*)”. Este é definido como a probabilidade de que o resultado tenha sido tomado ao acaso no conjunto de seqüências de proteínas do banco de dados, ou seja, é a probabilidade do acaso ao se retirar uma amostra do espaço amostral ao acaso. Apenas o programa AACompident não apresenta tal valor, porém é possível calcular esta probabilidade a partir das informações apresentadas pelo programa.

Pelo fato de que ambos, p-valor e e-valor, tratam de uma medida de probabilidade que demonstra a confiança no resultado obtido [41], neste projeto o e-valor apresentado por cada um dos programas de identificação de proteínas foi utilizado como o fator de probabilidade do algoritmo de QFAST, apresentado a seguir, para o cálculo do resultado consolidado. Esta consideração baseou-se na limitação do e-valor entre zero (certeza absoluta de que o resultado não foi obtido ao acaso) e 1 (incerteza absoluta de que o resultado não foi obtido ao acaso).

O método de Fisher [39] utiliza operações de produto, logaritmo e a distribuição de qui-quadrado para, a partir de um conjunto de p-valores independentes, inferir qual o resultado da identificação de proteínas é o mais preciso. Por se tratar de valores muito pequenos, tendendo a zero, o custo computacional para a aplicação destas operações é muito elevado. Nos testes realizados durante o desenvolvimento do sistema, foi necessário utilizar um ambiente estatístico de programação, o ambiente R ([www.rproject.com](http://www.rproject.com)). Os resultados obtidos foram satisfatórios, porém o tempo para obtenção do resultado foi demasiado longo.

Uma alternativa a este método é o algoritmo QFAST [40]. Este algoritmo foi elaborado para se fazer a combinação dos p-valores obtidos em programas de similaridade de seqüências. Este algoritmo foi baseado no método de Fisher e na característica dos p-valores, quanto menor o p-valor, maior a probabilidade de acerto do resultado associado. Os p-valores podem ser combinados por meio de seu produto (probabilidade conjunta), que também será menor tanto quanto maior for a probabilidade de acerto do resultado combinado. Timothy L. Bailey e Michael Gribskov demonstraram, em seu artigo, a forma de calcular a distribuição do produto de variáveis aleatórias independentes e uniformemente distribuídas no intervalo entre 0 e 1, sem a utilização da distribuição de qui-quadrado (utilizada no método de Fisher). Dessa forma, a necessidade de processamento computacional é reduzida em cerca de 10 vezes em comparação com o método de Fisher, conforme constatado por Bailey e Gribskov. A equação deste algoritmo é a seguinte:

$$F_n(p) = p \sum_{i=0}^{n-1} \frac{(-\ln p)^i}{i!}$$

**Figura 3-4 Equação para combinação de p-valores**

A equação acima pode ser implementada pelo algoritmo QFAST, apresentado abaixo:

```

function qfast ( n: number of rv's,
  p: product of rv's)
if (p == 0) then return 0
if (n > 1) then x = -ln p
  t = p
  q = p
  for i = 1 to n - 1 by 1 do
    t = t × x/i
    q = q + t
  end
  return q
end

```

**Figura 3-5 Algoritmo QFAST**

No capítulo de Resultados e discussões, são apresentados comparativamente alguns resultados obtidos com o método de Fisher e o método QFAST e são feitas as devidas comparações.

## 4. REVISÃO BIBLIOGRÁFICA

A identificação de proteínas é um procedimento comum em bioquímica e vem sendo descrito em publicações há mais de cinquenta anos, tendo como marco inicial o seqüenciamento da cadeia de aminoácidos da insulina, em 1951 [42]. Nesse período foram desenvolvidos bancos de dados, conforme descrito anteriormente, dentre eles o Swiss-Prot e o UniProt.

O Swiss-Prot foi desenvolvido por Amos Bairoch e sua primeira versão foi disponibilizada em 21 de Julho de 1986 [43]. Trata-se de um banco de dados curado (com anotações das proteínas) de seqüências de proteínas, com um elevado nível de anotações (como a descrição das funções das proteínas, modificações pós-traducionais, variantes, etc.), um baixo nível de redundâncias (as diferentes citações da mesma proteína são juntados em ocorrências únicas) e um elevado nível de integração com outros bancos de dados. É mantido pelo *Swiss Institute of Bioinformatics* (SIB) e pelo *EMBL Data Library* [44].

O UniProt (*Universal Protein Resource*) [45] é uma proposta de centralização das informações de seqüência de proteínas [44]. Seu objetivo é oferecer seqüências com anotações, baixa redundância de informações e alta velocidade nas buscas. Para isso, promove a união dos dados do PIR (*Protein Information Resource*) [46], SIB (*Swiss Institute of Bioinformatics*) [47] e EBI (*European Bioinformatics Institute*) [48]. O UniProt engloba três bancos de dados diferentes: UniProtKB – uma base de proteínas com dados do Swiss-Prot (banco que possui anotações manuais em seus registros) e dados do TrEMBL (banco com anotações automáticas) –, o UniRef – banco de dados de seqüências organizado em um *cluster* – e o UniParc – banco de dados não redundante de seqüências de proteína, reúne informações de vários outros bancos, dentre eles o PDB, o EMBL e o UniProtKB [33].

Uma das ferramentas mais rápidas e utilizadas para comparação de seqüências de proteínas e bases de nucleotídeos é o Blast (*Basic Local Alignment Search Tool*) [49], publicado em 1990. Seu algoritmo utiliza análise heurística para realizar alinhamentos locais. O programa seleciona uma subsequência, sem espaços em branco ou dúvidas (GAP), realiza uma busca no banco de seqüência pelo melhor resultado de alinhamento com a subsequência. Depois de encontrada a subsequência, é realizada uma extensão para os lados para verificar se o alinhamento continua correto e por consequência ocorre melhora na classificação do resultado. Essa extensão do alinhamento é realizada até que seja encontrado um GAP. São

atribuídos escores para os alinhamentos. O algoritmo proposto por Altschul e colaboradores é bastante veloz, devido a utilização de subsequências para formação de pares e a posterior extensão das mesmas para melhor classificação.

No programa Blast, o cálculo do escore é realizado em três etapas. Primeiro são classificados, por matriz de substituição, trechos de tamanho definido por:  $L - w + 1$ , onde  $L$  é o tamanho da sequência que se deseja localizar e  $w$  é, geralmente, 3 para proteínas. Em seguida, são realizadas buscas no banco de dados por sequências homólogas às obtidas na primeira etapa. Por fim, para cada uma das subsequências que foram identificadas no banco de dados, é realizada uma extensão, para ambos os sentidos da sequência, e efetuada nova busca no banco de dados, a fim de se aumentar o escore de similaridade [50].

Para cada alinhamento encontrado, a ferramenta utiliza análise estatística para produzir um “*bit score*” e um “*expect value* (e-valor)” correspondente. O e-valor de cada alinhamento traz a sua indicação da significância estatística e reflete o tamanho do banco de dados utilizado ( $M \times N$ ) e o sistema de escore. Quanto menor este indicador estatístico, mais significativo é o *bit-score*. O cálculo dos parâmetros estatísticos para os alinhamentos é demonstrado por Altschul [51].

Outro programa que realiza identificação de proteínas por meio de busca por sequências é o Fasta, descrito pela primeira vez em 1985 por Lipman e Pearson [52]. Este programa, assim como o Blast, realiza busca de alinhamentos locais. Seu algoritmo também realiza a comparação das sequências por meio de matrizes de substituição. Além disso, nem todo o conteúdo inserido para a busca é utilizado, pois é efetuada uma análise prévia para acelerar o programa, escolhendo apenas as regiões com maior escore, segundo a matriz de substituição.

O cálculo do escore das buscas que caracterizam as sequências similares com sucesso é realizado em quatro etapas pelo programa Fasta. Estas etapas são descritas por Barton (1996) [53]: na primeira etapa, são localizadas regiões com identidades (alinhamento local); em seguida, é utilizada a matriz de substituição adequada para eleger os melhores escores das identidades, que são mantidos pelo programa; depois disso, é realizada a separação das identidades que estejam dentro de um limite de proximidade da classificada com maior escore; por fim, são utilizadas técnicas de computação para alinhar os segmentos eleitos na etapa anterior.

As matrizes de substituição são utilizadas para se obter o escore de alinhamento de cada um dos possíveis pares de resíduos de aminoácidos, por meio de uma matriz de probabilidade de troca de um aminoácido por outro. Com o passar dos anos, várias matrizes de substituição foram propostas [54].

Dayhoff e colaboradores descreveram um modelo, baseado no modelo de Markov, chamado de matriz PAM (*Point Accepted Mutation*). Esta matriz apresenta valores de probabilidade de substituição entre dois aminoácidos no processo de identificação da proteína por sua seqüência. A construção desta matriz baseia-se nas mudanças ocorridas durante a evolução das proteínas e suas versões (por exemplo PAM30, PAM50, PAM120 e PAM250) referenciam à sensibilidade que se deseja utilizar para as substituições, pois PAM30, por exemplo, significa 30 substituições a cada 100 resíduos [55]. Algumas outras matrizes descritas, KMH, Paml, Proml, Molphy, DCMut e DCFreq, são variações do modelo proposto por Dayhoff [56].

Outra importante matriz de substituição foi descrita por S. Henikoff e J.G. Henikoff [57]. Chamada de BLOSUM (*Blocks Substitution Matrix*), esta matriz é mais utilizada para alinhamentos locais de seqüência, sendo que é utilizada por padrão pelo Blast.

Além destas, existem outras matrizes que são menos utilizadas pelos programas de identificação, como, por exemplo, a matriz descrita por Gonnet e colaboradores [58] e a JTT, descrita por Jones, Taylor e Thornton [59].

O programa Mascot, de propriedade da MatrixScience [60], utiliza dados de espectrometria de massa para realizar a identificação de proteínas. Suas buscas são realizadas em bancos de dados de seqüência de proteínas[61]. Essa ferramenta abrange buscas por três diferentes métodos [62]:

- *Peptide Mass Fingerprint* (PMF) – utiliza valores de massas de peptídeos e opcionalmente a intensidade do sinal gerado por cada peptídeo. Este é o método utilizado por este projeto.

- *Sequence query* – combina massas de peptídeos com trechos de suas seqüências de aminoácidos.

- MS/MS Ion search* – utiliza dados de experimentos de MS/MS (fragmentação em experimentos de espectrometria de massa) ainda não interpretados.

A análise por PMF é feita a partir de uma amostra da proteína, digerida por uma enzima proteolítica, geralmente a tripsina, cuja mistura de peptídeos produzida é submetida a um espectrômetro de massa para análise. São obtidos, então, valores das massas moleculares de diversos peptídeos que compunham inicialmente a proteína.

Os valores experimentais são, então, comparados aos valores de massas de peptídeos calculados a partir da digestão teórica das proteínas armazenadas em bancos de seqüência, como o Swiss-Prot. O Mascot aplica um algoritmo de cálculo de escore para as semelhanças encontradas entre os peptídeos experimentais e as seqüências armazenadas no banco de dados.

O cálculo do escore no Mascot é baseado na implementação do algoritmo de Mowse, que é completamente descrito em Pappin, 1993 [28]. A primeira fase de uma busca, utilizando o algoritmo de Mowse, compara a massa calculada do peptídeo de cada entrada no banco de dados de seqüências com a massa indicada experimentalmente. Cada valor teórico que condiz com a massa experimental, dada a tolerância de massa, é contado como um acerto. A tolerância de massa pode ser utilizada como um pré-filtro para a busca [63].

O Mascot utiliza o algoritmo de Mowse aliado à análise estatística dos resultados. Os valores de massa que configuram acertos são utilizados em uma base estatística. O escore total do acerto é igual à probabilidade deste acerto ter sido tomado ao acaso. Porém, para evitar confusões de interpretação, é exibido um escore calculado por  $10 \cdot \log(P)$ , sendo que P é o escore real [64].

Para identificação por dados de composição de aminoácidos, este projeto utilizou o programa AACompident [29]. O banco de dados de seqüências utilizado por esta ferramenta é o SwissProt/TrEMBL. Para preenchimento do formulário, o usuário deverá informar a composição em percentual molar para cada aminoácido, também poderá informar a composição da proteína de calibração, caso esta seja utilizada. O algoritmo de cálculo de escore desta ferramenta é bem diferente das demais até então analisadas. Quanto menor o escore, mais semelhante é a amostra experimental ao peptídeo [14].

A GeneBio disponibiliza o software Phenyx [65], um programa para identificação e caracterização de proteínas e peptídeos a partir de dados de espectrometria de massa. Ele foi produzido para atender as crescentes demandas por análise de dados de espectrometria de massa [66].

O software foi desenvolvido pela equipe da GeneBio em colaboração com o Instituto Suíço de Bioinformática (SIB). Ele utiliza um sistema probabilístico de escore chamado de OLAV. Este algoritmo baseia-se na teoria de detecção de sinais, explorando bem as características da espectrometria de massa. Para diminuir a ocorrência de falsos positivos, este algoritmo realiza uma análise estrutural das informações obtidas pelo espectrômetro, diminuindo a necessidade de verificação manual das proteínas identificadas dentro do contexto avaliado (levando-se em conta organismo em que foi obtida a amostra, condições de massa e pI, e outras informações estruturais) [67].

O Phenyx possibilita que o usuário submeta dados para identificação, visualize e avalie os resultados, de várias formas, manualmente valide e compare os resultados e os exporte em formatos integrados com o Phenyx. A interface com o usuário é feita pelo navegador *web* [68].

Este programa possui uma série de possibilidades na área de gerenciamento dos dados. Inclui filtros específicos para listas de massa, conexão com o espectrômetro, e uma série de outras utilidades. Porém, sua abrangência se restringe aos dados de espectrometria e é necessário comprar a licença para utilizar o produto, tanto pela interface *web* quanto para instalar uma cópia local do programa, fatos pelos quais esse programa não foi utilizado neste projeto.

Uma equipe de Bioinformática da Fiocruz, composta por Marcos Catanho e colaboradores, desenvolveu a ferramenta chamada de BioParser [69]. Este programa facilita a visualização dos resultados do Blast e do Fasta que, nos casos de buscas muito extensas, são pouco práticos para visualização humana.

O software desenvolvido na Fiocruz, é uma ferramenta em Perl, que utiliza o pacote BioPerl [70]. O BioParser, que utiliza o MySQL como SGBD, permite o *parsing* dos resultados de variadas opções do Blast e do Fasta, facilitando a visualização dos parâmetros escolhidos pelo cientista.

Para o *parsing* dos resultados, é necessário que o cientista proceda a busca e grave a página *web* com os resultados em seu computador. Após o armazenamento dos resultados, deverá ser acessada a ferramenta BioParser e feito o carregamento dos arquivos armazenados para a ferramenta, que possui uma interface GUI.

Portanto, esse programa não contempla a execução das buscas, utilização de outras técnicas para identificação, além do seqüenciamento, e nem a consolidação dos resultados para o cientista.

O algoritmo QFAST [40], elaborado por Bailey e Gribskov, é um algoritmo simples e rápido para o cálculo da distribuição do produto de variáveis aleatórias independentes e uniformemente distribuídas no intervalo (0,1). Segundo os autores, uma importante aplicação deste algoritmo é a combinação de dados biológicos de similaridade de DNA e proteínas. Este algoritmo de combinação de p-valores é utilizado neste projeto para o cálculo do e-valor e do escore consolidado das proteínas identificadas, por ser o único algoritmo encontrado na bibliografia com tal funcionalidade.

## 5. METODOLOGIA

### 5.1. METODOLOGIA DE DESENVOLVIMENTO DO SISTEMA

#### 5.1.1. Visão geral

A indústria de software, na busca por maior qualidade em menor tempo, utiliza metodologias para o desenvolvimento, adotando métricas e padrões para alcançar níveis de qualidade e prever custos e prazos nos projetos.

Um software com qualidade deve ser produzido, focando os requisitos dos clientes, como segurança e desempenho, e dos desenvolvedores, como o custo no desenvolvimento e a forma de trabalho.

A elaboração de testes durante o desenvolvimento, com diferentes níveis de aceitação, garante aos clientes a certeza da implementação de funcionalidades e ao desenvolvedor a garantia de poder passar para uma nova etapa no desenvolvimento [71].

Neste projeto, a metodologia adotada para o desenvolvimento do sistema foi a Programação Extrema (XP - *Extreme Programming*). A XP é uma metodologia ágil de desenvolvimento de software. Ela é voltada para projetos em que os requisitos mudam com frequência, possui equipe pequena de desenvolvedores e o desenvolvimento é de forma incremental, ou seja, o sistema é implementado logo no início do projeto e vai ganhando novas funcionalidades com o passar do tempo [72].

Para que um processo de desenvolvimento XP possa fluir normalmente, é imprescindível uma comunicação constante entre equipe de desenvolvimento e cliente, que pode reavaliar suas necessidades enquanto utiliza fases desenvolvidas do software. Para que isso beneficie também os desenvolvedores, estes devem programar preocupados com os problemas atuais, sem previsões de possíveis problemas, que provavelmente serão alterados pelas necessidades do cliente.

Dessa forma, o desenvolvimento e a documentação do software foram feitos conjuntamente. Primeiro era desenvolvida uma etapa, de acordo com as necessidades do projeto. Em seguida, era realizado um teste de aceitação dessa funcionalidade, que era refeita caso não atendesse aos interesses do projeto. Após a aprovação da etapa, esta era documentada.

### 5.1.2. Desenvolvimento do software

Neste projeto, primeiramente foram levantados os casos de uso, que, na medida em que eram implantados, foram documentados. No decorrer do desenvolvimento, outros casos de uso apareceram e alguns foram extintos. A documentação completa dos casos de uso encontra-se no Apêndice A – Especificação Funcional do Software.

A priorização dos casos de uso foi feita considerando fatores como:

- Dependência entre o caso de uso e os seguintes;
- Complexidade do desenvolvimento, levando-se em conta a equipe de desenvolvimento;
- Risco da funcionalidade, ou seja, a importância do sistema não desempenhar o papel previsto caso a etapa desenvolvida não funcionasse.

Para direcionar o desenvolvimento, foi elaborada uma tabela de priorização de casos de uso, de acordo com os fatores listados acima. Nesta tabela, levou-se em conta que quanto maior a dependência das próximas etapas, mais importante o caso de uso, bem como, quanto maior o risco, maior a prioridade da etapa. No quesito complexidade, a relação foi invertida, ou seja, quanto maior a complexidade, maior o tempo gasto pelo desenvolvedor para a codificação da etapa, e, portanto, menor a prioridade da etapa. A escala utilizada foi:

- Dependência: alta (3), média (2) e baixa (1);
- Risco: alto (3), médio (2) e baixo (1);
- Complexidade: alta (1), média (2) e baixa (3);

A métrica utilizada para atribuição dos pontos dessa escala foi o bom-senso do desenvolvedor e do orientador do projeto, tendo em vista a equipe reduzida. A pontuação dos fatores foi levada em consideração apenas nos casos de uso que, cronologicamente, deveriam ser desenvolvidos ao mesmo tempo. Nestes casos, quanto maior a pontuação, maior a prioridade do caso de uso. No restante dos casos, serviu como parâmetro para estimar o tempo decorrido e o tempo necessário para terminar cada fase.

Seguindo estes preceitos, a tabela de priorização utilizada neste projeto foi a seguinte:

**Tabela 5-1 Tabela de priorização das atividades**

Nome do caso de uso	Dependência	Risco	Complexidade	Pontuação
Criar novo usuário	Alta	Alto	Baixa	7
Efetuar <i>Login</i> no sistema	Alta	Alto	Média	8
Visualizar pesquisas do Usuário	Média	Médio	Alta	5
Criar Pesquisa	Alta	Médio	Baixa	8
Visualizar Detalhamento da Pesquisa	Média	Alto	Alta	6
Modificar pesquisa *	Baixa	Médio	Média	5
Remover Pesquisa	Baixa	Baixo	Baixa	5
Adicionar composição de aminoácidos	Média	Alto	Média	7
Modificar composição de aminoácidos*	Baixa	Médio	Média	5
Remover composição de aminoácidos	Baixa	Baixo	Baixa	5
Adicionar <i>fingerprint</i>	Média	Alto	Alta	6
Modificar <i>fingerprint</i> *	Baixa	Médio	Alta	4
Remover <i>fingerprint</i>	Baixa	Baixo	Média	4
Adicionar <i>sequence data</i>	Média	Alto	Alta	6
Modificar <i>sequence data</i> *	Baixa	Médio	Alto	4
Remover <i>sequence data</i>	Baixa	Baixo	Média	4
Avaliar possíveis buscas de uma pesquisa**	Alta	Alto	Média	8
Submeter Pesquisa para serviço de busca Proteômica	Alta	Alto	Alta	7
Receber resposta de pesquisa via <i>WEB</i>	Alta	Alto	Alta	7
Receber resposta de pesquisa via <i>E-MAIL</i> ***	Alta	Alto	Alta	7
Exibir Resultados consolidados	Média	Alto	Alta	6
Exibir Resultados originais*	Alta	Alto	Baixa	9

\* - os casos de uso de modificação de dados e exibição dos resultados originais

---

foram incluídos depois do início do projeto, no lugar de reprocessamento automático de dados.

\*\* - o caso de uso de avaliar possíveis buscas de uma pesquisa foi adicionado após a realização de alguns testes do sistema.

\*\*\* - o caso de uso foi adicionado após o início do projeto, pois o programa AACompident só disponibiliza resultados via *e-mail*.

---

Levando-se em conta a tabela de priorização, o primeiro passo no desenvolvimento foi o Caso de Uso (CDU) “Criação de Novo Usuário”. Esta etapa é indispensável para todo o sistema, uma vez que todo o banco de dados está relacionado ao usuário, que pode inserir os dados, apagar, modificar, fazer buscas e visualizar resultados apenas das pesquisas que estiverem relacionadas ao seu usuário no banco de dados. Para garantir essa unicidade de relações, foi definido que o *login* do usuário seria seu *e-mail* e que esse *login* é único, ou seja, não pode haver mais de um usuário com o mesmo *e-mail* cadastrado no sistema. As regras de negócio estão descritas no **Apêndice A – Especificação funcional**.

Em seguida foi desenvolvida a etapa de *login* e *logout* no sistema. Esta etapa também é crucial, pois é estabelecida uma sessão com o usuário, utilizada para relacionar suas atividades ao seu *login*. Outra preocupação dessa etapa foi a construção de um sistema seguro de *logout*, em que todas as variáveis de sessão são removidas, impedindo que seja retornada a página do navegador para as informações do usuário.

Neste ponto, após o *login*, houve a necessidade de eleger o próximo caso de uso para codificação, uma vez que cronologicamente, o usuário poderia gerenciar suas pesquisas ou criar uma nova pesquisa. A tabela de priorização auxiliou na definição de que primeiramente deveria ser construído o módulo de criação de nova pesquisa. Depois de construído esse módulo, foi feita a etapa de gerenciamento das pesquisas do usuário.

Mais uma vez, a etapa seguinte foi definida com o auxílio da tabela de priorização, que privilegiou a construção dos módulos de inserção de dados de uma pesquisa. Foram construídos os módulos de inserção composição de aminoácidos, dados de *fingerprnt* e dados de seqüência de proteína. Depois disso, foi construído o módulo de visualização detalhada dos dados de uma pesquisa. Em seguida, foram feitos os módulos de remoção dos dados da pesquisa e de remoção da pesquisa completa, devido à baixa complexidade destes módulos.

Levando-se em conta o elevado risco envolvido com a submissão de dados para os serviços de identificação de proteínas, essa foi a etapa eleita para a continuidade do projeto. Essa etapa envolveu profundas pesquisas das ferramentas de identificação de proteínas e dos métodos HTTP para automatização de formulários e pode ser considerada a etapa fundamental do projeto, uma vez que é parte do objetivo deste a utilização de diferentes programas de identificação de proteínas. Para conclusão desse módulo, foi necessário dividir o sistema de submissão automática em vários robôs, um para cada programa de identificação, devido a especificidade de informações necessárias para essas ferramentas. Para a construção de cada um dos robôs, foi necessário primeiramente obter o formulário de submissão de dados dos programas de identificação de proteínas. Estes formulários são preenchidos automaticamente, com as informações de pesquisas cadastradas pelos usuários no banco de dados do sistema, e submetidos para os *sites* dos programas. Os resultados são analisados e as suas informações armazenadas no banco de dados. Foram necessários conhecimentos de estabelecimento de sessões com *cookies* (utilizados por alguns *sites* para identificar o usuário), utilização do módulo *libcurl* com PHP (biblioteca do programa *cURL* para uso com o PHP, permitindo transferência informações via sintaxe URL, como dados de formulário e *cookies*) e conhecimentos básicos de *Shell script* (imprescindível para se trabalhar com sistemas Unix), para a conclusão deste caso de uso.

Após a conclusão da etapa de submissão automática de dados, foi definido que seria necessário também construir a ferramenta de recebimento de resultados via *web* juntamente com a submissão. Dessa forma, o mesmo robô que faz a submissão para os programas de identificação (exceto para o AACompident) realiza o recebimento e análise dos resultados. Essa etapa também foi de extrema importância para o projeto, sendo um objetivo do projeto permitir ao usuário utilizar resultados de diferentes programas de identificação de proteínas, porém os resultados são obtidos no *background* do sistema operacional, não sendo visíveis diretamente para o usuário. Dessa etapa dependiam as etapas posteriores de “exibir resultados originais dos programas” e “exibir resultados consolidados”. Para permitir a obtenção de determinados dados das páginas de resultados, foram elaboradas expressões regulares, que retiram da página original de resultados todos os dados necessários para análise de dados e mais alguns que poderão ser utilizados em projetos futuros de melhoria deste sistema.

Depois de concluídas essas duas etapas, de submissão automática e obtenção de resultados via *web*, que consumiram bastante tempo e dedicação do desenvolvedor e do

orientador, o escopo original do sistema foi reduzido. Originalmente, pretendia-se construir mais módulos, de pré-processamento e de re-submissão automática de dados. Porém, preferiu-se alterar essas etapas para permitir que o usuário modificasse os dados de suas pesquisas e fizesse várias submissões com conjuntos diferentes de dados. Dessa forma, a tomada de decisões para a melhoria dos resultados da identificação fica a cargo do usuário. Para isso, foram adicionados os casos de uso de modificação de pesquisa, composição de aminoácidos, dados de *fingerprint* e dados de seqüência de proteína.

Após a realização de alguns testes, foi observada a necessidade de se fazer uma análise prévia dos dados do usuário, indicando para ele quais os programas de identificação podem ser utilizados com quais conjuntos de dados. Essa necessidade da avaliação prévia deve-se ao fato de o PL ser bastante abrangente. Assim, os formulários do sistema permitem a inclusão de valores que não são compatíveis com todos os programas de identificação de proteínas que existem atualmente, porém, em um futuro próximo, poderão ser utilizados em novos programas que surgirem.

Mais uma vez auxiliado pela tabela de priorização, o próximo passo foi a exibição dos resultados originais das buscas. Para armazenar os resultados originais dos programas de identificação de proteínas, foi necessário incluir um campo `LONGBLOB` na tabela de resultados do banco de dados. Esse caso de uso também foi incluído após o início do projeto para permitir ao usuário a tomada de decisão para novas submissões, assim como os casos de uso de modificação dos dados já armazenados no banco de dados.

O próximo passo, de acordo com a tabela de priorização, foi a construção do robô de recebimento de resultados por *e-mail*, etapa de maior complexidade. Para tanto, foi utilizado um *socket* para estabelecimento de conexão entre o software e sua caixa de *e-mail*, hospedada em outro servidor. Para o estabelecimento desta conexão, foi necessário utilizar pacotes disponibilizados pelo projeto PEAR (*PHP Extension and Application Repository*) [73]. Estes pacotes possibilitam a criação do *socket* de conexão e a execução de ações como leitura, remoção, marcação e outras possíveis em um sistema de correio eletrônico via *web*. Foi utilizado um servidor externo de *e-mail*, pois o custo de configurar um servidor próprio seria demasiado alto. Este mesmo robô faz o recebimento do resultado, armazenamento do arquivo original de resultados e separação dos dados utilizados pelo software para identificação automática.

Por fim, foi implementado o caso de uso de consolidação de resultados. Essa foi uma etapa dificultada pela precisão dos resultados, que algumas vezes apresentam probabilidade de erro (e-valor) menor do que  $1E-100$ . Para o tratamento estatístico dos resultados, foi utilizado primeiramente o algoritmo já consolidado de FISHER [39]. Um dos cálculos deste algoritmo é a distribuição de qui-quadrado. Com as dificuldades encontradas neste CDU, precisão melhor do que  $1E-100$  e o elevado tempo para o cálculo da distribuição de qui-quadrado, foram necessárias novas buscas por algoritmos para combinação de informações estatísticas. O que melhor se aplicou foi o QFAST [40].

Porém, após alguns testes com o cálculo do e-valor consolidado pelo algoritmo, foi observada uma falha. Os programas que utilizam método heurístico de análise estatística para a identificação de proteínas, muitas vezes retornam resultados não confiáveis, em que várias proteínas são classificadas com probabilidade nula de erro (e-valor igual a zero). Como o algoritmo de combinação de p-valores utiliza o produto dos p-valores, o resultado consolidado apresenta várias proteínas com e-valor igual zero, independente do resultado apresentado por outros programas que não fazem tratamento heurístico de dados. Para corrigir essa falha, a solução apresentada pelo projeto foi a criação de um novo parâmetro de classificação dos resultados para manter a identificação mais precisa possível: é adicionado um valor infinitesimal em cada e-valor antes do cálculo do produto. Desta forma, a presença da proteína no conjunto de resultados de mais de um programa é levada em consideração para o cálculo do score de classificação, o PL score. Assim como os e-valores, quanto menor o PL score, maior a probabilidade de acerto da proteína.

Após a escolha de quais dados utilizar e a quais programas submetê-los, foram elaborados os *scripts* de submissão automática de dados para os programas. Essa etapa é feita com muito cuidado, pois qualquer falha durante a submissão, que envolve várias informações, pode resultar em um falso resultado, prejudicando todo o sistema, uma vez que o usuário não tem acesso ao processo de submissão automática de dados. A fim de manter a forma silenciosa de submissão, foram construídos robôs que fazem requisições HTTP, preenchem formulários e submetem as buscas, executando no plano de fundo do sistema. Após essa etapa, os robôs foram melhorados para fazerem a requisição, preencherem o formulário, receberem o resultado e fazerem a escolha dos resultados mais corretos e os dados necessários para essa avaliação.

Por fim, o programa faz a análise estatística dos resultados encontrados nas buscas realizadas. Para a realização desta análise, foi utilizado o algoritmo QFAST, de implementação bastante simples. Para a utilização do algoritmo, é necessário realizar buscas no banco de dados para encontrar os resultados associados à pesquisa do usuário. Este *script* utiliza apenas os resultados tidos como aceitáveis (e-valor menor que 1, conforme explicado anteriormente). Então, calcula-se a combinação destes resultados.

Ao usuário, é apresentado o valor final da distribuição, a opção de visualizar todas as probabilidades de erros dos programas e também a possibilidade de visualizar o resultado original retornado pelos programas de identificação.

## 5.2. ADICIONANDO SERVIÇOS AO PROGRAMA

Para implementação do sistema, foram eleitos os programas, já descritos anteriormente: Blast, Fasta, AACompident e Mascot. Porém, o sistema não se limita a utilizar apenas os resultados provenientes destas ferramentas. Com o avanço da Bioinformática, novos programas podem surgir bem como novas versões destes mesmos programas. Para se adaptar a essas mudanças, é possível adicionar novos serviços de identificação de proteínas no Protein Locator. Alguns programas candidatos a serem adicionados são o DeepMind [74] que permite identificação por PMF e o MultiIdent [75] que permite utilizar dados estruturais da proteína, de composição de aminoácidos, de *sequence tag* e de PMF.

Além da adição de novos programas, podem ser incluídos novos formulários para que o usuário cadastre outros tipos de dados no sistema, como informações de *sequence tag*. Para realizar estas mudanças, são necessárias alterações no banco de dados (adicionando tabelas para armazenar os dados e os resultados do programa), alterações e criação de novos formulários HTML para inserção e edição de novos tipos de dados, alteração das páginas de exibição dos resultados consolidados (HTML e *script* de cálculo do resultado consolidado).

Para se adicionar novos serviços ao programa, primeiramente deve ser avaliado que tipo de informação será necessária para que sejam efetuadas buscas por este serviço. Se o programa for um adicional aos serviços já existentes (composição de aminoácidos, seqüência de proteínas ou *fingerprint*), a primeira etapa é acrescentar o serviço ao banco de dados, na tabela “services”. Senão, deverá ser elaborado um formulário com, no mínimo, os dados obrigatórios para submissão de uma busca no novo programa e a respectiva tabela no banco

de dados, seguindo o modelo de relacionamentos já utilizado no banco de dados. A partir destas informações, deve ser avaliada a necessidade de se criar uma nova tabela para armazenar os dados experimentais para o novo serviço de identificação, bem como a tabela para armazenar os resultados das buscas por proteínas. Depois disso, verificar, e fazer as modificações necessárias, se os campos apresentados no formulário do serviço são suficientes para se submeter uma busca ao novo programa, se não forem, devem ser feitas as modificações necessárias.

Após a etapa de armazenamento de dados para o novo programa, será necessário construir o conjunto de regras de negócio que permitem a submissão dos dados ao serviço. Isto é estabelecido na etapa de visualização detalhada da pesquisa. Após isso, será necessária a construção do robô para submissão e recepção automática de resultados para o novo programa.

É necessário utilizar um robô para cada programa de identificação, devido a sua especificidade. Para a construção de um robô de submissão, primeiramente é necessário que se identifiquem todos os campos do formulário. Em seguida, devem ser estabelecidas as regras de negócio, que impeçam a tentativa de submissão de dados incompletos. A seguir, é utilizada a extensão cURL do PHP para a submissão e recepção dos resultados. Esta extensão possui uma quantidade muito grande de opções de uso, incluindo utilização de autenticação por *cookie*, preenchimento de dados de formulário, estabelecimento de conexões SSL, dentre outras. A execução da submissão é seguida da recepção dos resultados completos. É necessário que se avalie se este resultado é a página final, que fornece a lista completa das possíveis proteínas ou se é apenas uma página intermediária para espera do resultado final.

Após a recepção da página com os resultados finais, é necessária a execução de uma expressão regular que faça a separação de todos os dados do resultado que sejam interessantes para o cálculo do resultado consolidado.

Após a preparação do formulário para o novo serviço e do robô de submissão e recepção de resultados, basta que seja criado um arquivo executável que faça a chamada ao robô e que este arquivo seja incluído na “crontab” ou nas “tarefas agendadas” do servidor.

## 5.3. UTILIZAÇÃO DO SISTEMA

Em relação à utilização do sistema, de acordo com a tabela de prioridade e com a visão de projeto, o usuário deverá proceder da seguinte maneira:

1 – criação de um usuário: o usuário, ao tentar entrar em qualquer opção de acesso restrito do programa, ou ao clicar na opção *Login*, é direcionado para a tela de *login*. Nesta tela, existe a opção de “Criar Novo Usuário”. Para tanto, o usuário deve preencher um formulário simples de cadastro, que está protegido contra as formas mais conhecidas de ataques por *SQL-injection*, bem como o restante dos formulários do sistema. A tela apresentada para o usuário é ilustrada abaixo:



Not Logged in

# PROTEIN LOCATOR

Home | Login | Data Entry | Available Searches | Start Search | Results | Help | CBSP | UnB

Create new user

Please fill this form to access the restricted areas:

Full name:

User-id (E-mail):

Password:

Confirm Password:

**Figura 5-1 Criação de novo usuário**

2 – após a criação, o usuário é direcionado para a tela de *Login*. Nesta etapa, também protegida contra ataques de *SQL-injection*, é calculado o resumo da senha do usuário, utilizando o algoritmo SHA-1, para comparação com a senha armazenada no banco, também um resumo da senha. Dessa forma, a senha que circula pela rede entre a máquina do cliente e o servidor é na verdade o resumo da senha, não sendo passado seu texto em claro. A tela apresentada ao usuário para que faça o *login* é apresentada abaixo:

**Figura 5-2 Tela de *login* de usuário**

3 – Depois de feito o *login*, o usuário é direcionado para uma tela com a lista de todas as pesquisas cadastradas por ele. No caso inicial, ele deve criar uma nova pesquisa, preenchendo o formulário *Generic*, na versão Básica ou Avançada. Em todas as etapas do sistema, após o *login* do usuário, são exibidas, no canto superior direito da tela, informações de nome de usuário, *link* para *logout*, qual a pesquisa em que o usuário está trabalhando e um link para que ele possa visualizar todas as pesquisas (mesma página para a que ele é direcionado após o *login*). A tela de visualização de pesquisas é exibida na figura abaixo:

**Figura 5-3 Visualização das pesquisas do usuário**

4 – Para criação de uma pesquisa, o usuário deve selecionar a opção “*Generic*” na aba “*Data Entry*”. Após o preenchimento e a submissão do formulário, desde que atendidas as regras de negócio, são indicadas as possíveis próximas etapas para o usuário. As telas de formulário de criação de pesquisa e a exibição das possíveis etapas estão exibidas abaixo:

# PROTEIN LOCATOR

Logged in as higor@unb.br  
[logout](#)  
using query NONE  
[view queries](#)

[Home](#) | [Login](#) | [Data Entry](#) | [Available Searches](#) | [Start Search](#) | [Results](#) | [Help](#) | [CBSP](#) | [UnB](#)

## Protein generic data(basic form):

Query name

pI \* within pI range \* pl units

MW(in Daltons) \* within mw range \* %

Taxonomy \*

Keywords:

Comments:

Figura 5-4 Criação de uma pesquisa

# PROTEIN LOCATOR

Logged in as higor@unb.br  
[logout](#)  
[view queries](#)

[Home](#) | [Login](#) | [Data Entry](#) | [Available Searches](#) | [Start Search](#) | [Results](#) | [Help](#) | [CBSP](#) | [UnB](#)

Your query teste has been added to the database.

Now, you can fill:

- ◆ [advanced Aminoacid Composition form](#) or [basic Aminoacid Composition form](#)
- ◆ [advanced Protein Sequence form](#) or [basic Protein Sequence form](#)
- ◆ [advanced Peptide Mass Fingerprint form](#) or [basic Peptide Mass Fingerprint form](#)

Figura 5-5 Possíveis próximas etapas

5 – após a criação de uma pesquisa, o usuário pode inserir os dados de Composição de Aminoácido (apenas um formulário é disponibilizado por pesquisa), informações de “*Peptide Mass Fingerprint*” e informações de seqüência de proteína (permitidos tantos quantos forem necessários). Todas essas informações estão relacionadas diretamente com a pesquisa previamente criada. Para cada um dos formulários são utilizadas as regras de negócio previamente estabelecidas. As telas do sistema que permitem a inserção dessas informações são exibidas a seguir:

# PROTEIN LOCATOR

Home | Login | Data Entry | Available Searches | Start Search | Results | Help | CBSP | UnB

## Amino Acid Composition of Unknown Protein

mol percent

Calibration protein:

	Comp.	Weight	Calibr.		Comp.	Weight	Calibr.
ALA	-1	1	-1	ILE	-1	1	-1
ARG	-1	1	-1	LEU	-1	1	-1
ASN	-1	1	-1	LYS	-1	1	-1
ASP	-1	1	-1	MET	-1	1	-1
ASX	-1	1	-1	PHE	-1	1	-1
CYS	-1	1	-1	PRO	-1	1	-1
GLN	-1	1	-1	SER	-1	1	-1
GLU	-1	1	-1	THR	-1	1	-1
GLX	-1	1	-1	TRP	-1	1	-1
GLY	-1	1	-1	TYR	-1	1	-1
HIS	-1	1	-1	VAL	-1	1	-1

Fragment Search Window

Figura 5-6 Adicionar composição de aminoácidos

# PROTEIN LOCATOR

Home | Login | Data Entry | Available Searches | Start Search | Results | Help | CBSP | UnB

## Peptide Mass Fingerprint

Data for digestion #

Cleavage agent:

Ordinary:

User defined:

Cleaves at:  N-term  C-term

Arg Ala of Asn but not near Arg Ala Asn  
coupled modification:

Missed cleavages:

Modifications:

Fixed modifications

Acetyl (K)  
Acetyl (N-term)  
Acetyl (Protein N-term)

Variable modifications

Acetyl (K)  
Acetyl (N-term)  
Acetyl (Protein N-term)

Mass tolerance:  Dalton Instrument:

*Monoisotopic*

*Average*

file:

Figura 5-7 Adicionar informações de fingerprint

**Figura 5-8 Adicionar informações de seqüência de proteína**

6 – após o preenchimento de cada um dos formulários, o usuário pode revisar sua pesquisa, podendo editar ou apagar qualquer um dos formulários preenchidos. Para acessar esta funcionalidade, o usuário poderá selecionar a visualização da pesquisa específica a partir da tela de visualização de todas as pesquisas, ou selecionar a opção “*Review Saved Forms*” na aba “*Start Search*”.

**Figura 5-9 Visualizar informações detalhadas**

7 – ao revisar os formulários, é realizada a análise dos dados e a verificação de qual programa pode ser utilizado para a identificação. Se algum programa estiver disponível para a pesquisa, é oferecida a opção de submeter para busca de proteínas com os dados disponíveis. Esta é a etapa de verificação das informações do usuário, de acordo com os programas que podem ser utilizados para a identificação de proteínas. Após a análise dos dados, são oferecidas ao usuário as opções de submeter aos possíveis serviços. Caso os dados do usuário sejam incompatíveis com o programa, são exibidas ao usuário as informações incompatíveis. Após a seleção dos programas desejados, o usuário deverá pressionar o botão “Search”, que indicará que a pesquisa foi submetida com sucesso. A tela que indica o sucesso na submissão é apresentada abaixo:



**Figura 5-10 Sucesso na submissão de pesquisa**

8 – alguns minutos após selecionar a busca, o usuário pode acessar o *link* de resultados para visualizar tanto os resultados consolidados quanto os originais dos programas e tomar sua decisão sobre a sua pesquisa. É permitido que o usuário faça novas buscas com os dados, que podem ser alterados a fim de se obter melhores resultados. A tela com os resultados consolidados e os isolados de cada programa de identificação é exibida abaixo:

The screenshot shows the PROTEIN LOCATOR website interface with search results. It includes the same navigation menu as Figure 5-10. Below the menu, there are instructions: "To view the original results from the selected programs, you must go to the [queries](#) visualisation." and "To view the Consolidated results from the selected programs, you must go to the [results](#) visualisation." Below this, it says "Complete consolidated results (separated by the query request):". The results are presented in a table for query ID 20080528165054.

20080528165054						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
P80674	4.34471217678E-59	0.0317676807308	7.16096872546E-10	5.8e-06	1e-48	--
B0XAP8	1.0E-12	0.1000000000001	--	--	1e-12	--
Q7Q3Q1	4.0E-12	0.1000000000004	--	--	4e-12	--
Q17D93	8.0E-12	0.1000000000008	--	--	8e-12	--

**Figura 5-11 Resultados consolidados**

## 6. RESULTADOS E DISCUSSÕES

### 6.1. AMBIENTE DE TESTE

O ambiente computacional utilizado para teste foi um servidor com memória RAM de 1GB, processador Intel Pentium 4, disco rígido de 120GB, placa de rede ethernet 10/100 Mbps e sistema operacional Linux Fedora 8. O servidor está conectado à rede local da Universidade de Brasília, utilizando seus serviços para comunicação de dados.

A estação conta com servidor web “Apache”, utilizando linguagem de programação “PHP” e “JavaScript”, servidor de banco de dados “MySQL”, a ferramenta “phpMyAdmin” para administração do banco de dados, os programas “Crontab” e “Cron” para execução programada dos robôs de submissão e recepção de resultados e firewall “Iptables”.

### 6.2. METODOLOGIA DE TESTE

Para avaliar a confiabilidade do sistema e confirmar a melhora da confiabilidade nas pesquisas ao se realizar identificação com vários programas, foram realizados testes com proteínas teóricas. Para tanto, foram eleitas algumas proteínas do banco de dados Uniprot.

A metodologia definida para a escolha das proteínas foi cobrir uma ampla faixa de pI, desde 3 até 12, e de massas moleculares (de 9000 a 180000 *Daltons*), e também a cobertura de diferentes composições de aminoácidos. Para atender a esses requisitos, foram realizadas buscas com o programa Multident, que localiza proteínas utilizando informações de taxonomia, pI e MW, composição de aminoácidos, dentre outras. Esse programa apresenta o código da proteína, que precisa ser obtida diretamente do banco de dados por meio de buscas no site do NCBI [76]. A partir da seqüência, foram calculados os dados de composição percentual de aminoácidos e, utilizando o programa GPMAW [77], foi realizada a digestão teórica da proteína com a enzima Tripsina, o mesmo programa foi utilizado para calcular a lista de massas de peptídeos das proteínas.

Todos os dados obtidos foram cadastrados no sistema e foram realizadas as pesquisas para identificação das proteínas. Os resultados são apresentados no próximo capítulo desta dissertação.

Para simular a realização de um experimento, foram feitas supressões de informações da seqüência em sua composição de aminoácidos e também da lista de massas de peptídeos, bem como a modificação da composição de aminoácidos. Para cada uma das seqüências das proteínas, foi realizada a digestão teórica com tripsina, utilizando o programa GPMAW, e foram utilizadas apenas as seqüências dos peptídeos que continham de 20 a 30 resíduos de aminoácidos. Esta limitação de quantidade de resíduos é estabelecida pelas próprias técnicas de seqüenciamento e pelo manuseamento das amostras em bancada de laboratório. Nas listas de massa dos peptídeos, de cada uma das pesquisas, foram deixadas apenas as massas entre 700 e 2600, pois as demais não podem ser distinguidas do ruído que também aparece no espectro. As alterações na composição de aminoácidos das proteínas visam simular situações de contaminação da amostra em laboratório, geralmente por algum pó ou metal, bem como problemas ocorridos na etapa de hidrólise.

As proteínas obtidas conforme as condições estabelecidas acima foram cadastradas no sistema, criando-se dezoito diferentes pesquisas (nove pesquisas com os dados completos das proteínas e nove com os dados parciais, simulando experimento em bancada). Para efeito de comparação entre o método de Fisher e o algoritmo QFAST, foram obtidos resultados consolidados por meio dos dois algoritmos para a primeira pesquisa. Com as demais, os resultados consolidados foram obtidos somente pelo algoritmo QFAST.

Para a avaliação final da funcionalidade do programa, foram comparados os resultados isolados, fornecidos por cada um dos programas utilizados para identificação das proteínas, e os resultados consolidados pelo sistema, a fim de se demonstrar que a utilização de um número maior de programas favorece a identificação da proteína. Estes dados serão abordados no tópico 6.4 desta dissertação.

### **6.3. DESCRIÇÃO DAS PROTEÍNAS UTILIZADAS**

Para realizar os testes no sistema, foram eleitas proteínas bem distribuídas dentre a faixa de pI e de massa molecular (MW), ou seja, foram utilizadas informações de proteínas com funções diferentes, de organismos diferentes, com pI variando de 3 a 12 e massa molecular variando de 9000 a 180000 Daltons. A seguir, é apresentada a lista com as informações das nove proteínas utilizadas para testes;

**Proteína 1 - “Gas vesicle protein gvpJ 2”, cujo código de identificação no banco de dados do NCBI é P33956**

pI: 7 MW: 400000 Da

**Seqüência:**

MSDPKPTRSQGDLAETLELLLDKGVVVNADIAVSVGDTTELLGVELRAAIASFETA AEYGLDFPTGTDME  
RVTAAGVDADDSKSVLERPDPPTTEGSE

**Proteína 2 - “Serine-aspartate repeat-containing protein C precursor”, cujo código de identificação no banco de dados do NCBI é Q7A781**

pI: 4 MW: 90000 Da

**Seqüência:**

MNNKKTATNRKGMIPNRLNKFSIRKYSVGTASILVGTTLIFGLSGHEAKAAEHTNGELNQSKNETTAPS  
ENKTTEKVD SRQLKDNTQTATADQPKVTMSDSATVKETSSNMQSPQNATASQSTTQTSNVTTNDKSST  
TYSNETDKSNLTQAKNVSTTPKTTTIKQRALNRMAVNTVAAPQQGTNVNDKVHFTNIDIAIDKGHVNK  
TTGNTEFWATSSDVLKLNKANYTIDDSVKEGDTFTFKYGYFRPGSVRLPSQTQNLNAQGNIIAKGIYD  
SKTNTTTYTFTNYVDQYTNVSGSFEQVAFAKRENATTDKTAYKMEVTLGNDTYSKDVIVDYGNQKGG  
QLISSTNYINNEDLSRNMTVYVNQPKKTYTKETFTVNTLTGYKFNPDANKFKIYEVTDQNFVDSFTPDT  
SKLKDVTGQFDVIYSNDNKTATVDLLNGQSSDKQYIIQQVAYPDNSSTDNGKIDYTLETQNGKSSWSN  
SYSNVNGSSTANGDQKKYNLGDYVWEDTNKDGKQDANEKGIKGVYVILKDSNGKELDRTTTDENGK  
YQFTGLSNGTYSVEFSTPAGYTPPTANAGTDDAVSDGLTTTGVIKDADNMTLDSGFYKTPKYS LGDY  
VWYDSNKDGKQDSTEKGIKGVKVTLQNEKGEVIGTTETDENGKYRFDNLDGKYKVFIEKPAGLTQTG  
TNTTEDDKDADGGEVDVTITDHDFTLDNGYEEETS DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS  
DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS  
DSDNDS DSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDSDS  
DSDAGKHTPTKPMSTVKDQHKTAKALPETGSENNNSNNGTLFGGLFAALGSLLLFGRRKKQNK

**Proteína 3 - “Cuticle protein 1 (Bc-NCP1)”, cujo código de identificação no banco de dados do NCBI é P80674**

pI: 6 MW: 9000 Da

**Seqüência:**

QADKYPAGLNPALCPNYPNCDNALIALYSNVAPAIPYAAAAYNYPAGVSPAACP NYPF CGAIAPLGYHV  
REYPAGVHPAACPNYPYCV

**Proteína 4 - “Complement C3 [Precursor]”, cujo código de identificação no banco de dados do NCBI é P01024**

pI: 6 MW: 180000 Da

**Seqüência:**

MGPTSGPSLLLLLTHLPLALGSPMYSIITPNILRLESEETMVLEAHDAQGDVPVTVTHDFPGKKLVLS  
SEKTVLTPATNHMGNVFTIPANREFKSEKGRNKFVTVQATFGTQVVEKVVLVSLQSGYLFIQTDKTIY  
TPGSTVLYRIFTVNHKLLPVGRTVMVNIENPEGIPVKQDSLSSQNQLGVLPLSWDIPELVNMGQWKIRA  
YYENSPQQVFSTEFVKEYVLPSEFVIVEPTEKFYIYNEKGLEVTITARFLYGKKVEGTAFFVIFGIQDGE  
QRISLPESLKRIPIEDGSGEVVLRSRKLVDGQNPRAEDLVGKSLYVSATVILHSGSDMVQAERSGIPIVT  
SPYQIHFTKTPKYFKPGMPFDLMVFTVNPDGSPAYRVPVAVQGEDTVQSLTQGDGVAKLSINTHPSQKP  
LSITVTRTKKQELSEAEQATRMTQALPYSTVGNSNNYLHLSVLRTELRPGETLNVNFLLRMDRAHEAKIR  
YYTYLIMNKGRLLKAGRQVREPGQDLVVLPLSITTDIFIPSFRLVAYYTLIGASGQREVVADSVWVDVKD  
SCVGSLLVVKSGQSEDRQPVPQGQMTLIEGDHGARVVLVAVDKGVFVLNKNKLTQSKIWDVVEKA  
DIGCTPGSGKDYAGVFS DAGLTFTSSSGQQAQRAELQCPQPAARRRRSVQLTEKRMKVGKYPKELR  
KCCEDGMRENPMRFSCQRRTRFISLGEACKKVFLDCCNYITELRRQHARASHLGLARSNLDEIIAENI  
VSRSEFPESWLWNVEDLKEPPKNGISTKLMNIFLKDSITTWEILAVSMSDKKGCVADPFEVTVMQDFFI  
DLRLPYSVVRNEQVEIRAVLYNYRQNLKVRVELLHNPFCSLATTKRRHQQTVTIPPKSSLSVPYVIV  
PLKTGLQEVEVKAAYVHHFISDGVKSLKVVPEGIRMNKTVAVRTLDPERLGREGVQKEDIPPADLSD  
QVPDTESETRILLQGTPVAQMTEDA VDAERLKHIVTPSGCGEQNMIGMTPTVIAVHYLDETEQWEKFG  
LEKRQGALELIKKGTYQQLAFRQPSSAFAAFVKRAPSTWLTAYVVKVFS LAVNLI AIDSVL CGAVKW  
LILEKQKPDGVFQEDAPVIHQEMIGGLRNNNEKDMALTAFLVLSLQEAKDICEEQVNSLPGSITKAGDFL  
EANYMNLQRSYTVAIAGYALAQMGRLKGPLLNKFLTTAKDKNRWEDPGKQLYNVEATS YALLALLQ  
LKDFDFVPPVVRWLNEQRYYGGYGSTQATFMVFQALAQYQKDAPDHQELNLDVSLQLPSRSSKITH  
RIHWESASLLRSEETKENEGFTVTAEGKGQGTLSVVTMYHAKAKDQLTCNKFDLKVTIKPAPETEKRP  
QDAKNTMILEICTRYRGDQDATMSILDISMGTGFAPDTDDLKQLANGVDRYISKYELDKAFSDRNTLII  
YLDK VSHSEDDCLAFKVHQYFNVELIQPGAVKVYAYYNLEESCTRFYHPEKEDGKLNKLCRDEL CRCA  
EENCFIQKSDDKVTLEERLDKACEPGVDYVYKTRLVKVQLSNDFDEYIMAIEQTIKSGSDEVQVGGQRT  
FISPIKREALKLEEKHYLMWGLSSDFWGEKPNLSYIIGKDTWVEHWPEEDECQDEENQKQCQDLGA  
FTESMVVFGCPN

**Proteína 5 - “Cell division topological specificity factor”, cujo código de identificação no banco de dados do NCBI é A6WMJ7**

pI: 8 MW: 9000 Da

**Seqüência:**

MSLLDYFKSKKKPSTAVMAKERLQIIVAHQRGQRDTPDYFPQMKQEIIAVIRKYVQISDDQVSVQLDQN  
DANLSVLELNVTL PDR

**Proteína 6 - “Acyl-coenzyme A dehydrogenase (ACDH)”, cujo código de identificação no banco de dados do NCBI é Q8Z937**

**pI:** 8 **MW:** 80000 Da

**Seqüência:**

MMILSIIATVVLLGALFYHRVSLFLSSLILLAWTAALGVAGLWSIWLLVPLAAILVPFNLTMPMRKSMISAP  
VFRGFRKVMPPMSRTEKEAIDAGTTWEGDLFQGKPDWKKLHNYQPQLTAEQAFLDGPVEEACR  
MANDFQITHELADLPPELWAYLKEHRFFAMIIKKEYGGLEFSAYVQSRVLQKLSGVSGILAITVGVPNLSL  
GPGELLQHYGTEEQKNHYLPRLARGQEIPCFALTSPEAGSDAGAIPDTGVVCMGEWQGGQVLMGRLT  
WNKRYITLAPIATVLGLAFKLSDPDRLLGGEEELGITCALIPTSTPGVEIGRRHFPLNVPFQNGPTRGNDIF  
VPIDYIIGGPKMAGQGWRMLVECLS VGRGITLPSNSTGGVKSVALATGAYAHIRRQFKISIGKMEGIEEP  
LARIAGNAYVMDAAASLITYGIMLGEKPAVLSAIVKYHCTHRGQQSIIDAMDITGGKGIMLGESNFLAR  
AYQGAPIAITVEGANILTRSMIFGQGAIRCHPYVLEEMAAAQNNDVNAFDKLLFKHIGHVGSNTVRSF  
WLGLTRGLTSHPTGDATKRYQHLNRLSANLALLSDVSMAVLGGSLKRRERISTR LGDVLSQLYLAS  
AVLKRYDDEGRHEADLPLVHWGVQDALYRAEQAMDDLQNFNRVVAGLLTAMIFPTGRHYLAPSD  
KLDHAVAKILQVPNATRSRIGRQYLTPAEHNPVGLLEEALRDVIAADPIHQICKELGKNLPFTRLDEL  
ARNALAKGLIDKDEAAILAKAESRLRSINVDDFEPEALATKPKLPEKVRKVEAA

**Proteína 7 - “Chloroplast 30S ribosomal protein S17”, cujo código de identificação no banco de dados do NCBI é O46903**

**pI:** 10 **MW:** 9000 Da

**Seqüência:**

MSIKERLGLVISDKMDKTVVVSIANRVTHKRYGKIVTKTKKYKVHDPNNNCQVGDLLINETRPLSKTK  
RWMFKEIKQKSLKLDKDTIGE

**Proteína 8 - “Bcl-2-associated transcription factor 1 (Btf)”, cujo código de identificação no banco de dados do NCBI é Q8K019**

**pI:** 10 **MW:** 80000 Da

**Seqüência:**

MGRSNSRSHSSRSKRSQSSSRSRSRSHSRKKRYSSRSRSRTYSRSRSRDRIYSRDYRRDYRNNRGMRRP  
YGYRGRGRGYYQGGGGRYHRGGYRPVWNRHRSRPRRGRSRSPKRRSVSSQRSRSRRRSYRSSRS  
PRSSSRSSSPYSKSPVSKRRGSQEKQTKKAEGEPQEE SPLKSKSQEEPKDTFEHDPSESIDFNKSATSG  
DIWPGLSAYDNSRSPHSPPIATPPSQSSSCSDAPMLSTVHSAKNTPSQHSHSIQHSPERSGSGSVGNSS  
RYSQNSPIHHIPSRRSPAKTITPQNAPREESRGRSSFYPEGDQETAKTGKFLKRFTDEESRVFLDRGNI  
RDKEAPKEKGSEKGRADGDWDDQEVLDYFSDKESAKQKFHDSEGDDTEETEDYRQFRKSVLADQGKS  
FATSSHRNTEEEGPKYKSKVSLKGNRESDFREEKNYKLLKETAYIVERPSTAKDKHKEEDKGS DRITVK  
KEVQSPEQVKSEKLKELFDYSPPLHKSLDAREKSIFREESPLRIKMIASDSHRPEVKLKMAPVPLDDSNRP  
ASLTKDRLLASTLVH SVKKEQEFRSIFDHIKLPQANKSTSEFIQHIVSLVHHVKEQYFKSPAVTLNERFT  
SYQKATEEHSTRQKSPEIHRRIDISPSALRKHTRLAGEERGFKEEIQKGDKKLRCD SADRHDIDRRRKE  
RSKERGDSKGSRESSGRKQEKTPKDYKEYKPYKDDSKHKGRERDHSRSSSSSSASPSSPSSREEKESKKE

REEEFKTHHEMKDYSGFAGVSRPRGTTFFRIRGRGRARGVFAQTNTGPNNSTTFQKRPKEEEWDPEYTP  
KSKKYFLHDDRDDGVDYWAKRGRGRGTFQRGRGRFNFKKSGSSPKWTHDKYQGDGIVEDD  
EETMENNEEKKDRRKEEKE

**Proteína 9 - “30S ribosomal protein S20”, cujo código de identificação no banco de dados do NCBI é A1BAN4**

**pI:** 12 **MW:** 9000 Da

**Seqüência:**

MANTPQSKKRARQLERRTAVNKARRSRIRTFRLKVEEAIASGNAEIAREALNSAQPELMRGVTKGVIHK  
NTAARKMSRLSARVKALATA

## **6.4. RESULTADOS DOS TESTES DE IDENTIFICAÇÃO**

A seguir, serão exibidos os resultados obtidos para cada um dos experimentos. Em cada experimento, foram feitas análises pelos programas isolados e pela combinação do conjunto de programas disponíveis no sistema *Protein Locator*, para fins de comparação e demonstração da melhora nos resultados obtidos. Para cada proteína analisada são exibidas duas tabelas, uma corresponde aos resultados de buscas com dados completos da proteína e outra corresponde aos resultados de buscas a partir de dados parciais, simulando situações experimentais, conforme descrito na metodologia. Nos resultados, quanto menor o score obtido, maior a probabilidade de a proteína mostrada pelo programa corresponder aos dados experimentais. Vale ressaltar também que os valores estão em notação científica e a análise de cada resultado é feita no final do experimento.

**Proteína 1 - “Gas vesicle protein gvpJ 2”, cujo código de identificação no banco de dados do NCBI é P33956**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-1 Resultados da busca com dados completos da proteína P33956**

20080529140933						
<u>NCBI ID</u>	<u>Consolided Evalue</u>	<u>PL Score</u>	<u>AACompident Evalue</u>	<u>FASTA Evalue</u>	<u>Blast Evalue</u>	<u>Mascot Evalue</u>
P33956	3.893455668293E-75	0.018294132805025	4.9420904465175E-10	7.4e-05	6e-60	1.6483516067108e-08
P24374	2.2810997471436E-41	0.066397319004226	--	0.023	1e-41	--
Q02235	1.1703150281875E-40	0.066836729199002	--	0.024	5e-41	--
A3KIB1	4.0E-20	0.1	--	--	4e-20	--
B0R8K7	1.0E-41	0.1	--	--	1e-41	--
B0R9R4	6.0E-60	0.1	--	--	6e-60	--
B1I4U0	2.0E-19	0.1	--	--	2e-19	--
Q18JB0	6.0E-39	0.1	--	--	6e-39	--
Q46FM3	9.0E-18	0.1	--	--	9e-18	--
A0B598	6.0E-16	0.1	--	--	6e-16	--
Q9HP20	2.7325213998456E-10	0.10000000027325	2.7325213998456E-10	--	--	--
Q9HP43	2.7325213998456E-10	0.10000000027325	2.7325213998456E-10	--	--	--
Q9HPT6	2.7327965059151E-10	0.10000000027328	2.7327965059151E-10	--	--	--
Q9HMK4	2.7330820226304E-10	0.10000000027331	2.7330820226304E-10	--	--	--
O52027	2.7333785523311E-10	0.10000000027334	2.7333785523311E-10	--	--	--
Q9HR31	2.7336867447372E-10	0.10000000027337	2.7336867447372E-10	--	--	--
Q9HMC7	2.7350510772987E-10	0.10000000027351	2.7350510772987E-10	--	--	--
Q5V3P6	2.7375985429732E-10	0.10000000027376	2.7375985429732E-10	--	--	--
P02944	2.7380976994281E-10	0.10000000027381	2.7380976994281E-10	--	--	--
Q5UX95	2.7380976994281E-10	0.10000000027381	2.7380976994281E-10	--	--	--
Q9HPB3	2.7380976994281E-10	0.10000000027381	2.7380976994281E-10	--	--	--
Q9HQW6	2.739757711642E-10	0.10000000027398	2.739757711642E-10	--	--	--
Q5V0I9	2.7403727863462E-10	0.10000000027404	2.7403727863462E-10	--	--	--
Q3INE3	2.7410231583223E-10	0.1000000002741	2.7410231583223E-10	--	--	--
Q3INP4	2.7410231583223E-10	0.1000000002741	2.7410231583223E-10	--	--	--
A1WVW4	2.7468834563726E-10	0.10000000027469	2.7468834563726E-10	--	--	--
Q28430	0.0001098901115276	0.10010989011153	--	--	--	0.0001098901115276
Q8U3E3	0.00037087912742908	0.10037087912743	--	--	--	0.00037087912742908
Q2FTF8	0.00096153846153846	0.10096153846154	--	--	--	0.00096153846153846
Q57944	0.0020604395604396	0.10206043956044	--	--	--	0.0020604395604396
Q58453	0.0021978021978022	0.1021978021978	--	--	--	0.0021978021978022
Q8TMG9	0.0025412087912088	0.10254120879121	--	--	--	0.0025412087912088
Q9YAU5	0.0026098901098901	0.10260989010989	--	--	--	0.0026098901098901
A2BN38	0.0028846153846154	0.10288461538462	--	--	--	0.0028846153846154
A8A929	0.0028846153846154	0.10288461538462	--	--	--	0.0028846153846154
Q58754	0.0028846153846154	0.10288461538462	--	--	--	0.0028846153846154
Q8TYB7	0.003021978021978	0.10302197802198	--	--	--	0.003021978021978

**Tabela 6-2 Resultados da busca com dados parciais da proteína P33956**

20080718111624						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
P33956	1.220270541387E-20	0.031767013954412	5.0756640345655E-10	0.64	3e-06	5.4656007364234e-09
Q9HN60	4.6196905899943E-10	0.10000000046197	4.6196905899943E-10	--	--	--
Q9HP43	4.6196905899943E-10	0.10000000046197	4.6196905899943E-10	--	--	--
Q9HPT6	4.6201732453549E-10	0.10000000046202	4.6201732453549E-10	--	--	--
Q9HR31	4.621195505527E-10	0.10000000046212	4.621195505527E-10	--	--	--
Q9HP20	4.621737395122E-10	0.10000000046217	4.621737395122E-10	--	--	--
Q9HRE8	4.6235018588475E-10	0.10000000046235	4.6235018588475E-10	--	--	--
Q9HMC7	4.6241413235346E-10	0.10000000046241	4.6241413235346E-10	--	--	--
Q9HPB3	4.6278082545779E-10	0.10000000046278	4.6278082545779E-10	--	--	--
Q3ISR6	4.6286520599518E-10	0.10000000046287	4.6286520599518E-10	--	--	--
Q5UX95	4.6286520599518E-10	0.10000000046287	4.6286520599518E-10	--	--	--
Q5V3P6	4.6286520599518E-10	0.10000000046287	4.6286520599518E-10	--	--	--
Q3IQL3	4.629539303288E-10	0.10000000046295	4.629539303288E-10	--	--	--
Q9HQW6	4.6314582486992E-10	0.10000000046315	4.6314582486992E-10	--	--	--
Q5V0I9	4.6359971003419E-10	0.1000000004636	4.6359971003419E-10	--	--	--
Q3INE3	4.6373099406765E-10	0.10000000046373	4.6373099406765E-10	--	--	--
Q3INP4	4.6373099406765E-10	0.10000000046373	4.6373099406765E-10	--	--	--
A1WVN4	4.6570472177573E-10	0.1000000004657	4.6570472177573E-10	--	--	--
B0R8K7	3.0E-6	0.100003	--	--	3e-06	--
B0R9R4	3.0E-6	0.100003	--	--	3e-06	--
P24374	3.0E-6	0.100003	--	0.8	3e-06	--
Q02235	6.0E-6	0.100006	--	1	6e-06	--
Q18JB0	6.0E-6	0.100006	--	--	6e-06	--
Q28430	6.8320010931202E-5	0.10006832001093	--	--	--	6.8320010931202e-05
Q8U3E3	0.00019129602734961	0.10019129602735	--	--	--	0.00019129602734961
Q2FTF8	0.00066270409300165	0.100662704093	--	--	--	0.00066270409300165
Q57944	0.001366400218624	0.10136640021862	--	--	--	0.001366400218624
Q58453	0.0014347202295552	0.10143472022956	--	--	--	0.0014347202295552
Q9YAU5	0.0017763202842112	0.10177632028421	--	--	--	0.0017763202842112
Q58754	0.0019129603060736	0.10191296030607	--	--	--	0.0019129603060736
Q8TJC6	0.0019129603060736	0.10191296030607	--	--	--	0.0019129603060736
A2BN38	0.0019812803170049	0.101981280317	--	--	--	0.0019812803170049
Q46FM3	0.003	0.103	--	--	0.003	--
B1I4U0	0.036	0.136	--	--	0.036	--

Pode-se perceber, ao analisar os resultados obtidos com dados parciais e com os completos, que a identificação da proteína pode ser feita com exatidão em ambas as situações, com boa discriminação mesmo em relação a seqüências similares, e que a utilização de mais programas possibilitou a diferenciação entre alguns resultados. Caso fosse utilizado apenas o programa Blast, que geralmente é o único programa utilizado pelos cientistas para análises de seqüência de proteínas, o resultado seria inconclusivo, sendo necessário que fossem feitas alterações nos dados submetidos.

Para fins de comparação entre os métodos QFAST e Fisher, a seguir, é apresentada uma tabela com o e-valor consolidado, obtido pelas buscas por dados completos da proteína P33956:

**Tabela 6-3 Resultados comparativos dos métodos QFAST e Fisher**

20080529140933

<u>NCBI ID</u>	<u>Fisher e-value</u>	<u>QFAST e-value</u>
P33956	4.837778e-71	<b>4.8377781269775E-71</b>
P24374	2.281100e-41	<b>2.2810997471436E-41</b>
Q02235	1.170315e-40	<b>1.1703150281875E-40</b>

Com estes resultados, percebe-se que o método de QFAST revela o mesmo valor que o método de Fisher, ou valores muito próximos, porém de forma muito mais rápida, uma vez que o tempo observado para cálculo pelo método de Fisher foi cinco vezes maior do que o tempo para o método QFAST. A demora do método de Fisher é devida às operações exponenciais e também à necessidade de se utilizar um programa externo, o R, para executar as operações, que necessitam de elevada precisão. Em experimentos realizados neste projeto, o tempo medido para o processamento dos resultados combinados pelo método de Fisher foi de cerca de 30 segundos, enquanto que pelos algoritmos QFAST e PLscore, 5 segundos.

**Proteína 2 - “Serine-aspartate repeat-containing protein C precursor”, cujo código de identificação no banco de dados do NCBI é Q7A781**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-4 Resultados da busca com dados completos da proteína Q7A781**

20080529180011						
<u>NCBI ID</u>	<u>Consolidated Evalue</u>	<u>PL Score</u>	<u>AACompident Evalue</u>	<u>FASTA Evalue</u>	<u>Blast Evalue</u>	<u>Mascot Evalue</u>
Q2G0L5	0	0.018284495517647	1.0669650330363E-11	1.6e-81	0.0	1.2530555395429e-15
Q5HIB4	0	0.018284495517707	1.1119244889273E-11	1.7e-86	0.0	1.2530555395429e-15
Q7A781	0	0.018284495517728	1.1286726405334E-11	2.2e-90	0.0	4.9300546646825e-32
Q99W48	0	0.018284495517728	1.1286726405334E-11	2.2e-90	0.0	4.9300546646825e-32
Q86487	0	0.018284495517729	1.1286726405334E-11	2.1e-86	0.0	1.2530555395429e-15
Q2FJ79	0	0.01828449551773	1.1286726405334E-11	2.1e-86	0.0	1.5817259468765e-14
Q6GBS6	1.8608203546033E-109	0.031766296778762	1.1008973513709E-11	4.9e-87	--	1.0065528872712e-16
Q8NXX7	3.2463341692663E-109	0.031766296778788	1.1119244889273E-11	8.5e-87	--	1.0065528872712e-16
Q6GJA7	1.0402080124971E-87	0.031766296778788	1.1119244889273E-11	1.7e-65	--	2.4650274135803e-16
A5IQB5	0	0.1	--	--	0.0	--
A6TZ38	0	0.1	--	--	0.0	--
A7WY25	0	0.1	--	--	0.0	--
A8YZQ9	0	0.1	--	--	0.0	--
Q6GBS5	1.3E-45	0.1	--	1.3e-45	--	--
Q99W47	1.0631564729141E-11	0.10000000001063	1.0631564729141E-11	--	--	--
A4IPX4	4.5192169184673E-6	0.10000451921692	--	--	--	4.5192169184673e-06

**Tabela 6-5 Resultados da busca com dados parciais da proteína Q7A781**

20080718113014						
NCBI ID	Consolided Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
Q7A781	3.0349834620396E-27	0.031786318362497	4.764357600999E-10	0.86	9e-08	2.899999915228525237953505211407900787889957427978515625e-14
Q99W48	3.0349834620396E-27	0.031786318362497	4.764357600999E-10	0.86	9e-08	2.899999915228525237953505211407900787889957427978515625e-14
Q6GBS6	2.5175513108359E-18	0.031778246776616	4.7356656986762E-10	0.86	9e-08	4.999999873689375817775726318369375e-05
Q8NXX7	2.5175513108359E-18	0.031778246776616	4.7356656986762E-10	0.86	9e-08	4.999999873689375817775726318369375e-05
Q6GJA7	8.1016719599745E-18	0.031806867932549	4.7159033558164E-10	0.83	9e-08	0.00016999999343417584896087646484375
Q2G0L5	1.9967940419237E-17	0.031871229256117	4.6747635544521E-10	0.89	9e-08	0.00044000000343658030033111572265625
O86487	2.0216809744964E-17	0.031871229257575	4.7356656986762E-10	0.86	9e-08	0.00044000000343658030033111572265625
Q5HIB4	2.0216809744964E-17	0.031871229257575	4.7356656986762E-10	0.86	9e-08	0.00044000000343658030033111572265625
Q2FJ79	1.5477424030013E-16	0.032644422077822	4.7356656986762E-10	0.86	9e-08	0.00370000000111758708953857421875
Q99W47	4.6619811143795E-10	0.1000000004662	4.6619811143795E-10	--	--	--
A8YZQ9	9.0E-8	0.10000009	--	--	9e-08	--
A5IQB5	6.0E-7	0.1000006	--	--	6e-07	--
A6TZ38	6.0E-7	0.1000006	--	--	6e-07	--
A7WYZ5	6.0E-7	0.1000006	--	--	6e-07	--
Q2UWJ0	6.0E-7	0.1000006	--	--	6e-07	--
O86488	3.0E-6	0.100003	--	--	3e-06	--
Q2FJ78	3.0E-6	0.100003	--	--	3e-06	--
Q5HIB3	3.0E-6	0.100003	--	--	3e-06	--
Q6GBS5	3.0E-6	0.100003	--	--	3e-06	--
Q8NXX6	3.0E-6	0.100003	--	--	3e-06	--
Q2UWH0	0.0005	0.1005	--	--	5e-04	--
Q2UWH5	0.0005	0.1005	--	--	5e-04	--
Q2UWI3	0.0005	0.1005	--	--	5e-04	--
Q2UWI8	0.0005	0.1005	--	--	5e-04	--
Q2UWK0	0.0005	0.1005	--	--	5e-04	--
Q14U76	0.19	0.29	--	0.19	--	--
Q6GJA6	0.19	0.29	--	0.19	--	--

Nesta pesquisa, foram encontradas duas proteínas homólogas, cujas seqüências são idênticas, identificadas com o mesmo escore, que não recebem a mesma identificação no NCBI por serem de subespécies diferentes. Esta diferenciação da taxonomia (até o nível de subespécie) não é realizada em nenhum programa de identificação. Pode-se perceber, ao se analisar os resultados obtidos com dados parciais, que a utilização de mais programas possibilitou a diferenciação entre alguns resultados. Caso fossem utilizados apenas dados de seqüência de proteína ou composição de aminoácidos, seria impossível identificar a proteína correta. A discriminação em função de taxonomia é uma das funcionalidades de pós-processamento que poderá ser implementada em versões futuras do sistema aqui proposto.

**Proteína 3 - “Cuticle protein 1 (Bc-NCP1)”, cujo código de identificação no banco de dados do NCBI é P80674**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-6 Resultados da busca com dados completos da proteína P80674**

20080529180025						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
P80674	3.2816269746637E-63	0.018285524976903	7.1609687254649E-10	6.9e-06	1e-48	1.0049003178389e-06
B0XAP8	1.0E-12	0.1000000000001	--	--	1e-12	--
Q7Q3Q1	4.0E-12	0.1000000000004	--	--	4e-12	--
Q17D93	8.0E-12	0.1000000000008	--	--	8e-12	--
P83933	3.9387423243997E-10	0.10000000039387	3.9387423243997E-10	--	--	--
P83681	3.9392570377238E-10	0.10000000039393	3.9392570377238E-10	--	--	--
P80675	3.9393376125757E-10	0.10000000039393	3.9393376125757E-10	--	--	--
P82122	3.9399284754716E-10	0.10000000039399	3.9399284754716E-10	--	--	--
P82118	3.9400015273707E-10	0.100000000394	3.9400015273707E-10	--	--	--
P82119	3.9403968393543E-10	0.10000000039404	3.9403968393543E-10	--	--	--
P82120	3.940655762359E-10	0.10000000039407	3.940655762359E-10	--	--	--
P82121	3.941227780446E-10	0.10000000039412	3.941227780446E-10	--	--	--
P80676	3.9416069139327E-10	0.10000000039416	3.9416069139327E-10	--	--	--
P06936	3.942319788378E-10	0.10000000039423	3.942319788378E-10	--	--	--
Q08BY2	3.942319788378E-10	0.10000000039423	3.942319788378E-10	--	--	--
A1KXE4	3.9423703370432E-10	0.10000000039424	3.9423703370432E-10	--	--	--
Q0VFP2	3.9425474955842E-10	0.10000000039425	3.9425474955842E-10	--	--	--
Q10014	3.9426800816355E-10	0.10000000039427	3.9426800816355E-10	--	--	--
Q5RDV6	3.9427392440459E-10	0.10000000039427	3.9427392440459E-10	--	--	--
Q06521	3.9431507643014E-10	0.10000000039432	3.9431507643014E-10	--	--	--
Q8BGZ2	3.9435556621756E-10	0.10000000039436	3.9435556621756E-10	--	--	--
Q92567	3.9435556621756E-10	0.10000000039436	3.9435556621756E-10	--	--	--
Q293D6	2.0E-9	0.100000002	--	--	2e-09	--
A1Z7G3	4.0E-9	0.100000004	--	--	4e-09	--
Q8MYV9	4.0E-9	0.100000004	--	--	4e-09	--
Q8IV90	1.9194725376371E-05	0.10001919472538	--	--	--	1.9194725376371e-05
Q3T0E9	4.4034958058029E-5	0.10004403495806	--	--	--	4.4034958058029e-05
Q05826	5.7584173437127E-5	0.10005758417344	--	--	--	5.7584173437127e-05
Q19200	0.00010952283956897	0.10010952283957	--	--	--	0.00010952283956897
Q2TAF4	0.0001242011607163	0.10012420116072	--	--	--	0.0001242011607163
Q5JSJ4	0.0001242011607163	0.10012420116072	--	--	--	0.0001242011607163
Q5U4W6	0.0001242011607163	0.10012420116072	--	--	--	0.0001242011607163
Q7SYD9	0.0001242011607163	0.10012420116072	--	--	--	0.0001242011607163
Q8BND4	0.0001242011607163	0.10012420116072	--	--	--	0.0001242011607163
Q13409	0.0050809565747578	0.10508095657476	--	--	--	0.0050809565747578

**Tabela 6-7 Resultados da busca com dados parciais da proteína P80674**

20080718113143						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
P80674	5.6827563084267E-16	0.024339915675658	4.8017903451756E-10	0.046	6e-07	0.002676599826021
P83933	4.5240257763855E-10	0.1000000004524	4.5240257763855E-10	--	--	--
P83681	4.5246026989262E-10	0.10000000045246	4.5246026989262E-10	--	--	--
P80675	4.5246687320312E-10	0.10000000045247	4.5246687320312E-10	--	--	--
P82122	4.5252999853604E-10	0.10000000045253	4.5252999853604E-10	--	--	--
P82118	4.5257391693644E-10	0.10000000045257	4.5257391693644E-10	--	--	--
P82119	4.5258964649484E-10	0.10000000045259	4.5258964649484E-10	--	--	--
P82120	4.526289191475E-10	0.10000000045263	4.526289191475E-10	--	--	--
P82121	4.5267468680649E-10	0.10000000045267	4.5267468680649E-10	--	--	--
P80676	4.5280720114199E-10	0.10000000045281	4.5280720114199E-10	--	--	--
Q6VFT6	4.5284516049763E-10	0.10000000045285	4.5284516049763E-10	--	--	--
Q00269	4.5284949341018E-10	0.10000000045285	4.5284949341018E-10	--	--	--
Q8MIP2	4.5284949341018E-10	0.10000000045285	4.5284949341018E-10	--	--	--
Q10014	4.5285612496028E-10	0.10000000045286	4.5285612496028E-10	--	--	--
P11733	4.5285837158907E-10	0.10000000045286	4.5285837158907E-10	--	--	--
P45583	4.5285837158907E-10	0.10000000045286	4.5285837158907E-10	--	--	--
Q8BGZ2	4.529211795866E-10	0.10000000045292	4.529211795866E-10	--	--	--
Q92567	4.529211795866E-10	0.10000000045292	4.529211795866E-10	--	--	--
Q06521	4.5296622287925E-10	0.10000000045297	4.5296622287925E-10	--	--	--
Q7M461	0.001784399884014	0.10178439988401	--	--	--	0.001784399884014
P01458	0.0021189748622666	0.10211897486227	--	--	--	0.0021189748622666
Q9P1J3	0.0023420248477684	0.10234202484777	--	--	--	0.0023420248477684
P41314	0.0024535498405193	0.10245354984052	--	--	--	0.0024535498405193
Q2I2P2	0.002676599826021	0.10267659982602	--	--	--	0.002676599826021
P59861	0.0027881248187719	0.10278812481877	--	--	--	0.0027881248187719
A127G3	0.003	0.103	--	--	0.003	--
Q293D6	0.003	0.103	--	--	0.003	--
Q7Q3Q1	0.003	0.103	--	--	0.003	--
Q8MYV9	0.003	0.103	--	--	0.003	--
Q14210	0.0031226997970245	0.10312269979702	--	--	--	0.0031226997970245
P20658	0.0032342247897754	0.10323422478978	--	--	--	0.0032342247897754
Q6UY13	0.0033457497825263	0.10334574978253	--	--	--	0.0033457497825263
B0XAP8	0.008	0.108	--	--	0.008	--
Q17D93	0.008	0.108	--	--	0.008	--

Nesta pesquisa pode-se observar que, apesar de todos os programas terem fornecido a identificação da proteína correta isoladamente, a combinação deles permitiu um poder discriminatório mais marcante, como pode ser observado comparando-se o PLscore e o AACompIdent *e-value* para as duas primeiras candidatas quando se realizou a busca com dados truncados.

**Proteína 4 - “Complement C3 [Precursor]”, cujo código de identificação no banco de dados do NCBI é P01024**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-8 Resultados da busca com dados completos da proteína P01024**

20080529180051						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
P01024	0	0.01828449551764	1.0608311849555E-11	0	0.0	0
P01027	0	0.031766296777635	6.2863953529179E-12	0	0.0	--
Q2UVX4	0	0.031766296777635	6.2863953529179E-12	0	0.0	--
P01026	0	0.031766296777751	6.7743121273583E-12	0	0.0	--
P12387	0	0.031766296777751	6.7743121273583E-12	0	0.0	--
A7E236	0	0.1	--	--	0.0	--
P12247	1.2E-103	0.1	--	1.2e-103	--	--
P98093	3.0E-123	0.1	--	3e-123	--	--
P98094	2.9E-94	0.1	--	2.9e-94	--	--
Q01833	1.3E-157	0.1	--	1.3e-157	--	--
Q3L2T4	0	0.1	--	--	0.0	--
Q80XP1	0	0.1	--	--	0.0	--
Q90633	0	0.1	--	--	0.0	--
Q91132	9.3E-126	0.1	--	9.3e-126	--	--
Q9GKP1	0	0.1	--	--	0.0	--
O15031	5.9693487275672E-12	0.10000000000597	5.9693487275672E-12	--	--	--
Q14204	5.9693487275672E-12	0.10000000000597	5.9693487275672E-12	--	--	--
Q9HCM2	5.9693487275672E-12	0.10000000000597	5.9693487275672E-12	--	--	--
P54296	5.9943637262436E-12	0.10000000000599	5.9943637262436E-12	--	--	--
Q5VV42	5.9943637262436E-12	0.10000000000599	5.9943637262436E-12	--	--	--
Q8NDG6	5.9943637262436E-12	0.10000000000599	5.9943637262436E-12	--	--	--
A8E7C5	6.0306816822532E-12	0.10000000000603	6.0306816822532E-12	--	--	--
P41252	6.0306816822532E-12	0.10000000000603	6.0306816822532E-12	--	--	--
Q8BU30	6.0306816822532E-12	0.10000000000603	6.0306816822532E-12	--	--	--
Q9P015	0.00018339276131597	0.10018339276132	--	--	--	0.00018339276131597
P48595	0.00091696383087112	0.10091696383087	--	--	--	0.00091696383087112
Q86XP6	0.0010697911360163	0.10106979113602	--	--	--	0.0010697911360163
Q75912	0.0012226184411615	0.10122261844116	--	--	--	0.0012226184411615
Q68D91	0.0014773306164035	0.1014773306164	--	--	--	0.0014773306164035
Q68J44	0.0026490066225166	0.10264900662252	--	--	--	0.0026490066225166
A6NK58	0.0031074885379521	0.10310748853795	--	--	--	0.0031074885379521
Q8NFQ5	0.0034131431482425	0.10341314314824	--	--	--	0.0034131431482425
Q9NRV9	0.0034131431482425	0.10341314314824	--	--	--	0.0034131431482425

**Tabela 6-9 Resultados da busca com dados parciais da proteína P01024**

20080718113727						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
P01024	4.5592593894353E-55	0.031766296895215	4.9870796464417E-10	0.66	4e-13	2.4988755653512e-37
P12387	2.1306941294815E-17	0.056051702547784	4.937621959509E-10	--	1e-09	--
P01027	1.216851466907E-16	0.056051704848756	4.9025956186788E-10	1	6e-09	--
P01026	1.6382686194096E-16	0.056051705773681	4.9870796464417E-10	1	8e-09	--
A7E236	4.0E-13	0.1000000000004	--	--	4e-13	--
Q2PFN7	2.0E-10	0.1000000002	--	--	2e-10	--
P16152	4.8068909224139E-10	0.10000000048069	4.8068909224139E-10	--	--	--
P29475	4.8161129649829E-10	0.10000000048161	4.8161129649829E-10	--	--	--
P48723	4.8161129649829E-10	0.10000000048161	4.8161129649829E-10	--	--	--
Q9UKN8	4.8161129649829E-10	0.10000000048161	4.8161129649829E-10	--	--	--
Q16720	4.8270345554214E-10	0.1000000004827	4.8270345554214E-10	--	--	--
Q8N3T6	4.8270345554214E-10	0.1000000004827	4.8270345554214E-10	--	--	--
Q9NXZ2	4.8270345554214E-10	0.1000000004827	4.8270345554214E-10	--	--	--
P09327	4.8401731614561E-10	0.10000000048402	4.8401731614561E-10	--	--	--
P41090	4.8562800327809E-10	0.10000000048563	4.8562800327809E-10	--	--	--
Q5R6Y0	4.8562800327809E-10	0.10000000048563	4.8562800327809E-10	--	--	--
Q62468	4.8764890172516E-10	0.10000000048765	4.8764890172516E-10	--	--	--
Q9Y450	4.8764890172516E-10	0.10000000048765	4.8764890172516E-10	--	--	--
Q6AXM7	4.937621959509E-10	0.10000000049376	4.937621959509E-10	--	--	--
Q69ZS7	4.9870796464417E-10	0.10000000049871	4.9870796464417E-10	--	--	--
A6MK21	5.0E-9	0.100000005	--	--	5e-09	--
Q80XP1	6.0E-9	0.100000006	--	--	6e-09	--
Q9GKP1	1.0E-7	0.1000001	--	--	1e-07	--
Q3L2T4	3.0E-7	0.1000003	--	--	3e-07	--
Q2UVX4	4.0E-7	0.1000004	--	--	4e-07	--
Q9P015	0.0001199460290553	0.10011994602906	--	--	--	0.0001199460290553
P48595	0.00029486731447686	0.10029486731448	--	--	--	0.00029486731447686
P54136	0.00037982907214916	0.10037982907215	--	--	--	0.00037982907214916
Q9NRB5	0.00079964016192713	0.10079964016193	--	--	--	0.00079964016192713
Q86XP6	0.00089959518216802	0.10089959518217	--	--	--	0.00089959518216802
Q8NFQ5	0.00094957269228847	0.10094957269229	--	--	--	0.00094957269228847
Q6ZSC3	0.00099955020240892	0.10099955020241	--	--	--	0.00099955020240892
Q9BWH6	0.00099955020240892	0.10099955020241	--	--	--	0.00099955020240892
O75912	0.0011994602428907	0.10119946024289	--	--	--	0.0011994602428907
O97940	0.003	0.103	--	--	0.003	--
O97941	0.066	0.166	--	--	0.066	--

Nesta pesquisa, para os dados completos, percebe-se que apenas as informações de seqüência de proteína não seriam suficientes para identificar a proteína correta. Já a submissão de dados parciais permitiu a identificação por qualquer dos programas. Esse resultado mostra, de forma paradoxal, que nem sempre um maior número de informações submetido leva a uma identificação mais exata, porém o uso de sistemas de busca distintos, aumentando a diversidade de dados em vez de sua quantidade, parece ter sido mais eficaz na identificação.

**Proteína 5 - “Cell division topological specificity factor”, cujo código de identificação no banco de dados do NCBI é A6WMJ7**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-10 Resultados da busca com dados completos da proteína A6WMJ7**

20080529180103						
NCBI ID	Consolided Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
A6WMJ7	1.7236128622674E-74	0.01831183235467	7.8770655980114E-10	0.00021	1e-51	9.9194240490595e-17
A9KYY3	1.7236128622674E-74	0.01831183235467	7.8770655980114E-10	0.00021	1e-51	9.9194240490595e-17
A1RK91	1.3967081088049E-73	0.018322241398738	4.7856512361083E-10	0.00029	1e-50	9.9194240490595e-17
A4Y6A5	1.3967081088049E-73	0.018322241398738	4.7856512361083E-10	0.00029	1e-50	9.9194240490595e-17
Q8EE12	1.3967081088049E-73	0.018322241398738	4.7856512361083E-10	0.00029	1e-50	9.9194240490595e-17
Q0HI69	8.17082630912E-73	0.01833264777464	4.5207130426087E-10	0.00037	5e-50	9.9194240490595e-17
Q0HUG5	8.17082630912E-73	0.01833264777464	4.5207130426087E-10	0.00037	5e-50	9.9194240490595e-17
A3D3Q9	1.8083712318826E-60	0.031816384682086	7.8770655980114E-10	0.00021	1e-51	--
A0KXU4	3.7547963843198E-26	0.031854526270858	4.5207130426087E-10	0.00037	--	9.9194240490595e-17
Q083I0	5.2642439964931E-11	0.058022809982074	4.3723437758456E-10	0.0043	--	--
A2V4U9	1.0E-50	0.1	--	--	1e-50	--
A5NKRO	1.0E-51	0.1	--	--	1e-51	--
A3D4J1	4.3450528528457E-10	0.10000000043451	4.3450528528457E-10	--	--	--
A6WNG4	4.3450528528457E-10	0.10000000043451	4.3450528528457E-10	--	--	--
A3D7K5	4.3453348306988E-10	0.10000000043453	4.3453348306988E-10	--	--	--
A6WRG7	4.3453348306988E-10	0.10000000043453	4.3453348306988E-10	--	--	--
A3DA75	4.3476992907336E-10	0.10000000043477	4.3476992907336E-10	--	--	--
A6WHR3	4.3476992907336E-10	0.10000000043477	4.3476992907336E-10	--	--	--
A9KW87	4.3476992907336E-10	0.10000000043477	4.3476992907336E-10	--	--	--
Q03974	6.027957678074E-6	0.10000602795768	--	--	--	6.027957678074e-06
A5EWQ0	6.4857772543177E-6	0.10000648577725	--	--	--	6.4857772543177e-06
P52138	6.4857772543177E-6	0.10000648577725	--	--	--	6.4857772543177e-06
O52376	6.7909900019449E-6	0.10000679099	--	--	--	6.7909900019449e-06
Q1QSR9	8.393358746199E-6	0.10000839335875	--	--	--	8.393358746199e-06
Q39ZS1	9.9194233939406E-6	0.10000991942339	--	--	--	9.9194233939406e-06
Q3KFM0	9.9194233939406E-6	0.10000991942339	--	--	--	9.9194233939406e-06
Q7YGC1	1.0682456172614E-5	0.10001068245617	--	--	--	1.0682456172614e-05
Q0CB04	1.3734586377703E-5	0.10001373458638	--	--	--	1.3734586377703e-05
Q39V99	1.4497619156377E-5	0.10001449761916	--	--	--	1.4497619156377e-05
POA736	1.9838846787881E-5	0.10001983884679	--	--	--	1.9838846787881e-05
Q0TIK2	1.9838846787881E-5	0.10001983884679	--	--	--	1.9838846787881e-05

**Tabela 6-11 Resultados da busca com dados parciais da proteína A6WMJ7**

20080718113849						
NCBI ID	Consolided Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
A6WMJ7	3.3952304696275E-29	0.026358180124426	6.3757550572992E-10	0.068	2e-13	4.9817334762691e-11
A9KYY3	3.3952304696275E-29	0.026358180124426	6.3757550572992E-10	0.068	2e-13	4.9817334762691e-11
A3D3Q9	3.3952304696275E-29	0.026358180124426	6.3757550572992E-10	0.068	2e-13	4.9817334762691e-11
A0KXU4	5.6176665196381E-28	0.031766296906782	4.8877141813082E-10	0.2	9e-12	4.9817334762691e-11
Q0HUG5	5.6176665196381E-28	0.031766296906782	4.8877141813082E-10	0.2	9e-12	4.9817334762691e-11
A1RK91	1.3225588356503E-28	0.031766296907065	4.96958262508E-10	0.12	2e-12	4.9817334762691e-11
A4Y6A5	1.3225588356503E-28	0.031766296907065	4.96958262508E-10	0.12	2e-12	4.9817334762691e-11
Q8EE12	1.3225588356503E-28	0.031766296907065	4.96958262508E-10	0.12	2e-12	4.9817334762691e-11
Q0HI69	1.124006642982E-18	0.05605170210791	4.8877141813082E-10	0.2	--	4.9817334762691e-11
A5NKRO	2.0E-13	0.1000000000002	--	--	2e-13	--
A2V4U9	2.0E-12	0.1000000000002	--	--	2e-12	--
A3D4J1	4.7433921555755E-10	0.10000000047434	4.7433921555755E-10	--	--	--
A6WNG4	4.7433921555755E-10	0.10000000047434	4.7433921555755E-10	--	--	--
A3D7K5	4.7435615654633E-10	0.10000000047436	4.7435615654633E-10	--	--	--
A6WRG7	4.7435615654633E-10	0.10000000047436	4.7435615654633E-10	--	--	--
A9KW87	4.7468877418463E-10	0.10000000047469	4.7468877418463E-10	--	--	--
A3DA75	4.7471434819666E-10	0.10000000047471	4.7471434819666E-10	--	--	--
A6WHR3	4.7471434819666E-10	0.10000000047471	4.7471434819666E-10	--	--	--
Q083I0	4.7770006439941E-10	0.1000000004777	4.7770006439941E-10	0.22	--	--
O33406	4.9062526505702E-6	0.10000490625265	--	--	--	4.9062526505702e-06

Nesta pesquisa, foram encontradas duas proteínas homólogas (A3D3Q9 e A6WMJ7), que não recebem a mesma identificação no NCBI por serem de subespécies diferentes. Esta diferenciação da taxonomia (até o nível de subespécie) não é realizada em nenhum programa de identificação de forma a interferir na identificação. Pode-se perceber que a utilização de mais programas foi imprescindível para a correta identificação da proteína em questão. Caso fossem utilizados qualquer um dos programas isoladamente, seria impossível realizar corretamente identificação.

**Proteína 6 - “Acyl-coenzyme A dehydrogenase (ACDH)”, cujo código de identificação no banco de dados do NCBI é Q8Z937**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-12 Resultados da busca com dados completos da proteína Q8Z937**

20080529180117						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
Q8Z937	0	0.018284495519192	2.2526438198845E-11	5.6e-136	0.0	2.5539756107837e-45
Q8ZRJ7	0	0.018284495519192	2.2526438198845E-11	1e-135	0.0	3.2351438595106e-42
Q47146	0	0.031766296779567	1.4385052555089E-11	9.7e-131	0.0	--
Q8ZBY6	2.8934031754074E-120	0.056051701866183	1.3685766034326E-11	7.5e-112	--	--
Q8XR2	8.4078271942846E-139	0.056051701866505	1.4385052555089E-11	1.8e-130	--	--
A7ZHY0	0	0.1	--	--	0.0	--
A7ZWI5	0	0.1	--	--	0.0	--
ASMNS5	0	0.1	--	--	0.0	--
ASMY17	0	0.1	--	--	0.0	--
B1LHM7	0	0.1	--	--	0.0	--
Q57SU5	0	0.1	--	--	0.0	--
Q5PF88	0	0.1	--	--	0.0	--
P96142	3.2E-17	0.1	--	3.2e-17	--	--
Q5SJA5	3.2E-17	0.1	--	3.2e-17	--	--
Q72JG7	2.6E-17	0.1	--	2.6e-17	--	--
Q21MG1	1.1E-16	0.1	--	1.1e-16	--	--
Q70LM4	1.1E-16	0.1	--	1.1e-16	--	--
P37412	1.2621190038163E-11	0.10000000001262	1.2621190038163E-11	--	--	--
Q8ZNA7	1.2621190038163E-11	0.10000000001262	1.2621190038163E-11	--	--	--
P55912	1.2636923736166E-11	0.10000000001264	1.2636923736166E-11	--	--	--
Q8Z429	1.2654929123613E-11	0.10000000001265	1.2654929123613E-11	--	--	--
Q8Z8X3	1.2654929123613E-11	0.10000000001265	1.2654929123613E-11	--	--	--
Q8ZMC6	1.2654929123613E-11	0.10000000001265	1.2654929123613E-11	--	--	--
Q8ZRD1	1.2654929123613E-11	0.10000000001265	1.2654929123613E-11	--	--	--
Q8ZRJ0	1.2675736451369E-11	0.10000000001268	1.2675736451369E-11	--	--	--
Q23729	1.2859266121986E-11	0.10000000001286	1.2859266121986E-11	--	--	--
Q1D4L3	1.2859266121986E-11	0.10000000001286	1.2859266121986E-11	--	--	--
Q39HY9	1.2859266121986E-11	0.10000000001286	1.2859266121986E-11	--	--	--
Q6D2L7	1.2859266121986E-11	0.10000000001286	1.2859266121986E-11	--	--	--
P07314	1.2928109040345E-11	0.10000000001293	1.2928109040345E-11	--	--	--
Q57LE6	0.00059594755661502	0.10059594755662	--	--	--	0.00059594755661502
Q5PIG8	0.00059594755661502	0.10059594755662	--	--	--	0.00059594755661502
Q8Z4L6	0.00061297461342288	0.10061297461342	--	--	--	0.00061297461342288
P74881	0.00071513703546146	0.10071513703546	--	--	--	0.00071513703546146
P26984	0.0012600034216529	0.10126000342165	--	--	--	0.0012600034216529
P43669	0.0013451387868836	0.10134513878688	--	--	--	0.0013451387868836
Q8ZRW3	0.0013451387868836	0.10134513878688	--	--	--	0.0013451387868836
ASMPP3	0.0015835178257679	0.10158351782577	--	--	--	0.0015835178257679

**Tabela 6-13 Resultados da busca com dados parciais da proteína Q8Z937**

20080718114014						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
Q8Z937	3.2991427684006E-41	0.031766297132468	4.9352374554609E-10	0.22	1e-09	1.2653956068823e-26
Q8ZRJ7	1.9514283957169E-39	0.031766297135439	5.0597519228999E-10	0.22	1e-09	7.9298124697958e-25
Q47146	2.0784793279509E-17	0.056051702542081	4.8137871407388E-10	0.22	1e-09	--
Q8X7R2	2.0784793279509E-17	0.056051702542081	4.8137871407388E-10	0.22	1e-09	--
A6T517	3.0E-10	0.1000000003	--	--	3e-10	--
A8AKQ7	3.0E-10	0.1000000003	--	--	3e-10	--
A9MNS5	3.0E-10	0.1000000003	--	--	3e-10	--
A9MY17	3.0E-10	0.1000000003	--	--	3e-10	--
Q57SU5	3.0E-10	0.1000000003	--	--	3e-10	--
P37412	4.6610419782712E-10	0.1000000004661	4.6610419782712E-10	--	--	--
Q8Z429	4.6661685144044E-10	0.10000000046662	4.6661685144044E-10	--	--	--
Q8Z4Z0	4.6661685144044E-10	0.10000000046662	4.6661685144044E-10	--	--	--
Q8ZNA7	4.6661685144044E-10	0.10000000046662	4.6661685144044E-10	--	--	--
POA2K1	4.6719854053646E-10	0.1000000004672	4.6719854053646E-10	--	--	--
POA2K2	4.6719854053646E-10	0.1000000004672	4.6719854053646E-10	--	--	--
Q8ZMC6	4.6719854053646E-10	0.1000000004672	4.6719854053646E-10	--	--	--
Q8ZRJ0	4.678642160531E-10	0.10000000046786	4.678642160531E-10	--	--	--
Q83QQ0	4.7059732521725E-10	0.1000000004706	4.7059732521725E-10	--	--	--
P55584	4.7187823439369E-10	0.10000000047188	4.7187823439369E-10	--	--	--
Q92Q15	4.7187823439369E-10	0.10000000047188	4.7187823439369E-10	--	--	--
Q2W4T2	4.7344852573427E-10	0.10000000047345	4.7344852573427E-10	--	--	--
Q3J716	4.7344852573427E-10	0.10000000047345	4.7344852573427E-10	--	--	--
Q8ZBY6	4.8620044362951E-10	0.1000000004862	4.8620044362951E-10	0.85	--	--
A1A7T4	1.0E-9	0.100000001	--	--	1e-09	--
Q3Z5B5	1.0E-9	0.100000001	--	--	1e-09	--
Q5PF88	1.0E-9	0.100000001	--	--	1e-09	--
Q83M90	1.0E-9	0.100000001	--	--	1e-09	--
Q8FKN6	1.0E-9	0.100000001	--	--	1e-09	--
Q8KTJ8	1.0E-9	0.100000001	--	--	1e-09	--
A7MI46	5.0E-9	0.100000005	--	--	5e-09	--
A4W6W4	2.0E-8	0.10000002	--	--	2e-08	--
A2X158	2.0E-7	0.1000002	--	--	2e-07	--
Q7N7F8	7.0E-5	0.10007	--	--	7e-05	--
P26984	0.001147292085496	0.1011472920855	--	--	--	0.001147292085496
A9MQG6	0.0011979078631066	0.10119790786311	--	--	--	0.0011979078631066
A9MYL9	0.0021933524548676	0.10219335245487	--	--	--	0.0021933524548676
Q5PIL3	0.0021933524548676	0.10219335245487	--	--	--	0.0021933524548676
Q8ZRW2	0.0021933524548676	0.10219335245487	--	--	--	0.0021933524548676

Nesta pesquisa, foram encontradas duas proteínas homólogas (Q8ZRJ7 e Q8Z937), situação semelhante à descrita anteriormente. Pode-se perceber, ao se analisar os resultados obtidos com dados parciais e completos, que a utilização apenas de dados de sequência da proteína seria insuficiente para distinguir entre os resultados.

**Proteína 7 - “Chloroplast 30S ribosomal protein S17”, cujo código de identificação no banco de dados do NCBI é O46903**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-14 Resultados da busca com dados completos da proteína O46903**

20080529180145						
NCBI ID	Consolided Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
O46903	9.9013303827867E-64	0.031828305809573	6.3016524784091E-10	0.00026	5e-55	--
A0ZG66	2.0E-28	0.1	--	--	2e-28	--
A3IQ12	2.0E-27	0.1	--	--	2e-27	--
A6MW10	1.0E-36	0.1	--	--	1e-36	--
B1U7C6	5.0E-28	0.1	--	--	5e-28	--
B1WQR9	1.0E-27	0.1	--	--	1e-27	--
Q2JIL8	1.0E-27	0.1	--	--	1e-27	--
Q3MFB2	9.0E-29	0.1	--	--	9e-29	--
Q6H092	9.0E-28	0.1	--	--	9e-28	--
Q8YPI8	9.0E-29	0.1	--	--	9e-29	--
O78413	3.4715913876475E-10	0.10000000034716	3.4715913876475E-10	--	--	--
O46908	3.4716292851097E-10	0.10000000034716	3.4716292851097E-10	--	--	--
O78487	3.471992247144E-10	0.1000000003472	3.471992247144E-10	--	--	--
O46898	3.4723048958687E-10	0.10000000034723	3.4723048958687E-10	--	--	--
O78423	3.4731432423321E-10	0.10000000034731	3.4731432423321E-10	--	--	--
O46895	3.4734482574368E-10	0.10000000034734	3.4734482574368E-10	--	--	--
O46896	3.4735770496346E-10	0.10000000034736	3.4735770496346E-10	--	--	--
O78429	3.4745254719669E-10	0.10000000034745	3.4745254719669E-10	--	--	--
O46905	3.4753211679006E-10	0.10000000034753	3.4753211679006E-10	--	--	--
P92959	3.4832545406483E-10	0.10000000034833	3.4832545406483E-10	--	--	--
Q5UPF5	3.4835695860828E-10	0.10000000034836	3.4835695860828E-10	--	--	--
P47396	3.4838959131791E-10	0.10000000034839	3.4838959131791E-10	--	--	--
Q74MB7	3.4838959131791E-10	0.10000000034839	3.4838959131791E-10	--	--	--
Q461S0	3.4842341389719E-10	0.10000000034842	3.4842341389719E-10	--	--	--
P48154	3.4853270928834E-10	0.10000000034853	3.4853270928834E-10	--	--	--
P57589	3.4865542862768E-10	0.10000000034866	3.4865542862768E-10	--	--	--
Q0AUI9	3.4874597270836E-10	0.10000000034875	3.4874597270836E-10	--	--	--
Q318J5	3.4907079969313E-10	0.10000000034907	3.4907079969313E-10	--	--	--
P87292	3.2475029449734E-6	0.10000324750294	--	--	--	3.2475029449734e-06
Q6DYE7	4.5465042089922E-6	0.10000454650421	--	--	--	4.5465042089922e-06
Q84JP1	4.9795046303318E-6	0.10000497950463	--	--	--	4.9795046303318e-06
P11854	5.2681717212739E-6	0.10000526817172	--	--	--	5.2681717212739e-06
Q9DGA4	6.7836729808889E-6	0.10000678367298	--	--	--	6.7836729808889e-06
P25380	8.660008426792E-6	0.10000866000843	--	--	--	8.660008426792e-06
Q5X129	1.1546677615625E-5	0.10001154667762	--	--	--	1.1546677615625e-05
Q8BTY2	1.1546677615625E-5	0.10001154667762	--	--	--	1.1546677615625e-05
Q8N684	1.1546677615625E-5	0.10001154667762	--	--	--	1.1546677615625e-05
Q8WXC6	1.2268345127906E-5	0.10001226834513	--	--	--	1.2268345127906e-05

**Tabela 6-15 Resultados da busca com dados completos da proteína O46903**

20080718114144						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
O46903	5.7494693541265E-17	0.038643803486673	6.2148431644219E-10	0.03	3e-09	--
O78487	4.6173502092941E-10	0.10000000046174	4.6173502092941E-10	--	--	--
O78413	4.6176321315884E-10	0.10000000046176	4.6176321315884E-10	--	--	--
O46908	4.6178102556062E-10	0.10000000046178	4.6178102556062E-10	--	--	--
O46898	4.6187447608309E-10	0.10000000046187	4.6187447608309E-10	--	--	--
O46896	4.6194535881493E-10	0.10000000046195	4.6194535881493E-10	--	--	--
O78423	4.619801690131E-10	0.10000000046198	4.619801690131E-10	--	--	--
O46895	4.6202697677542E-10	0.10000000046203	4.6202697677542E-10	--	--	--
O78429	4.6214523083605E-10	0.10000000046215	4.6214523083605E-10	--	--	--
O46905	4.6226155980446E-10	0.10000000046226	4.6226155980446E-10	--	--	--
Q5UPQ8	4.6324952881576E-10	0.10000000046325	4.6324952881576E-10	--	--	--
Q74MB7	4.6329142774888E-10	0.10000000046329	4.6329142774888E-10	--	--	--
P47396	4.6342646120251E-10	0.10000000046343	4.6342646120251E-10	--	--	--
P92959	4.6342646120251E-10	0.10000000046343	4.6342646120251E-10	--	--	--
Q0AUI9	4.6352516438449E-10	0.10000000046353	4.6352516438449E-10	--	--	--
P57589	4.6363178743348E-10	0.10000000046363	4.6363178743348E-10	--	--	--
Q46IS0	4.6387293865683E-10	0.10000000046387	4.6387293865683E-10	--	--	--
P48154	4.6393994736514E-10	0.10000000046394	4.6393994736514E-10	--	--	--
Q318J5	4.6471124359362E-10	0.10000000046471	4.6471124359362E-10	--	--	--
O60111	1.8456994524183E-5	0.10001845699452	--	--	--	1.8456994524183e-05
O15417	2.2716302124569E-5	0.10002271630212	--	--	--	2.2716302124569e-05
Q80WC3	2.2716302124569E-5	0.10002271630212	--	--	--	2.2716302124569e-05
Q03649	2.6975608032458E-5	0.10002697560803	--	--	--	2.6975608032458e-05
P02789	3.194479938666E-5	0.10003194479939	--	--	--	3.194479938666e-05
P16243	5.8920409111614E-5	0.10005892040911	--	--	--	5.8920409111614e-05
P02675	6.7439020927393E-5	0.10006743902093	--	--	--	6.7439020927393e-05
Q14574	7.0988443081466E-5	0.10007098844308	--	--	--	7.0988443081466e-05
Q5N7W4	8.518613169776E-5	0.1000851861317	--	--	--	8.518613169776e-05
Q5SP85	8.518613169776E-5	0.1000851861317	--	--	--	8.518613169776e-05
A7WJC5	0.0004	0.1004	--	--	4e-04	--
Q11HR1	0.0007	0.1007	--	--	7e-04	--
A4XLS1	0.001	0.101	--	--	0.001	--
A9B420	0.001	0.101	--	--	0.001	--
O66439	0.002	0.102	--	--	0.002	--
Q7MYG0	0.002	0.102	--	--	0.002	--
A5GYW9	0.003	0.103	--	--	0.003	--
Q4C108	0.003	0.103	--	--	0.003	--
Q8R7W3	0.003	0.103	--	--	0.003	--

Neste caso observa-se que tanto as buscas individuais quanto a busca consolidada foram capazes de identificar corretamente a proteína, quer seja com dados completos ou parciais, exceto pelos dados de PMF.

**Proteína 8 - “Bcl-2-associated transcription factor 1 (Btf)”, cujo código de identificação no banco de dados do NCBI é Q8K019**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-16 Resultados da busca com dados completos da proteína Q8K019**

20080529180201						
NCBI ID	Consolided Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
Q8K019	0	0.018284495518826	1.9713981586623E-11	1.7e-122	0.0	6.2832595855193e-39
Q9NYF8	0	0.031766296779623	1.4619088542989E-11	9.7e-118	0.0	--
Q569Z6	4.6568785687092E-37	0.056051701864999	1.1114435688747E-11	4.7e-28	--	--
Q5BJ39	6.8072484118766E-37	0.056051701864999	1.1114435688747E-11	6.9e-28	--	--
Q5M7V8	9.8728592675966E-37	0.056051701865025	1.1169960861911E-11	1e-27	--	--
Q9Y2W1	1.2828857947084E-43	0.056051701865025	1.1169960861911E-11	1.1e-34	--	--
A2RU75	0	0.1	--	--	0.0	--
A5D7B8	0	0.1	--	--	0.0	--
B1WC16	0	0.1	--	--	0.0	--
Q3UDL9	0	0.1	--	--	0.0	--
Q3UR37	0	0.1	--	--	0.0	--
Q8BT8	1.3E-19	0.1	--	1.3e-19	--	--
Q9UQ35	7.8E-19	0.1	--	7.8e-19	--	--
Q7Z6E9	2.0E-17	0.1	--	2e-17	--	--
P20930	7.5E-17	0.1	--	7.5e-17	--	--
Q61136	1.0902254673796E-11	0.1000000000109	1.0902254673796E-11	--	--	--
Q8C5N3	1.0902254673796E-11	0.1000000000109	1.0902254673796E-11	--	--	--
Q9D0F4	1.0902254673796E-11	0.1000000000109	1.0902254673796E-11	--	--	--
A2AR02	1.0918375077703E-11	0.10000000001092	1.0918375077703E-11	--	--	--
A2AJT4	1.0924355825301E-11	0.10000000001092	1.0924355825301E-11	--	--	--
Q80Z37	1.0924355825301E-11	0.10000000001092	1.0924355825301E-11	--	--	--
Q8TF01	1.0926560084355E-11	0.10000000001093	1.0926560084355E-11	--	--	--
Q8K2H1	1.0936679446821E-11	0.10000000001094	1.0936679446821E-11	--	--	--
P30415	1.095327451523E-11	0.10000000001095	1.095327451523E-11	--	--	--
Q5M9Q1	1.0996380843736E-11	0.100000000011	1.0996380843736E-11	--	--	--
Q8BL66	8.3349358999569E-5	0.100083349359	--	--	--	8.3349358999569e-05
P35601	0.00076937872667821	0.10076937872668	--	--	--	0.00076937872667821
Q6NSR3	0.0010899531961275	0.10108995319613	--	--	--	0.0010899531961275
P84075	0.0016028723472463	0.10160287234725	--	--	--	0.0016028723472463
Q8BH34	0.0018593319228057	0.10185933192281	--	--	--	0.0018593319228057
Q91ZU8	0.0023081361800346	0.10230813618003	--	--	--	0.0023081361800346
Q3ULZ2	0.0023722510739245	0.10237225107392	--	--	--	0.0023722510739245
Q6AW69	0.0023722510739245	0.10237225107392	--	--	--	0.0023722510739245
Q9DA08	0.0023722510739245	0.10237225107392	--	--	--	0.0023722510739245

**Tabela 6-17 Resultados da busca com dados parciais da proteína Q8K019**

20080727192621						
<u>NCBI ID</u>	<u>Consolidated Evalue</u>	<u>PL Score</u>	<u>AACompident Evalue</u>	<u>FASTA Evalue</u>	<u>Blast Evalue</u>	<u>Mascot Evalue</u>
Q8K019	2.2239955314127E-35	0.031767012640685	4.6696562266869E-10	0.36	3e-06	4.0010162093168e-24
Q9NYF8	4.9313900486724E-14	0.056053083621487	4.6696562266869E-10	0.36	3e-06	--
Q80SY5	4.5212036491425E-10	0.10000000045212	4.5212036491425E-10	--	--	--
Q7TQC7	4.5215987102039E-10	0.10000000045216	4.5215987102039E-10	--	--	--
Q80Z37	4.5233256785049E-10	0.10000000045233	4.5233256785049E-10	--	--	--
Q9D0E3	4.5242890819248E-10	0.10000000045243	4.5242890819248E-10	--	--	--
Q8K2H1	4.5264574899161E-10	0.10000000045265	4.5264574899161E-10	--	--	--
Q60948	4.5270574458031E-10	0.10000000045271	4.5270574458031E-10	--	--	--
A2AJT4	4.5290214938402E-10	0.1000000004529	4.5290214938402E-10	--	--	--
Q05519	4.5312740118283E-10	0.10000000045313	4.5312740118283E-10	--	--	--
P30415	4.5321002157571E-10	0.10000000045321	4.5321002157571E-10	--	--	--
Q5M9Q1	4.5434067790385E-10	0.10000000045434	4.5434067790385E-10	--	--	--
Q569Z6	4.6946555353044E-10	0.10000000046947	4.6946555353044E-10	--	--	--
Q5BJ39	4.6946555353044E-10	0.10000000046947	4.6946555353044E-10	--	--	--
Q5M7V8	4.7755562335123E-10	0.10000000047756	4.7755562335123E-10	--	--	--
Q9Y2W1	4.8474871430287E-10	0.10000000048475	4.8474871430287E-10	--	--	--
B1WC16	3.0E-6	0.100003	--	--	3e-06	--
Q05C67	3.0E-6	0.100003	--	--	3e-06	--
Q3TRC6	3.0E-6	0.100003	--	--	3e-06	--
Q3TSZ0	3.0E-6	0.100003	--	--	3e-06	--
Q3UDL9	3.0E-6	0.100003	--	--	3e-06	--
Q3UR37	3.0E-6	0.100003	--	--	3e-06	--
Q5RBF8	3.0E-6	0.100003	--	--	3e-06	--
Q7TP73	3.0E-6	0.100003	--	--	3e-06	--
Q8BL66	0.00037469834214197	0.10037469834214	--	--	--	0.00037469834214197
Q9JKK8	0.00069859011812524	0.10069859011813	--	--	--	0.00069859011812524
Q9QY06	0.00069859011812524	0.10069859011813	--	--	--	0.00069859011812524
Q8BZ05	0.0009526228883526	0.10095262288835	--	--	--	0.0009526228883526
Q8VCC8	0.0009526228883526	0.10095262288835	--	--	--	0.0009526228883526
Q9D0C3	0.0011431474660231	0.10114314746602	--	--	--	0.0011431474660231
Q8VEB1	0.00120665565858	0.10120665565858	--	--	--	0.00120665565858
Q6NSR3	0.0014606884288073	0.10146068842881	--	--	--	0.0014606884288073
Q9JLV6	0.0016512130064778	0.10165121300648	--	--	--	0.0016512130064778

Nesta pesquisa, pode-se perceber, ao se analisar os resultados obtidos com dados parciais e completos, que caso fosse utilizado apenas o programa Blast, o que é geralmente feito nas pesquisas por seqüência de proteínas, seria impossível distinguir-se a proteína correta das demais. Portanto, a utilização de diferentes programas proporcionou o acerto na pesquisa.

**Proteína 9 - “30S ribosomal protein S20”, cujo código de identificação no banco de dados do NCBI é A1BAN4**

A pesquisa para identificação desta proteína resultou nas seguintes informações:

**Tabela 6-18 Resultados da busca com dados completos da proteína A1BAN4**

20080529180210						
NCBI ID	Consolided Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
A1BAN4	4.3618568378498E-79	0.018353452455492	6.5642213316761E-10	0.00053	2e-53	5.0360150114142e-20
Q28JI3	8.5914225672949E-16	0.043174467456074	3.6471434199455E-10	0.051	--	5.112318241335e-08
Q16DK7	1.8579396726805E-15	0.044848727652656	3.6514848448668E-10	0.059	--	9.9194239624442e-08
A8LNL0	2.5155796239788E-40	0.067713136659311	--	0.026	1e-40	--
Q1GC53	4.3529254855658E-10	0.076310312980202	3.6669884295326E-10	0.046	--	--
A3PFL4	5.8789879710214E-8	0.076310350894363	--	0.046	--	6.1805641940135e-08
Q3IY62	5.8789879710214E-8	0.076310350894363	--	0.046	--	6.1805641940135e-08
A4WNE9	8.0488324084038E-8	0.083811814348368	--	0.064	--	6.1805641940135e-08
A3JM09	6.0E-43	0.1	--	--	6e-43	--
A3K329	7.0E-41	0.1	--	--	7e-41	--
A3SKA5	2.0E-40	0.1	--	--	2e-40	--
A3TY06	8.0E-42	0.1	--	--	8e-42	--
A3VCU9	8.0E-43	0.1	--	--	8e-43	--
A6E0Y0	6.0E-40	0.1	--	--	6e-40	--
A6FKX7	3.0E-40	0.1	--	--	3e-40	--
Q0FEB1	8.0E-40	0.1	--	--	8e-40	--
A1B022	3.6176781357726E-10	0.10000000036177	3.6176781357726E-10	--	--	--
A1B0F7	3.6177370515989E-10	0.10000000036177	3.6177370515989E-10	--	--	--
A1B8V3	3.6179191611281E-10	0.10000000036179	3.6179191611281E-10	--	--	--
A1B051	3.617981719952E-10	0.1000000003618	3.617981719952E-10	--	--	--
P29909	3.6183781178004E-10	0.10000000036184	3.6183781178004E-10	--	--	--
A1B4S6	3.6187387674582E-10	0.10000000036187	3.6187387674582E-10	--	--	--
A1B8N9	3.6197407606828E-10	0.10000000036197	3.6197407606828E-10	--	--	--
A1B4C1	3.6202272651858E-10	0.10000000036202	3.6202272651858E-10	--	--	--
A1B053	3.6213574634404E-10	0.10000000036214	3.6213574634404E-10	--	--	--
Q2IQW5	3.6484866424602E-10	0.10000000036485	3.6484866424602E-10	--	--	--
A9HI31	3.6499299139695E-10	0.10000000036499	3.6499299139695E-10	--	--	--
Q2G300	3.6499299139695E-10	0.10000000036499	3.6499299139695E-10	--	--	--
Q2N7Y7	3.6514848448668E-10	0.10000000036515	3.6514848448668E-10	--	--	--
Q0BYU4	3.6569661821093E-10	0.1000000003657	3.6569661821093E-10	--	--	--
Q5FN16	3.6591277552552E-10	0.10000000036591	3.6591277552552E-10	--	--	--
A5G879	1.2208522127914E-7	0.10000012208522	--	--	--	1.2208522127914e-07
Q9CPM8	1.6023684315772E-7	0.10000016023684	--	--	--	1.6023684315772e-07
Q62G98	4.8071054368574E-7	0.10000048071054	--	--	--	4.8071054368574e-07
Q63Q46	4.8071054368574E-7	0.10000048071054	--	--	--	4.8071054368574e-07
Q0K8N7	9.1563909563691E-7	0.1000009156391	--	--	--	9.1563909563691e-07
Q31FR9	1.4497619156377E-6	0.10000144976192	--	--	--	1.4497619156377e-06
Q1LTC2	1.9838846787881E-6	0.10000198388468	--	--	--	1.9838846787881e-06
A6TCH9	2.1364912800032E-6	0.10000213649128	--	--	--	2.1364912800032e-06

**Tabela 6-19 Resultados da busca com dados parciais da proteína A1BAN4**

20080718114414						
NCBI ID	Consolidated Evalue	PL Score	AACompident Evalue	FASTA Evalue	Blast Evalue	Mascot Evalue
A1BAN4	4.5990932402841E-15	0.056051789058543	6.4749191035806E-10	0.36	0.38	1.8870203475536e-07
A1AXX8	4.8121051200166E-10	0.10000000048121	4.8121051200166E-10	--	--	--
A1B8Y3	4.812944840468E-10	0.10000000048129	4.812944840468E-10	--	--	--
P29910	4.812944840468E-10	0.10000000048129	4.812944840468E-10	--	--	--
A1B051	4.8132234142492E-10	0.10000000048132	4.8132234142492E-10	--	--	--
P29909	4.8137192210298E-10	0.10000000048137	4.8137192210298E-10	--	--	--
A1B4S6	4.8143727015414E-10	0.10000000048144	4.8143727015414E-10	--	--	--
A1B8N9	4.8159152672975E-10	0.10000000048159	4.8159152672975E-10	--	--	--
A1B4C1	4.8163606359304E-10	0.10000000048164	4.8163606359304E-10	--	--	--
A1B053	4.8176953203796E-10	0.10000000048177	4.8176953203796E-10	--	--	--
Q2G300	4.8512987794605E-10	0.10000000048513	4.8512987794605E-10	--	--	--
Q2N7Y7	4.8530854856566E-10	0.10000000048531	4.8530854856566E-10	--	--	--
Q0BVU4	4.8593083542868E-10	0.10000000048593	4.8593083542868E-10	--	--	--
Q16DK7	4.8593083542868E-10	0.10000000048593	4.8593083542868E-10	--	--	--
A9HI31	4.8643646638004E-10	0.10000000048644	4.8643646638004E-10	--	--	--
Q28JI3	4.8643646638004E-10	0.10000000048644	4.8643646638004E-10	--	--	--
Q2IQW5	4.8672399105228E-10	0.10000000048672	4.8672399105228E-10	--	--	--
Q5FN16	4.8703909403094E-10	0.10000000048704	4.8703909403094E-10	--	--	--
Q1GC53	4.8867347939001E-10	0.10000000048867	4.8867347939001E-10	--	--	--
A1VE89	4.9817335713238E-5	0.10004981733571	--	--	--	4.9817335713238e-05
Q72CF3	4.9817335713238E-5	0.10004981733571	--	--	--	4.9817335713238e-05
Q0BMC8	5.7365416990977E-5	0.10005736541699	--	--	--	5.7365416990977e-05
Q7NXI6	6.0384650221914E-5	0.10006038465022	--	--	--	6.0384650221914e-05
P74849	6.4913501867922E-5	0.10006491350187	--	--	--	6.4913501867922e-05
Q483C7	6.4913501867922E-5	0.10006491350187	--	--	--	6.4913501867922e-05
Q87E14	6.4913501867922E-5	0.10006491350187	--	--	--	6.4913501867922e-05
Q9PEI5	6.4913501867922E-5	0.10006491350187	--	--	--	6.4913501867922e-05
A8AL30	7.2461583145661E-5	0.10007246158315	--	--	--	7.2461583145661e-05

Neste caso observa-se uma situação semelhante à anterior, em que tanto as buscas individuais quanto a busca consolidada foram capazes de identificar corretamente a proteína, quer seja com dados completos ou parciais.

Na grande maioria dos experimentos (6 experimentos de 9), se fosse utilizado apenas um programa para a identificação, seria impossível distinguir a proteína procurada dentre os resultados apresentados, uma vez que o valor utilizado como score indexador para identificação pelos referidos programas apresentava os mesmos valores para diversas proteínas candidatas. Em nenhum caso foram obtidos resultados falso-positivos como identificação. O sistema mostrou-se eficiente tanto com os dados completos (experimentos teóricos) quanto com os dados parciais (simulação de experimentos reais). Em todos os experimentos realizados, a maior demora para obtenção de resultados ocorreu em função do programa AACompident, pois este só oferece a exibição dos resultados por *e-mail*. Esse fato requer a execução de dois robôs para cada busca realizada, além do tempo de espera para o envio e o recebimento do *e-mail* com os resultados.

## 7. CONCLUSÕES E RECOMENDAÇÕES

Após o desenvolvimento do sistema, que possibilita o armazenamento, organização e disponibilização dos dados experimentais dos cientistas, foram realizados vários testes que comprovaram a eficiência do Protein Locator. A metodologia de teste abordou amostras de proteínas ao longo de toda a faixa de pI e diferentes valores de peso molecular. As etapas de suporte aos experimentos (armazenamento e disponibilização das informações) e de combinação dos resultados foram bem sucedida. Em todos os casos pode-se perceber a melhora nos resultados com a adição de mais programas de identificação, em relação ao uso de programas isolados para identificação de proteínas (situação muito comum em química de proteínas). Esta comparação pode ser feita observando-se os resultados em separado de uma pesquisa (disponível para o cientista) e dos resultados consolidados.

Outra funcionalidade disponibilizada no sistema foi a consolidação dos resultados por meio do PLscore, um algoritmo desenvolvido pela equipe para possibilitar a diferenciação de resultados que possuem e-valores nulos dentre que são consolidados, tratamento que não é feito por nenhum outro programa avaliado.

Para ampliar as funcionalidades do sistema, permitindo uma melhora na qualidade dos resultados, aumento da velocidade das buscas e qualidade do código fonte do sistema, as sugestões para trabalhos futuros são:

- Avaliar o efeito no cálculo do e-valor consolidado nos casos em que a proteína é encontrada em mais de um programa, porém com e-valor mais alto do que as que são encontradas por apenas um programa com um e-valor muito baixo.
- Realizar pré-processamento dos dados antes de submetê-los aos programas de identificação, de acordo com as informações fornecidas pelo cientista. Para informações de *fingerprint*, pode-se efetuar filtragem de contaminantes da lista de massas ou propor modificações pós-traducionais. Para informações de sequência de proteína, as buscas podem ser realizadas levando-se em conta as possíveis ambigüidades da sequência, provenientes do método utilizado para sequenciamento (espectrometria de massa ou degradação de Edman).

- Realizar pós-processamento dos resultados, possibilitando a discriminação em função de taxonomia, levando em consideração todos os níveis da classificação.
- Realizar a melhoria da busca por meio de maior robustez nos filtros para inserção de dados, possibilitando o reenvio automático para os programas de busca. Novos picos de massa para *fingerprint* ou possíveis modificações nas proteínas e ambigüidade de seqüência são exemplos de informações que poderiam ser utilizadas na ressubmissão dos dados para avaliar melhoria nos resultados obtidos.
- A inclusão de outros programas de identificação, como a utilização de *sequence tag*, também poderia melhorar a qualidade dos resultados obtidos.
- Instalar, configurar e utilizar alguns dos programas de identificação em servidores da rede local. O programa Blast, por exemplo, é distribuído livremente. Já o programa Mascot necessita da compra de licença para uso.

Do ponto de vista computacional, poderia ser feita a normalização completa do banco de dados e a utilização de programação orientada a objetos, facilitando a reutilização do código. Outra medida seria desenvolver resultados em XML, promovendo a compatibilidade com os projetos *open-ms*, como o TPP (*Trans Proteomic Pipeline*) e o TOPP (*the OpenMS proteomics pipeline*).

## 8. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Watson, J.D. e Crick, F.H. (1953) *Molecular structure of Nucleic Acid*. Nature, Vol. 171 pp. 737-738.
- [2] Berg, J.M.; Tymoczko, L.L. e Stryer, L. (2008) *BIOQUIMICA*. Ed. Guanabara Koogan. pp. 25 a 63 (Capítulo 2).
- [3] Carr, S., Aebersold, R., Baldwin, M., Burlingame, A., Clauser, K., Nesvizhskii, A. (2004) *The Need for Guidelines in Publication of Peptide and Protein Identification Data*. Molecular & Cellular Proteomics. Editorial.
- [4] González, L.J., Castellanos-Serra, L., Badock, V., Díaz, M., Moro, A., Perea, S., Santos, A., Paz-Lago, D., Otto, A., Müller, E.C., Kostka, S., Wittmann-Liebold, B., Padrón, G. (2003) *Identification of nuclear proteins of small cell lung cancer cell line H82: An improved procedure for the analysis of silver-stained proteins*. Electrophoresis, Vol. 24 pp. 1-16.
- [5] Kohlbacher, O., Reinert, K., Gröpl, C., Lange, E., Pfeifer, N., Schulz-Trieglaff, O., Sturm, M. (2006) *TOPP—the OpenMS proteomics pipeline*. Bioinformatics, Vol. 23 pp. 191-197.
- [6] Castro, M.S., de Sá, N.M., Gadelha, R.P., de Sousa, M.V., Ricart, C.A., Fontes, B., Fontes, W. (2006) *Proteome analysis of resting human neutrophils*. Protein Pept. Lett. Vol. 13 n°5 pp. 481-487.
- [7] Speicher, D.W. (2004) *Proteome analysis Interpreting the genome*. Ed. Elsevier B.V. pp. 1-15 (Capítulo 1 - *Overview of proteome analysis*).
- [8] Westermeier, R.; Navem, T. e Höpker, H.R. (2008) *Proteomics in Practice – A guide to Successful Experimental Design*. 2ª ed, Ed. Wiley-VCH.
- [9] Wasinger, V.C., Cordwell, S.J., Cerpa-Poljak, A., Yan, J.X., Gooley, A.A., Wilkins, M.R., Duncan, M.W., Harris, R., Williams, K.L., Humphery-Smith, I. (1995) *Progress with gene-product mapping of the Mollicutes: Mycoplasma genitalium*. Electrophoresis. Vol. 16 n° 7 pp. 1090-1094
- [10] Matthiesen, R. (2007) *Methods, algorithms and tools in computational proteomics: A practical point of view*. Proteomics Vol. 7 n° 16: pp. 2815-2832.
- [11] Westermeier, R. e Navem, T. (2002) *Proteomics in Practice: a Laboratory Manual of Proteome Analysis*. Ed. Wiley-VCH.
- [12] International Union of Pure and Applied Chemistry e International Union of Biochemistry (1983) *Nomenclature and Symbolism for Amino Acids and Peptides (Recommendations 1983)*. Pure & Appl. Chem., Vol. 56 n° 5.
- [13] Nelson, D.L. e Cox, M.M. (2004) *Lehninger Principles of Biochemistry*. 4ª ed. Ed. W. H. Freeman & Co.
- [14] Wilkins, M.R., Gasteiger, E., Bairoch, A., Sanchez, J.C., Williams, K.L., Appel, R.D., Hochstrasser, D.F. (1999) *Protein Identification and Analysis Tools in the ExPASy Server*, Methods in molecular biology, Vol. 112 pp. 531-552
- [15] Pratt, C.W., Voet, D. e Voet, J.G. (2002) *Fundamentos de Bioquímica*. 1ª ed, Ed. Artmed
- [16] Eriksson, J., Fenyö, D. (2004) *The Statistical Significance of Protein Identification Results as a Function of the Number of Protein Sequences Searched*. Journal of Proteome Research, Vol. 3 n° 5 pp. 979-982.
- [17] Luscombe, N.M., Greenbaum, D., Gerstein, M. (2001) *What is Bioinformatics? A Proposed Definition and Overview of the Field*. Method Inform Med, Vol. 40 pp. 346-358.

- [18] Xiong, J. (2006) *Essential Bioinformatics*. Ed. Cambridge University Press.
- [19] Niessen, W.M.A. (2006) *Liquid chromatography and sample pretreatment*, in *Liquid Chromatography – Mass Spectrometry*. Ed. CRC Press.
- [20] Magalhães, A.D. (2006) *Análise Proteômica de Trypanosoma cruzi: construção de mapas bidimensionais em pH alcalino*. Universidade de Brasília: Brasília. Dissertação de Mestrado na Faculdade de Ciências da Saúde.
- [21] Herbert, C.G., Johnstone, R.A.W. (2003) *MASS SPECTROMETRY BASICS*. Ed. CRC Press LLC.
- [22] Matthiesen, R. (2007) *Mass Spectrometry Data Analysis in Proteomics*. Ed. Humana Press Inc.
- [23] Henzel, W.J., Billeci, T.M., Stults, J.T., Wong, S.C., Grimley, C., Watanabe, C. (1993) *Identifying proteins from two-dimensional gels by molecular mass searching of peptide fragments in protein sequence databases*. Proc Natl Acad Sci U S A, Vol. 90 nº11 pp. 5011-5015.
- [24] Huang, H.D., Lee, T.Y., Wu, L.C., Lin, F.M., Juan, H.F., Horng, J.T., Tsou, A.P. (2004) *MultiProtIdent: Identifying Proteins Using Database Search and Protein-Protein Interactions*. Journal of Proteome Research, Vol. 4 pp. 690-697.
- [25] CHANG, J.Y. CREASER, E.H. (1976) *A Novel Manual Method for Protein-Sequence Analysis* Biochem Journal, Vol. 157 pp. 77-85.
- [26] Centro Brasileiro de Serviços e Pesquisas em Proteínas. Disponível em: <http://www.unb.br/cbsp/> Visitado em 20/05/2008
- [27] Wilkins, W.R., Oua, K., Appel, R.D., Sanchez, J.C., Yan, J.X., Golaz, O., Farnsworth, V., Cartier, P., Hochstrasser, D.F., Williams, K.L., Gooley, A.A. (1996) *Rapid Protein Identification Using N-Terminal "Sequence tag" and Amino Acid Analysis*. Biochemical and biophysical research communications Vol. 221 nº 3 pp. 609-613.
- [28] Pappin, D.J.C., Hojrup, P., Bleasby, A.J. (1993) *Rapid identification of proteins by peptide-mass fingerprinting*. Current Biology, Vol. 3 nº 6 pp. 327-332.
- [29] *AACompIdent*. Disponível em: <http://www.expasy.org/tools/aacomp/>. Visitado em: 10/12/2007
- [30] *MultiIdent*. Disponível em: <http://expasy.org/tools/multiident/multiident4.html>. Visitado em: 10/06/2008
- [31] Korth, H.F., Silberchatz, A., Sudarshan, S. (1999) *Sistemas de Bancos de Dados*. 3ª ed., Ed. Makron Books.
- [32] Codd, E.F. (1970) *A Relational Model of Data for Large Shared Data Banks*. ACM Vol13 pp. 377-387.
- [33] The UniProt Consortium (2007) *The Universal Protein Resource (UniProt.)* Nucleic Acids Research, Vol. 35 pp. D190-D195.
- [34] Fielding, R., Gettys, J., Mogul, J.C., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T. (1999) *Hypertext Transfer Protocol -- HTTP/1.1*. IETF RFC 2616.
- [35] Laurie, B., Laurie, P. (2000) *Apache: The Definitive Guide*. 2ª ed., Ed. O'Reilly
- [36] *Netcraft*. Disponível em: <http://www.netcraft.com/> Visitado em 23/07/2008
- [37] *PHP: Documentation*. Disponível em: <http://www.php.net/docs.php> Visitado em 23/01/2008
- [38] Thomson, L. Welling, L. (2003) *PHP e MYSQL Desenvolvimento Web*. Ed. Campus.
- [39] Fisher, R.A. (1925) *Statistical Methods for Research Workers*. Ed. Edinburgh: Oliver & Boyd. Disponível em: <http://psy.ed.asu.edu/~classics/Fisher/Methods/> Visitado em 10/03/2008
- [40] Bailey, T.L., Gribnikov, M. (1998) *Combining evidence using p-values:*

- application to sequence homology searches*. Bioinformatics, Vol 14 pp. 48-54.
- [41] NCBI (2004) *Glossary*. Disponível em: <http://www.ncbi.nlm.nih.gov/Education/BLASTinfo/glossary2.html>. Visitado em 20/06/2008
- [42] Sanger, F., Tuppy, H. (1951) *The amino-acid sequence in the phenylalanyl chain of insulin*. Biochem Journal, Vol. 49 pp. 481-490.
- [43] Bairoch, A. (2000) *Serendipity in Bioinformatics, the tribulations of a Swiss bioinformatician through exciting times!* Bioinformatics, Vol.16 n°1 pp. 48-64.
- [44] Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M.C., Estreicher, A., Gasteiger, E., Martin, M.J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S., Schneider, M. (2003) *The Swiss-Prot Protein Knowledgebase and its supplement TrEMBL*. Nucleic Acids Research, Vol.31 pp. 365-370.
- [45] *UniProt - Universal Protein Resource*. Disponível em: <http://www.uniprot.org>. Visitado em 10/05/2008
- [46] *PIR - Protein Information Resource*. Disponível em: <http://pir.georgetown.edu/> Visitado em 10/05/2008
- [47] *Swiss Institute of Bioinformatics*. Disponível em: <http://www.isb-sib.ch/> Visitado em 10/05/2008
- [48] *European Molecular Biology Laboratory* Disponível em: <http://www.ebi.ac.uk/embl> Visitado em 10/05/2008
- [49] Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J. (1990) *Basic Local Alignment Search Tool*. Journal of Molecular Biology, Vol. 215 pp. 403-410.
- [50] Karlin, S., Altschul, S.F. (1990) *Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes*. Proc. Natl. Acad. Sci. USA, Vol.87 pp. 2264-2268.
- [51] Altschul, S.F. (2008) *The Statistics of Sequence Similarity Scores*. Disponível em: <http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html>. Visitado em 15/12/2007
- [52] Lipman, D., Pearson, W. (1985) *Rapid and sensitive protein similarity searches*. Science, Vol. 227 n° 4693 pp. 1435-1441.
- [53] Barton, G.J. (1996) *Protein Sequence Alignment and Database Scanning*. Protein Structure prediction - a practical approach, Ed. Oxford University Press.
- [54] Altschul, S.F. (1991) *Amino acid substitution matrices from an information theoretic perspective*. Journal of Molecular Biology, Vol. 219 no 3 pp. 555-565.
- [55] Dayhoff, M.O., Schwartz, R.M., Orcutt, B. C. (1978) *A Model of Evolutionary Change in Proteins*. Atlas of Protein Sequence and Structure - 1978. pp. 345-352 (Cap. 22).
- [56] Kosiol, C., Goldman, M. (2005) *Different Versions of the Dayhoff Rate Matrix*. Molecular Biology and Evolution, Vol. 22 n° 2: p. 193-199.
- [57] Henikoff, S., Henikoff, J.G. (1992) *Amino acid substitution matrices from protein blocks*. Proc Natl Acad Sci U S A, Vol. 89 pp. 10915-10919.
- [58] Gonnet, G.H., Cohen, M.A., Benner, S.A. (1992) *Exhaustive matching of the entire protein sequence database*. Science, Vol. 256 no.5062 pp. 1443-1445.
- [59] Jones, D.T., Taylor, W.R., Thornton, J.M. (1992) *The rapid generation of mutation data matrices from protein sequences*. Comput. Appl. Biosci, Vol. 8 pp. 275-282.
- [60] *Matrixscience*. Disponível em: <http://www.matrixscience.com/> Visitado em: 10/10/2007
- [61] MatrixScience (2005) *Mascot Brochure*. Ed. Matrix Science Ltd. Vol. 01-2/2005.
- [62] MatrixScience (2008) *Mascot Search Overview*. Disponível em:

- [http://www.matrixscience.com/search\\_intro.html](http://www.matrixscience.com/search_intro.html) Visitado em: 10/06/2008
- [63] MatrixScience (2007) *Scoring Schemes*. Disponível em:  
[http://www.matrixscience.com/help/scoring\\_help.html](http://www.matrixscience.com/help/scoring_help.html) Visitado em: 11/06/2008
- [64] Perkins, D.N., Pappin, D.J., Creasy, D.M., Cottrell, J.S. (1999) *Probability-based protein identification by searching sequence databases using mass spectrometry data*. . Electrophoresis, Vol. 20 n° 18 pp. 3551-3567.
- [65] *Phenyx*. Disponível em: <http://www.phenyx-ms.com>. Visitado em 10/06/2008
- [66] GeneBio (2007) *Phenyx GENE BIO Product Brochure*. Ed. Geneva Bioinformatics (GENEBIO) S/A.
- [67] Colinge, J., Masselot, A., Giron, M., Dessingy, T., Magnin, J. (2003) *OLAV: towards high-throughput tandem mass spectrometry data identification*. Proteomics, Vol. 3 n° 8 pp. 1454-1463.
- [68] GeneBio (2005) *Phenyx web interface - User Manual*. Ed. Geneva Bioinformatics (GENEBIO) S/A.
- [69] Catanho, M., Mascarenhas, D., Degrave, W., de Miranda, A.B. (2006) *BioParser: A Tool for Processing of Sequence Similarity Analysis Reports*. Applied Bioinformatics, Vol. 5 n° 1 pp. 49-53.
- [70] Stajich, J.E., Block, D., Boulez, K., Brenner, S.E., Chervitz, S.A., Dagdigian, C., Fuellen, G., Gilbert, J.G., Korf, I., Lapp, H., Lehtväslaiho, H., Matsalla, C., Mungall, C.J., Osborne, B.I., Pocock, M.R., Schattner, P., Senger, M., Stein, L.D., Stupka, E., Wilkinson, M.D., Birney, E. (2002) *The Bioperl toolkit: Perl modules for the life sciences*. Genome Research, Vol. 12 n° 10 pp. 1611-1618.
- [71] Calçado, V.L.X.d.S. (2007) *Influência da Utilização de Processo Unificado, Testes e Métricas na Qualidade de Produtos de Software*. Universidade de Brasília: Brasília. Dissertação de Mestrado no Departamento de Engenharia Elétrica
- [72] Teles, V.M. (2004) *Extreme Programming: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade*. Ed. Novatec
- [73] *PEAR - PHP Extension and Application Repository*. Disponível em:  
<http://pear.php.net/> Visitado em: 15/01/2008
- [74] *ProFound*. Disponível em: <http://prowl.rockefeller.edu/prowl-cgi/profound.exe>. Visitado em 10/06/2008
- [75] *MultiIdent*. Disponível em: <http://expasy.org/tools/multiident/multiident4.html> Visitado em 18/06/2008
- [76] *NCBI - Protein Home*. Disponível em:  
<http://www.ncbi.nlm.nih.gov/sites/entrez?db=protein>. Visitado em: 10/06/2008
- [77] Peri, S., Steen, H., Pandey, A. (2001) *GPMAW - a software tool for analyzing proteins and peptides*. Trends in biochemical sciences. Vol. 26 n° 11 pp. 687-689.



## **APÊNDICES**

## **A.DOCUMENTAÇÃO DO SOFTWARE E CASOS DE USO**

### **A.1 PROPÓSITO DO DOCUMENTO**

Este documento visa detalhar a funcionalidade da aplicação “Protein Locator”, definindo o escopo da solução. O documento é dividido nos diferentes Casos de Uso do sistema, em cada um é feita uma descrição do Caso e são exibidos o Diagrama de Caso de Uso (*Use Case Diagram* – UCD), um cenário de falha, um cenário de sucesso e as Regras de Negócio.

### **A.2 ABREVIATURAS UTILIZADAS**

BD – Banco de Dados

PL – Protein Locator

RN – Regra de Negócio

UC – Caso de Uso

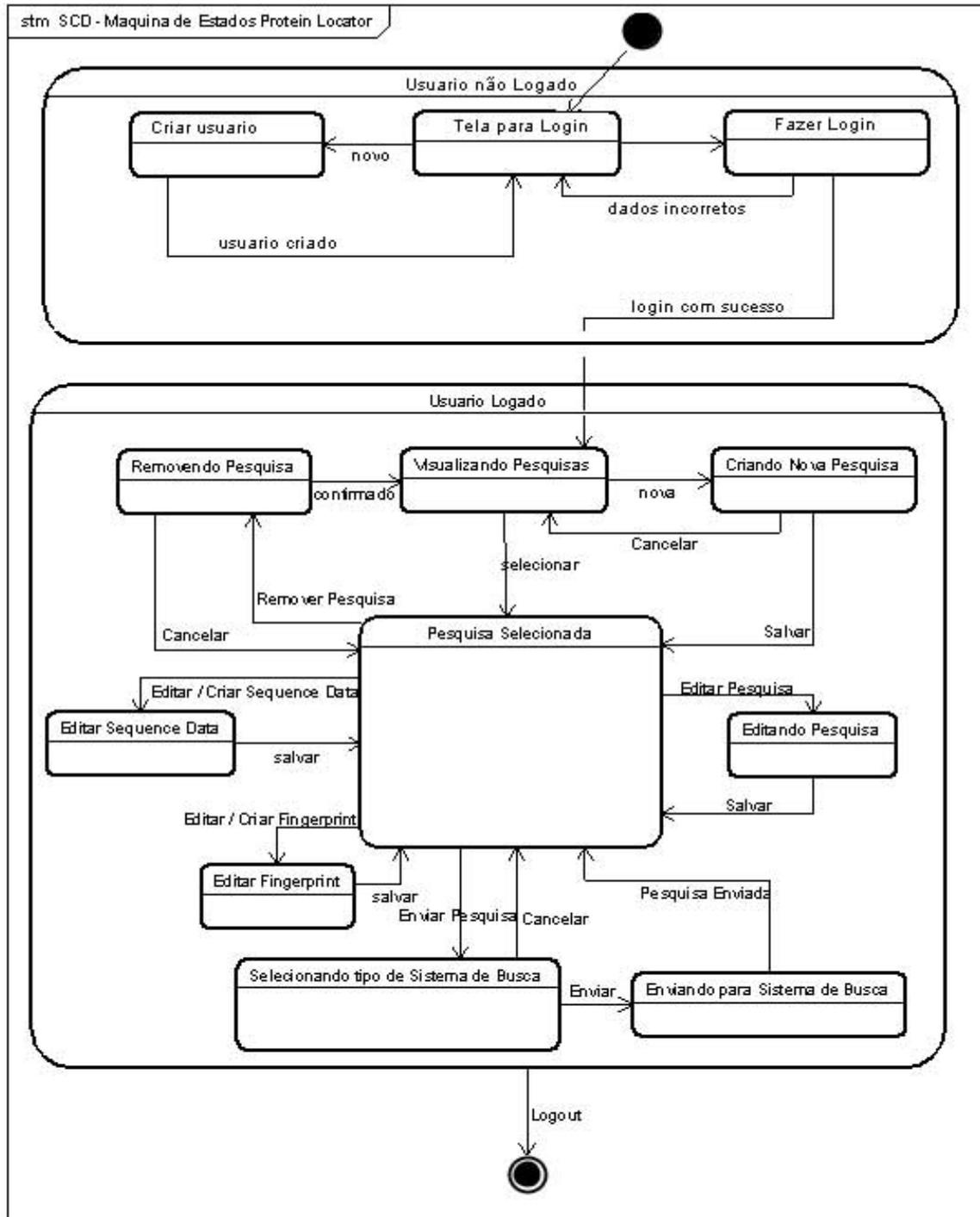
UCD – Diagrama de Caso de Uso

### **A.3 VISÃO GERAL DO SOFTWARE**

#### **A.3.1. Descrição**

O objetivo do software é: permitir a identificação de proteínas utilizando várias ferramentas de identificação disponíveis na internet, com diferentes algoritmos e tipos de dados; realizar a separação de dados de experimentos por usuário e por pesquisas de cada usuário; apresentar um resultado consolidado após as buscas em todos os programas escolhidos pelo usuário.

### A.3.2.Principais Interações do Usuário



- Legenda:
- Estado Inicial
  - Estado Final

Figura A-1 Diagramas de estados do Protein Locator

### **A.3.2.1 Criação de usuários**

Neste estado, será criado o usuário para que ele possa efetuar *login* no sistema. Após a criação, é informado o sucesso da operação e o novo usuário é informado que deve retornar a página de *login*.

### **A.3.2.2 Usuário Logado**

A sessão do usuário se encontra neste estado após o usuário efetuar *Login* no sistema. Imediatamente após o *Login* o usuário é colocado no estado “Visualizando Pesquisas”, descrito a seguir.

Ações:

- Visualizar Pesquisas;
- Selecionar uma Pesquisa;
- Criar nova Pesquisa;
- Apagar Pesquisa;
- *Logout*: Finaliza a sessão do usuário.

### **A.3.2.3 Visualizando Pesquisas**

Neste estado o usuário é capaz de visualizar suas pesquisas, mas não possui nenhuma pesquisa selecionada. É o estado no qual a sessão é colocada após a realização do *Login*.

Ações:

- Selecionar: Ocorre quando o usuário seleciona uma de suas pesquisas criadas.
- Apagar: Ocorre quando o usuário seleciona apagar uma de suas pesquisas criadas;

- Nova: Ocorre quando o usuário deseja criar uma nova pesquisa.

#### **A.3.2.4 Criando Nova Pesquisa**

Neste estado, o sistema recebe os dados para a criação de uma nova pesquisa, tais como: nome da pesquisa, pI, massa, taxonomia, palavras-chave e comentários sobre a pesquisa.

Ações:

- Salvar: Ocorre quando o usuário decide persistir as informações que estava digitando.
- Cancelar: Ocorre quando o usuário decide cancelar a criação da pesquisa.

#### **A.3.2.5 Pesquisa Seleccionada**

Ocorre quando o sistema identifica que o usuário possui alguma pesquisa seleccionada. É neste estado que o sistema permite a maior parte das operações.

Ações:

- Editar Pesquisa: Ocorre quando o usuário decide editar a pesquisa seleccionada.
- Editar / Criar Composição de Aminoácidos: Ocorre quando o usuário decide criar ou editar dados de composição de aminoácidos para a pesquisa seleccionada.
- Editar / Criar *Fingerprint*: Ocorre quando o usuário decide criar ou editar dados de *Peptide Mass Fingerprint* para a pesquisa seleccionada.
- Editar / Criar *Sequence data*: Ocorre quando o usuário decide criar ou editar dados de seqüências de aminoácidos.
- Enviar Pesquisa: Ocorre quando o usuário decide enviar a pesquisa para os sistemas de busca seleccionados.

- Remover Pesquisa: Ocorre quando o usuário decide remover a pesquisa selecionada.

#### **A.3.2.6 Editando Pesquisa**

Neste estado o usuário está alterando os dados possíveis da pesquisa, tais como: nome da pesquisa, pI, peso, taxonomia, palavras-chave e comentários sobre a pesquisa.

Ações:

- Salvar: Persistem os dados da pesquisa.

#### **A.3.2.7 Removendo Pesquisa**

Neste estado, o usuário precisa confirmar a remoção da pesquisa.

Ações:

- Confirmar: Confirma a remoção da pesquisa.
- Cancelar: Cancela a remoção da pesquisa.

#### **A.3.2.8 Adicionando Composição de Aminoácidos**

Neste estado o usuário está criando uma composição de aminoácidos.

Ações:

- Salvar: Persistem os dados de composição de aminoácidos.

#### **A.3.2.9 Editando Composição de Aminoácidos**

Neste estado o usuário está configurando os dados de uma composição de aminoácidos

Ações:

- Salvar: Persistem os dados de composição de aminoácidos.

### **A.3.2.10 Removendo Composição de Aminoácidos**

Neste estado, o usuário está removendo a composição de aminoácidos.

Ações:

- Remover: Remove composição de aminoácidos.

### **A.3.2.11 Criando *Fingerprint***

Neste estado o usuário está criando um *fingerprint*.

Ações:

- Salvar: Persistem os dados do *fingerprint*.

### **A.3.2.12 Editando *Fingerprint***

Neste estado o usuário está configurando os dados de um *fingerprint*.

Ações:

- Salvar: Persistem os dados do *fingerprint*.

### **A.3.2.13 Removendo *Fingerprint***

Neste estado, o usuário está removendo um conjunto específico de dados de um *fingerprint*.

Ações:

- Remover: Remove dados de um *fingerprint*.

### **A.3.2.14 Criando *Sequence data***

Neste estado o usuário está criando uma sequência.

Ações:

- Salvar: Persiste os dados da seqüência.

#### **A.3.2.15 Editando *Sequence data***

Neste estado o usuário está configurando os dados de uma seqüência.

Ações:

- Salvar: Persiste os dados da seqüência.

#### **A.3.2.16 Removendo *Sequence data***

Neste estado, o usuário está removendo um conjunto específico de dados de seqüência.

Ações:

- Remover: Remove dados de uma seqüência.

#### **A.3.2.17 Selecionando tipo de sistema de busca**

Neste estado o usuário decide para qual sistema de busca deseja enviar a sua pesquisa (Mascot, Fasta, Blast ou AACompident).

- Enviar: Ocorre quando o usuário seleciona os tipos de sistemas de busca para os quais irá submeter a pesquisa.

#### **A.3.2.18 Enviando para Sistema de busca**

Neste estado o sistema transforma os dados da pesquisa cadastrada no formato requerido pelo sistema de busca.

- Pesquisa Enviada: Ocorre depois que o sistema confirma o envio da pesquisa para o serviço selecionado.

### **A.3.2.19 Visualizando resultados**

Neste estado o sistema consolida os resultados recebidos dos programas de busca e exibe ao usuário o resultado consolidado e os resultados individuais dos programas.

- Resultado consolidado: Ocorre depois que o sistema calcula a consolidação dos resultados

## **A.4 CASOS DE USO**

### **A.4.1. Criar novo Usuário**

#### **A.4.1.1 Descrição Detalhada**

O sistema deve permitir a criação de novos usuários. O sistema só permitirá acesso a algumas de suas funcionalidades a usuários cadastrados. O cadastro de um novo usuário pode ser feito por ele mesmo, não sendo necessária aprovação por moderador. O cadastro de usuário é necessário apenas para que os dados armazenados no banco local sejam associados a quem os inseriu e possam ser recuperados pelo mesmo usuário no futuro.

#### **A.4.1.2 Atores**

- Usuário
- Banco de Dados (BD)

#### **A.4.1.3 Premissas / Pré-Condições**

O usuário só será criado com sucesso caso seu registro não exista no banco. Criação de usuários já existentes resultará em erro e solicitação de novos dados ao usuário.

#### A.4.1.4 Diagrama de Caso de Uso



**Figura A-2 UCD Criar novo usuário**

#### A.4.1.5 Principais Cenários

##### **Cenário de Criação de usuário com sucesso**

1. O caso começa com o Usuário acessando a aplicação PL (Protein Locator) pela página *'Welcome'*;
2. O usuário seleciona a opção *Login*;
3. O usuário escolhe a opção *'New User'*
4. O usuário preenche os campos de criação
5. O usuário clica no botão *'submit'*
6. O *e-mail* é validado de acordo com a regra RN1
7. O *Password* é validado de acordo com a regra RN2
8. O cadastro do usuário é submetido ao BD com sucesso.
9. É apresentado ao usuário um link para a página de *login*.
10. O caso de uso é encerrado com sucesso.

##### **Cenário de falha de criação por usuário já cadastrado:**

1. O caso começa com o Usuário acessando a aplicação PL pela página *'Welcome'*;

2. O usuário seleciona a opção *Login*;
3. O usuário escolhe a opção *'New User'*
4. O usuário preenche os campos de criação
5. O clica no botão *'submit'*
6. O *e-mail* é validado de acordo com a regra RN1
7. O *Password* é validade de acordo com a regra RN2
8. O cadastro do usuário é submetido ao BD que rejeita a criação, pois o usuário já está cadastrado.
9. O usuário é direcionado ao formulário para nova entrada de dados e é apresentada uma mensagem explicando a duplicidade de *e-mail*.
10. O caso de uso é encerrado com falha.

#### **A.4.1.6Regras**

##### **RN1**

O *e-mail* necessita do caracter '@' e do caracter '.'

##### **RN2**

O *'Password'* deve ser igual ao *'Confirm Password'*

#### A.4.1.7 Telas e interfaces

Create new user

**Please fill this form to access the restricted areas:**

Full name:

User-id (E-mail):

Password

Confirm Password:

**Figura A-3 Formulário para criação de novo usuário**

#### A.4.2. Efetuar *Login* no Sistema

##### A.4.2.1 Descrição Detalhada

O usuário deve poder efetuar o *login* no sistema. Esta operação visa conceder acesso às funcionalidades da aplicação restritas aos usuários cadastrados, permitindo a recuperação dos dados específicos de cada usuário. As funcionalidades que não envolvem dados experimentais submetidos nem resultados recuperados podem ser acessadas independentemente de *login* (ex: telas de ajuda e links para os serviços consultados)

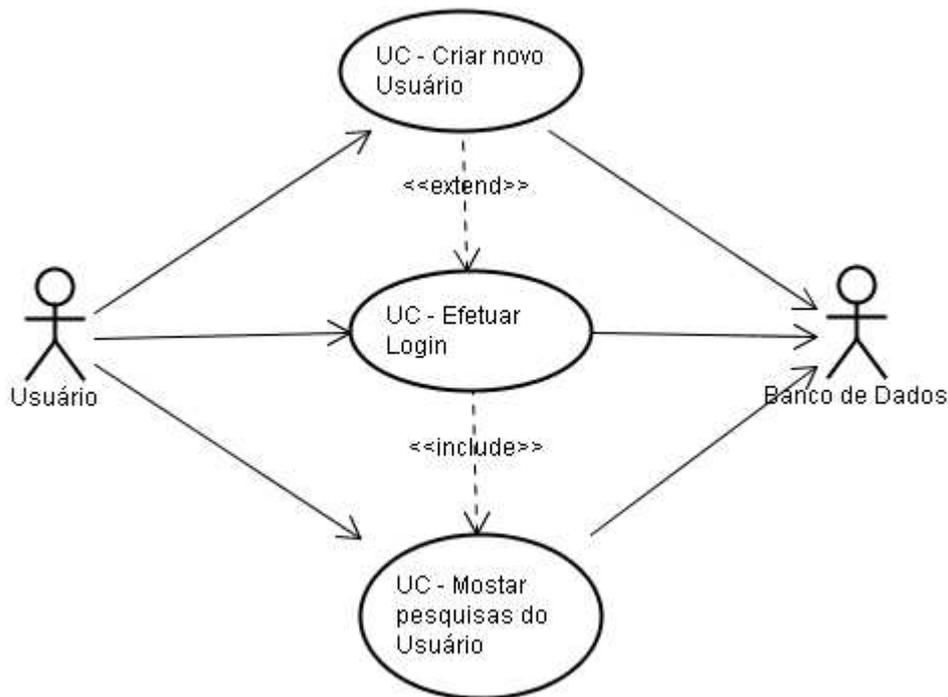
##### A.4.2.2 Atores

- Usuário
  
- Banco de Dados

#### A.4.2.3 Premissas / Pré-Condições

Para um *login* bem sucedido, é necessário que o usuário já esteja cadastrado no sistema e digite o *e-mail* e a senha correspondentes entre si.

#### A.4.2.4 Diagrama de Caso de Uso



**Figura A-4 UCD Login de usuário**

#### Legenda:

UC – Use Case (Caso de Uso)

<<*extend*>> - Estende a funcionalidade (ou seja, é executado opcionalmente no processo que está sendo apontado)

<<*include*>> - Inclui a funcionalidade (ou seja, sempre chama o processo que está sendo apontado)

#### A.4.2.5 Principais Cenários

##### **Login de Usuário com Sucesso**

1. O caso começa com o Usuário acessando a aplicação PL.
2. O usuário seleciona a tela de *Login*.

3. O usuário entra com o seu *e-mail* e seu *password*.
4. O usuário escolhe a opção '*Login*'.
5. O dados são validados com o banco de dados de acordo com a regra RN1.
6. O caso de uso '*Exibir pesquisas do usuário*' é invocado.
7. O caso de uso é encerrado com sucesso.

#### ***Login de Usuário com Falha***

1. O caso começa com o Usuário acessando a aplicação PL.
2. O usuário seleciona a tela de *Login*.
3. O usuário entra com o seu *e-mail* e seu *password*.
4. O usuário escolhe a opção '*Login*'.
5. O dados são validados com o banco de dados de acordo com a regra RN1.
6. Ocorre falha na validação dos dados com banco, por *e-mail* ou senha incorretos .
7. O caso de uso é encerrado com falha.

#### **A.4.2.6Regras**

##### **RN1**

O *e-mail* do usuário deverá estar cadastrado no banco de dados e o resumo (hash) da senha informada pelo usuário deverá estar de acordo com o resumo (hash) armazenado no banco de dados.

#### A.4.2.7 Telas e interfaces

Please log in to access this document

**Username (e-mail):**

**Password:**

**Figura A-5 Formulário para *login***

#### A.4.3. Visualizar pesquisas do Usuário

##### A.4.3.1 Descrição Detalhada

O sistema deve exibir ao usuário as pesquisas que ele cadastrou no sistema, permitindo que ele visualize a quantidade de formulários preenchidos, apague toda a pesquisa e visualize seus detalhes. Essa funcionalidade deve ser invocada toda vez que o usuário entrar no sistema (imediatamente após o *login*, sem interferência do usuário) ou clicar no link *<view queries>* (como exibido na Figura 14, Tela de criação de pesquisa avançada) em qualquer página do software.

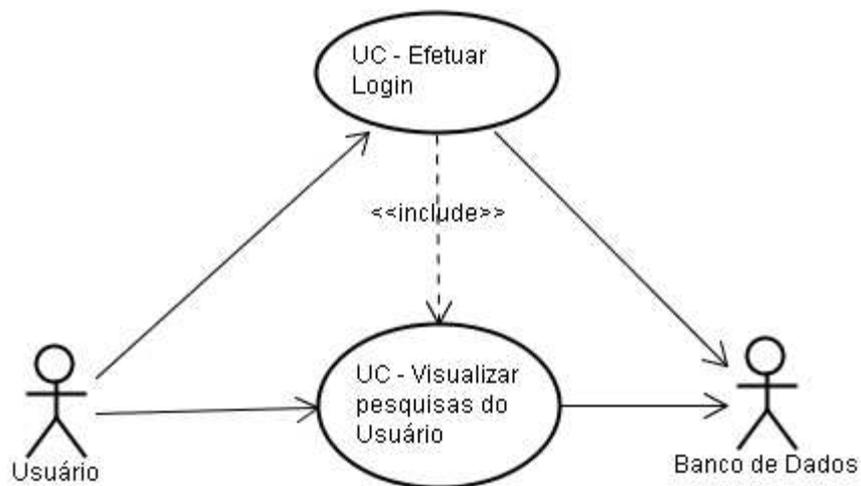
##### A.4.3.2 Atores

- Usuário
- Banco de Dados

##### A.4.3.3 Premissas / Pré-Condições

Para visualizar as pesquisas corretamente, o usuário já deverá ter dados armazenados no banco de dados e ter efetuado o *login* corretamente.

#### A.4.3.4 Diagrama de Caso de Uso



**Figura A-6 UCD Visualizar pesquisas do usuário**

#### A.4.3.5 Principais Cenários

##### **Cenário 1: Exibindo a pesquisa para o usuário**

1. O caso de uso começa quando o usuário é direcionado para uma tela com a lista de todas as pesquisas que ele já cadastrou no sistema (isto ocorre ao fazer *login* ou ao pressionar o link *<view queries>*).
2. É apresentada ao usuário uma tabela com o resumo de dados de cada pesquisa, contendo o nome da pesquisa e a quantidade de resultados de composição de aminoácidos, PMF, seqüências de peptídeos, seqüências da proteína, sequance-tags e se os dados foram submetidos a algum mecanismo de busca.
3. É apresentada ao usuário a opção de apagar cada pesquisa ou visualizar os dados detalhados de cada pesquisa (situações descritas nos casos de uso 3.5 e 3.6).
4. O caso de uso é finalizado com sucesso.

##### **Cenário 2: Usuário sem pesquisa cadastrada**

1. O caso de uso começa quando o usuário é direcionado para uma tela com a lista de todas as pesquisas que ele já cadastrou no sistema.

2. É apresentado ao um aviso de que não há pesquisa alguma cadastrada em seu nome no banco de dados (conforme Figura 7 Tela de usuário sem pesquisa cadastrada).

3. O caso de uso é finalizado com sucesso.

### A.4.3.6Regras

Não existem regras de negócio para este caso de uso.

### A.4.3.7Telas e interfaces

The user **teste** has 21 rows of data.

**Forms filled by the user**

Select	Query name	Aminoacid composition	Number of PMF	Number of Proteins	Delete	Submitted
Select	insulin receptor	0	0	1	DEL	<a href="#">fasta</a> 20080423115747 <a href="#">blast</a> 20080423115747
Select	Heat shock	0	0	1	DEL	<a href="#">fasta</a> 20080426173001 <a href="#">blast</a> 20080426173001
Select	insulina MLR497	1	0	2	DEL	<a href="#">blast</a> 20080429213609 <a href="#">blast</a> 20080429213727 <a href="#">fasta</a> 20080501094906

**Figura A-7 Visualização de lista de pesquisas**

### A.4.4.Criar Pesquisa

#### A.4.4.1 Descrição Detalhada

O usuário deve poder criar uma nova pesquisa no sistema. Uma pesquisa é o conjunto de dados experimentais sobre uma proteína e deve conter, pelo menos, alguns dos dados genéricos. O usuário deve nomear a pesquisa além de identificar dados como pI, massa molecular, taxonomia, palavras-chave, além de poder inserir um comentário sobre a pesquisa.

Para criar uma nova pesquisa, o usuário deve acessar o menu “Data Entry”, sub-menus “Form type” – “Basic” ou “Advanced” – “Protein - generic”, conforme mostra a figura 5

O presente caso de uso prevê somente a inserção dos dados gerais sobre a proteína. Dados específicos de outros experimentos serão descritos nos próximos casos de uso.

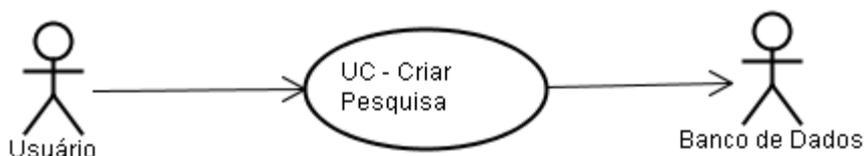
#### A.4.4.2 Atores

- Usuário
- Banco de Dados

#### A.4.4.3 Premissas / Pré-Condições

Para a criação de uma pesquisa, o usuário deve estar logado no sistema.

#### A.4.4.4 Diagrama de Caso de Uso



**Figura A-8 UCD Criar pesquisa**

#### A.4.4.5 Principais Cenários

##### **Cenário 1: Criação de pesquisa básica com sucesso**

1. O usuário seleciona a opção de criação de pesquisa básica acessando o menu “Data Entry”, sub-menus “Form type” – “Basic” – “Protein - generic”, conforme mostra a Figura 8 Link para criação de nova pesquisa.

2. É apresentado ao usuário um formulário conforme mostra a Figura 10 Tela de criação de pesquisa simples.
3. A pesquisa é validada de acordo com a regra “validação de formulário de pesquisa”.
4. A pesquisa é submetida para criação no banco de dados.
5. O banco de dados aceita a criação da pesquisa.
6. O usuário é encaminhado para uma tela confirmando a criação da pesquisa (Figura 11 Tela de confirmação de cadastro de pesquisa)
7. O caso de uso é concluído com sucesso.

### **Cenário 2: Criação de pesquisa avançada com sucesso**

1. O usuário seleciona a opção de criação de pesquisa avançada acessando o menu “Data Entry”, sub-menus “Form type” – “*Advanced*” – “Protein - generic”, conforme mostra a Figura 8 Link para criação de nova pesquisa.
2. É apresentado ao usuário um formulário conforme mostra a Figura 9 Tela de criação de pesquisa avançada.
3. A pesquisa é validada de acordo com a regra “validação de formulário de pesquisa”.
4. A pesquisa é submetida para criação no banco de dados.
5. O banco de dados aceita a criação da pesquisa.
6. O usuário é encaminhado para uma tela confirmando a criação da pesquisa (Figura 11 Tela de confirmação de cadastro de pesquisa)
7. O caso de uso é concluído com sucesso.

#### **A.4.4.6Regras**

##### **RN1 - Validação de formulário de pesquisa**

Para que um formulário seja considerado válido, é necessário que:

- A pesquisa precisa ter um nome;
- Não exista uma pesquisa com o mesmo nome cadastrada no banco de dados.
- O campo pI deve ser preenchido e conter valores numéricos de 0 a 14
- O campo massa deve ser preenchido e conter valores numéricos positivos.
- O campo taxonomy deve ser preenchido. O texto inserido deve ser validado pelo sistema com o banco de dados do NCBI (<http://www.ncbi.nlm.nih.gov/Taxonomy/CommonTree/wwwcmt.cgi>)

#### **A.4.4.7Telas e interfaces**

**Protein generic data:**

Query name

pI\*  within pI range\*  pI units  Weight

MW(in Daltons)\*  within mw range\*  %  Weight

Taxonomy \*

Keywords:

Comments:

**Figura A-9 Formulário para criação de pesquisa**

## A.4.5. Visualizar Detalhamento da Pesquisa

### A.4.5.1 Descrição Detalhada

Depois que forem apresentadas todas as suas pesquisas, o usuário deve poder visualizar detalhes de uma determinada pesquisa. Serão apresentados ao usuário:

- Dados gerais da pesquisa (vide item 3.4);
- O conteúdo de todos os formulários referentes à pesquisa (dados gerais, composição de aminoácidos, sequência e *fingerpint*), com opções de edição, submissão e remoção dos mesmos. Tais opções serão descritas adiante, em casos de uso específicos.

### A.4.5.2 Atores

- Usuário
- Bando de Dados

### A.4.5.3 Premissas / Pré-Condições

O usuário deve estar logado na aplicação e possuir pelo menos uma pesquisa pré-cadastrada (vide item 3.4).

### A.4.5.4 Diagrama de Caso de Uso

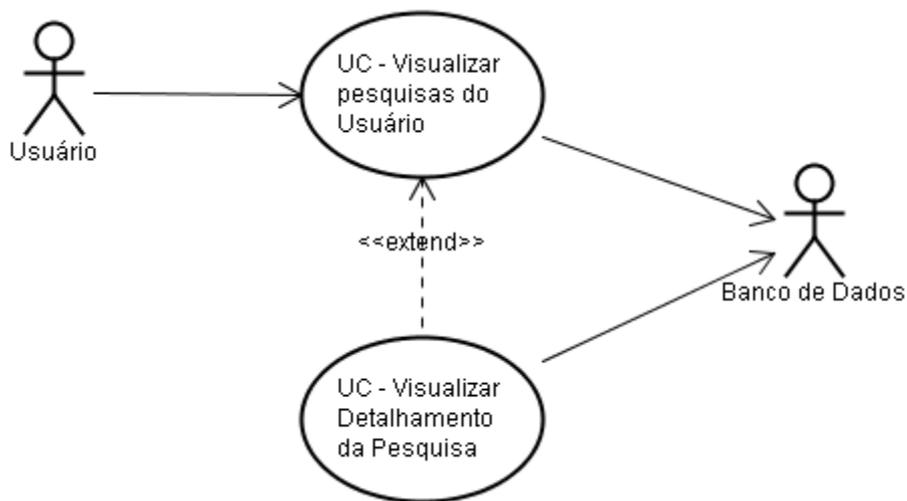


Figura A-10 UCD - Visualizar Detalhamento de Pesquisa

### A.4.5.5 Principais Cenários

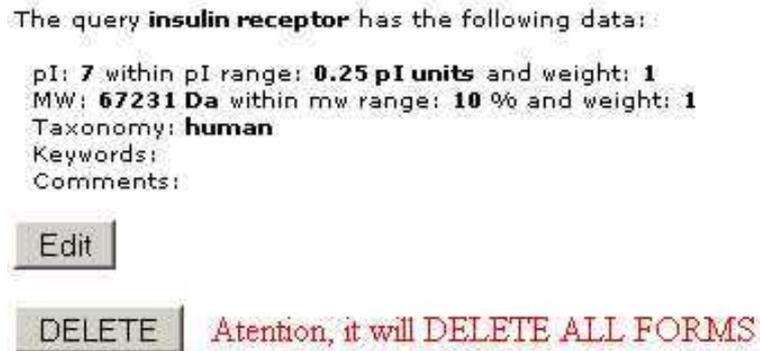
#### Cenário 1

1. É apresentada ao usuário sua lista de pesquisas.
2. O usuário seleciona uma pesquisa clicando no botão seleciona ao lado do nome da pesquisa.
3. O usuário é encaminhado para a tela de visualização de detalhes da pesquisa.
4. O caso de uso é encerrado com sucesso.

#### A.4.5.6Regras

Não existem regras para este caso de uso.

#### A.4.5.7Telas e interfaces



**Figura A-11 Visualizar detalhamento de pesquisa**

#### A.4.6.Remover Pesquisa

##### A.4.6.1Descrição Detalhada

O usuário deve poder remover uma pesquisa e todas as informações vinculadas a ela. Esta opção será invocada por meio da tela de visualização de pesquisas ou de detalhamento de pesquisa.

##### A.4.6.2Atores

- Usuário
- Banco de Dados

##### A.4.6.3Premissas / Pré-Condições

O usuário deve estar logado na aplicação e a pesquisa deve estar previamente criada.

#### A.4.6.4 Diagrama de Caso de Uso

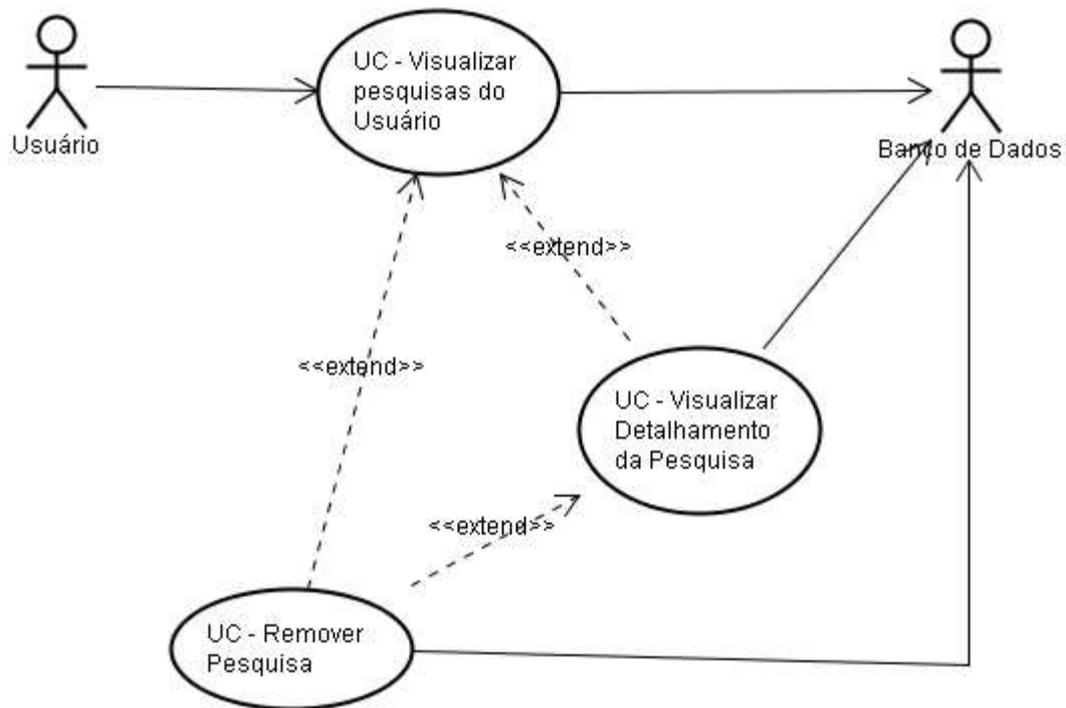


Figura A-12 UCD - Remover Pesquisa

#### A.4.6.5 Principais Cenários

##### Cenário 1: Removendo uma pesquisa após visualização dos seus detalhes

1. É apresentado ao usuário a tela de visualização de detalhamento de pesquisa.
2. O usuário seleciona a opção delete relativa à pesquisa, indicada no botão próximo aos dados gerais.
3. É apresentada ao usuário uma tela de confirmação de remoção.
4. O usuário confirma a remoção.
5. A pesquisa é removida do banco de dados com sucesso.
6. O usuário é encaminhado para uma tela informando que a remoção foi realizada com sucesso.

7. O caso de uso é encerrado com sucesso.

#### **A.4.6.6Regras**

##### **RN 1 Confirmar Remoção:**

Regra booleana, em que o usuário deve indicar se realmente deseja que a pesquisa seja removida.

#### **A.4.7.Adicionar composição de aminoácidos**

##### **A.4.7.1 Descrição Detalhada**

O usuário pode criar uma, e apenas uma, composição de aminoácidos para uma determinada pesquisa. Ele irá acessar essa funcionalidade depois de criar a pesquisa, ou enquanto estiver visualizando o detalhamento da mesma.

Ao cadastrar uma composição de aminoácidos, o usuário poderá escolher entre o modo básico e o avançado. No modo básico o usuário deve escolher entre a opção de “*mol percent*” e “*number of residues per sequence*”. Em ambas opções, será apresentada ao usuário uma tabela contendo as colunas para o preenchimento da composição e do peso (relevância) para cada aminoácido. Na opção “*mol percent*” a Composição é dada em porcentagem, enquanto que na opção “*number of residues per sequence*” esta quantidade é um número inteiro exato.

No modo avançado, além das funcionalidades do modo básico, o usuário ainda pode cadastrar uma proteína de calibração. Com isso, o usuário deve cadastrar para cada aminoácido a composição experimental encontrada para a proteína de calibração utilizada.

##### **A.4.7.2Atores**

- Usuário
- Banco de Dados

#### A.4.7.3 Premissas / Pré-Condições

O usuário deve estar logado no sistema, e a pesquisa deve estar previamente cadastrada.

#### A.4.7.4 Diagrama de Caso de Uso

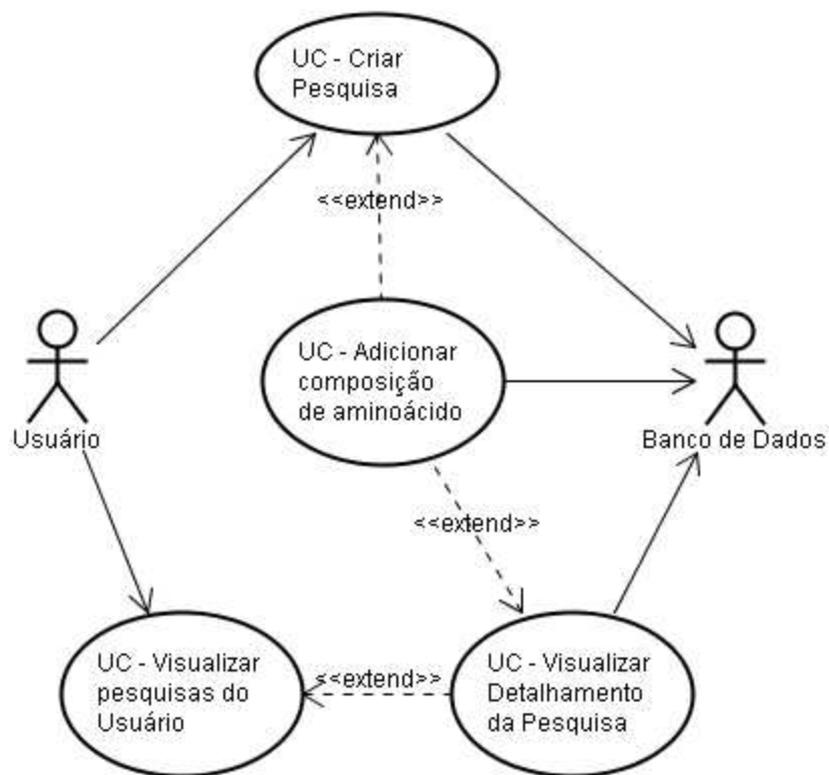


Figura A-13 UCD - Criar composição de aminoácido

#### A.4.7.5 Principais Cenários

**Cenário 1: Adicionando com sucesso uma composição de aminoácidos dentro da visualização detalhada da pesquisa pelo modo básico**

1. O usuário visualiza os detalhes da pesquisa;
2. O usuário seleciona a opção “Adicionar composição de aminoácido pelo modo básico” .

3. O sistema verifica a regra “Pesquisa sem composição de aminoácido” com sucesso.

4. O usuário é direcionado para uma tela de cadastro básico de aminoácidos presentes na amostra.

5. O usuário preenche os dados solicitados e seleciona a opção salvar formulário.

6. Cada um dos aminoácidos é validado pela regra “Validação de Aminoácido” com sucesso.

7. O usuário é encaminhado para uma tela de sucesso na criação de composição de aminoácidos.

8. O caso de uso é encerrado com sucesso.

**Cenário 2: Adicionando com falha uma composição de aminoácidos dentro da visualização detalhada da pesquisa pelo modo básico**

1. O usuário visualiza os detalhes da pesquisa;

2. O usuário seleciona a opção “Adicionar composição de aminoácido pelo modo básico”;

3. O sistema verifica a regra “Pesquisa sem composição de aminoácido” com sucesso.

4. O usuário é direcionado para uma tela de cadastro básico de aminoácidos presentes na amostra.

5. O usuário preenche os dados solicitados e seleciona a opção salvar formulário.

6. Cada um dos aminoácidos é validado pela regra “Validação de Aminoácido” ocorrendo uma falha na validação.

7. O usuário é encaminhado para uma tela de falha na criação de composição de aminoácidos.

8. O caso de uso é encerrado com falha.

#### **A.4.7.6 Regras**

##### **RN1 Pesquisa sem composição de aminoácido**

É verificado se a pesquisa não possui nenhuma composição de aminoácido cadastrada. Se possuir, a regra retorna falha.

##### **RN2 Validação de Aminoácido**

A regra será válida:

- Se a composição do ASX for maior que 0:
- a composição de ASN mais a composição de ASP deve ser maior que 0 e igual a composição de ASX.
  - OU as composições de ASP e ASN não podem ter sido preenchidas.
- Se a composição do GLX for maior que 0:
- a composição de GLN mais a composição de GLU deve ser maior que 0 e igual a composição de GLX.
  - OU as composições de GLN e GLU não podem ter sido preenchidas.
- Se pelo menos uma das composições de aminoácidos for preenchida;
- Se houver nome da proteína de calibração:
- A composição calibração de pelo menos um dos aminoácidos deve ser preenchida;

- Se houver alguma composição de calibração preenchida:
  - Deve haver o nome da proteína de calibração;

#### A.4.7.7 Telas e interfaces

mol percent

	Comp.				Comp.		
ALA	-1	1	-1	ILE	-1	1	-1
ARG	-1	1	-1	LEU	-1	1	-1
ASN	-1	1	-1	LYS	-1	1	-1
ASP	-1	1	-1	MET	-1	1	-1
ASX	-1	1	-1	PHE	-1	1	-1
CYS	-1	1	-1	PRO	-1	1	-1
GLN	-1	1	-1	SER	-1	1	-1
GLU	-1	1	-1	THR	-1	1	-1
GLX	-1	1	-1	TRP	-1	1	-1
GLY	-1	1	-1	TYR	-1	1	-1
HIS	-1	1	-1	VAL	-1	1	-1

**Figura A-14 Formulário de composição de aminoácidos**

#### A.4.8.Modificar composição de aminoácidos

##### A.4.8.1 Descrição Detalhada

Este caso de uso será utilizado para modificar os dados de uma composição de aminoácidos criado previamente.

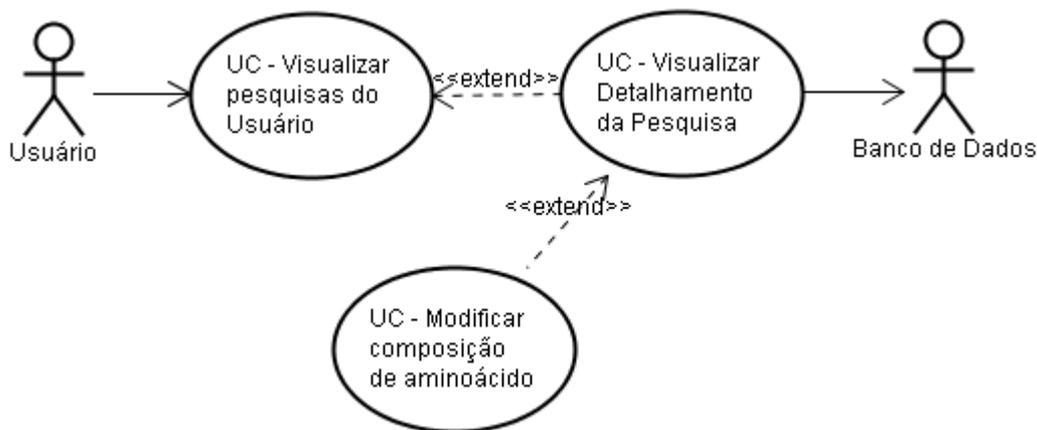
##### A.4.8.2 Atores

- Usuário
- Banco de Dados

#### A.4.8.3 Premissas / Pré-Condições

O usuário precisa estar logado, e a pesquisa e a composição de aminoácidos a ser modificada deve ter sido criada previamente.

#### A.4.8.4 Diagrama de Caso de Uso



**Figura A-15 UCD – Modificar composição de aminoácido**

#### A.4.8.5 Principais Cenários

**Cenário 1: Modificando com sucesso composição de aminoácidos dentro da visualização detalhada da pesquisa**

1. O usuário visualiza os detalhes da pesquisa;
2. O usuário seleciona a opção “Edit” na exibição dos dados da composição de aminoácidos.
3. O usuário é direcionado para uma tela de modificação da composição de aminoácidos.
4. O usuário preenche os dados solicitados e seleciona a opção salvar formulário.

5. Cada um dos aminoácidos é validado pela regra “Validação de Aminoácido” com sucesso.

6. O usuário é encaminhado para uma tela de sucesso na criação de composição de aminoácidos.

7. O caso de uso é encerrado com sucesso.

### **Cenário 2: Modificando com falha composição de aminoácidos dentro da visualização detalhada da pesquisa**

1. O usuário visualiza os detalhes da pesquisa;

2. O usuário seleciona a opção “Edit” na exibição dos dados da composição de aminoácidos.

3. O usuário é direcionado para uma tela de modificação da composição de aminoácidos.

4. O usuário preenche os dados solicitados e seleciona a opção salvar formulário.

5. Cada um dos aminoácidos é validado pela regra “Validação de Aminoácido” com falha.

6. O usuário é encaminhado para uma tela de falha na criação de composição de aminoácidos.

7. O caso de uso é encerrado com falha.

#### **A.4.8.6Regras**

##### **RN1: Validação de Aminoácido**

A regra será válida:

- Se a composição do ASX for maior que 0:

- a composição de ASN mais a composição de ASP deve ser maior que 0 e igual a composição de ASX.
- OU as composições de ASP e ASN não podem ter sido preenchidas.
- Se a composição do GLX for maior que 0:
  - a composição de GLN mais a composição de GLU deve ser maior que e igual a composição de GLX.
  - OU as composições de GLN e GLU não podem ter sido preenchidas.
- Se pelo menos uma das composições de aminoácidos for preenchida;
- Se houver nome da proteína de calibração:
  - A composição calibração de pelo menos um dos aminoácidos deve ser preenchida;
- Se houver alguma composição de calibração preenchida:
  - Deve haver o nome da proteína de calibração;

#### **A.4.9. Remover composição de aminoácidos**

##### **A.4.9.1 Descrição Detalhada**

Este caso de uso será utilizado para remover uma composição de aminoácidos criada.

##### **A.4.9.2 Atores**

- Usuário
- Banco de Dados

#### A.4.9.3 Premissas / Pré-Condições

O usuário precisa estar logado, e ter uma pesquisa com uma composição de aminoácidos criada previamente.

#### A.4.9.4 Diagrama de Caso de Uso

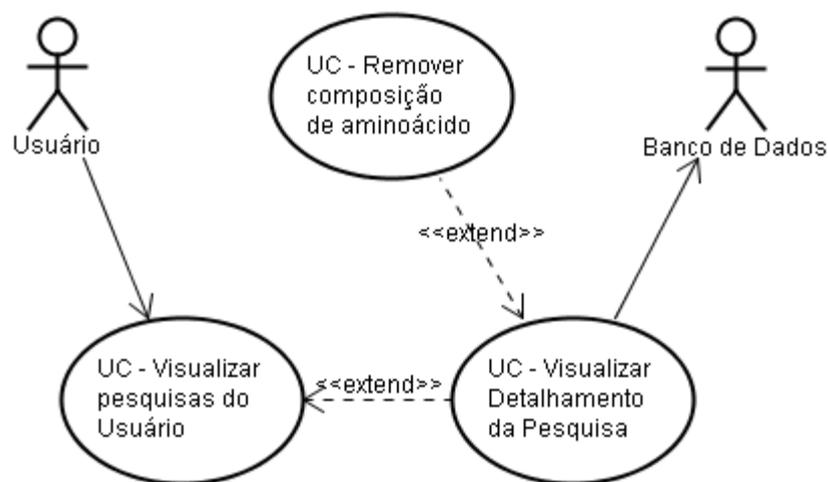


Figura A-16 UCD – Remover composição de aminoácido

#### A.4.9.5 Principais Cenários

**Cenário 1: Removendo com sucesso composição de aminoácidos dentro da visualização detalhada da pesquisa pelo modo básico**

1. O usuário visualiza o detalhamento da pesquisa
2. O usuário seleciona a opção REMOVE na composição de aminoácidos.
3. O conjunto de dados da composição de aminoácidos da pesquisa é removido do banco de dados com sucesso.
4. O usuário é encaminhado para uma tela de informando o sucesso da operação.

5. O caso de uso é encerrado com sucesso.

### **Cenário 2: Removendo com falha composição de aminoácidos dentro da visualização detalhada da pesquisa pelo modo básico**

1. O usuário visualiza o detalhamento da pesquisa
2. O usuário seleciona a opção REMOVE na composição de aminoácidos.
3. O conjunto de dados da composição de aminoácidos da pesquisa não é removido do banco de dados com sucesso, por problemas no banco de dados.
4. O usuário é encaminhado para uma tela informando a falha na operação.
5. O caso de uso é encerrado com falha.

#### **A.4.9.6Regras**

Não existem regras específicas para este caso de uso.

#### **A.4.10.Adicionar *fingerprint***

##### **A.4.10.1Descrição Detalhada**

*Peptide Mass Fingerprint* (PMF) é uma técnica utilizada para identificar proteínas em bancos de dados de seqüências sem a necessidade de seqüenciar a proteína em estudo. O usuário pode criar vários *fingerprints* para uma determinada pesquisa, enquanto estiver visualizando o detalhamento da mesma, utilizando o modo básico ou avançado.

No modo básico devem ser fornecidas informações a respeito do agente de clivagem, características de ionização dos peptídeos, tolerância e a lista de massas, dentre outros.

No modo avançado o usuário terá acesso às seguintes características:

- Possibilidade de definir um agente de clivagem diferente dos listados;
- Possibilidade de uso de modificações fixas ou variáveis (compatíveis com mascot);
- Possibilidade de definição de modificações não listadas;
- Definição do instrumento utilizado;
- Possibilidade de listagem de contaminantes;
- Possibilidade de busca de misturas de proteínas;
- Escolha de região de tradução;
- Uso de modo de homologia (compatível com prospector).

Estas funcionalidades não serão disponibilizadas no modo básico.

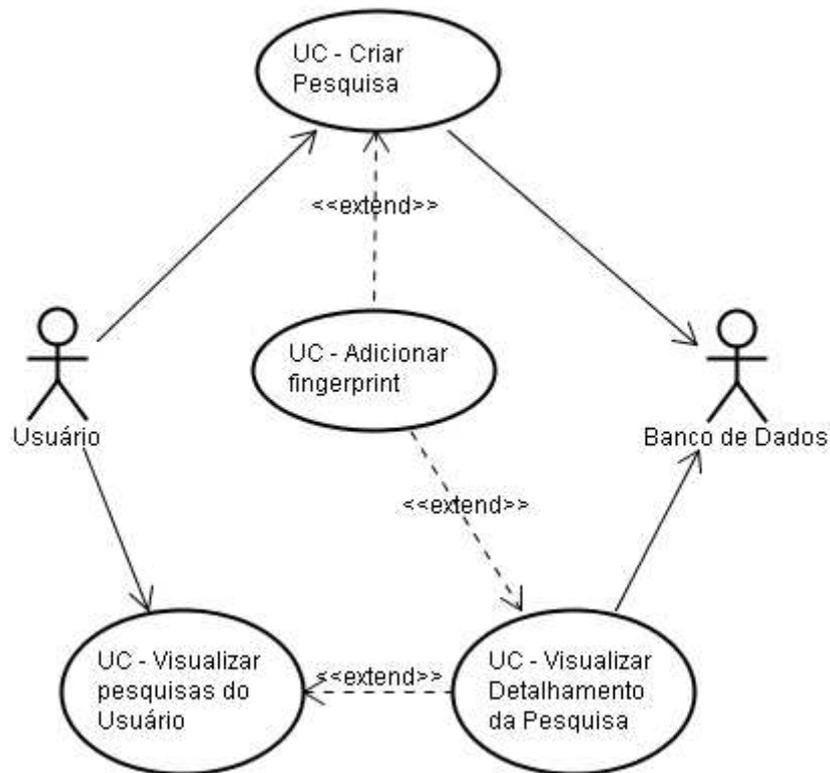
#### **A.4.10.2 Atores**

- Usuário
- Banco de Dados

#### **A.4.10.3 Premissas / Pré-Condições**

O usuário deve estar logado na aplicação. Além disso, para ter acesso a esta funcionalidade, o usuário deve estar criando ou editando uma pesquisa pré-cadastrada.

#### **A.4.10.4 Diagrama de Caso de Uso**



**Figura A-17 UCD - Adicionar *fingerprint***

#### A.4.10.5 Principais Cenários

**Cenário 1: Criando *fingerprint* dentro da visualização detalhada da pesquisa pelo modo básico**

1. O usuário visualiza os detalhes da pesquisa;
2. O usuário seleciona a opção “Criar *fingerprint*” pelo modo básico;
3. O usuário é direcionado para uma tela de cadastro de *fingerprints*.
4. O usuário preenche os dados solicitados e seleciona a opção salvar formulário.
5. Cada uma das massas é validada pela regra “Validação de *Fingerprint*” com sucesso.

6. O usuário é encaminhado para uma tela de sucesso na criação de composição de *fingerprint*.

7. O caso de uso é encerrado com sucesso.

#### **A.4.10.6Regras**

##### **RN1: Validação de *Fingerprint***

Para que um *fingerprint* seja considerado válido é necessário:

- Ter um, e somente um agente de clivagem;
- Selecionar um único estado de ionização;
- Selecionar uma faixa de tolerância, e a respectiva unidade;
- Conter pelo menos quatro entradas de massa molecular;
- Ter os dados das massas coerentes com o formato informado, da seguinte forma:
- Massa deve ser um valor numérico maior que 30
- Carga deve ser um valor numérico inteiro entre -10 e +10
- Intensidade deve ser um valor numérico maior que 1
- N-term deve ser um string de pelo menos um caractere composto pelas seguintes letras: A,B,C,D,E,F,G,H,I,K,L,M,N,P,Q,R,S,T,V,W,X,Y,Z, portanto não pode conter J,O,U
- Os componentes de formato não selecionados não devem estar presentes

#### A.4.10.7 Telas e interfaces

**Peptide Mass Fingerprint (basic form)**

Data for digestion #

Cleavage agent:

Modifications:

Peptides State:  [M+H]<sup>+</sup> or  [M] or  [M-H]<sup>-</sup>

Mass tolerance

Mass list:

Data Format:

Mass  Order(from 1 to 4):

Intensity  Order(from 1 to 4):

Charge  Order(from 1 to 4):

N-Term  Order(from 1 to 4):

Separator:

<i>Monoisotopic</i>	<i>Average</i>
Mass (m/z), Charge, N-term	Mass m/z, Charge, N-term
<input type="text"/>	<input type="text"/>

Peptides required for match:

Figura A-18 Formulário para inserir dados de *fingerprint*

## A.4.11.Modificar *Fingerprint*

### A.4.11.1Descrição Detalhada

Este caso de uso será utilizado para modificar os dados de um *fingerprint* criado previamente

### A.4.11.2Atores

- Usuário
- Banco de Dados

### A.4.11.3Premissas / Pré-Condições

O usuário precisa estar logado, e a pesquisa e o *fingerprint* a ser modificado devem ter sido criados previamente.

### A.4.11.4Diagrama de Caso de Uso

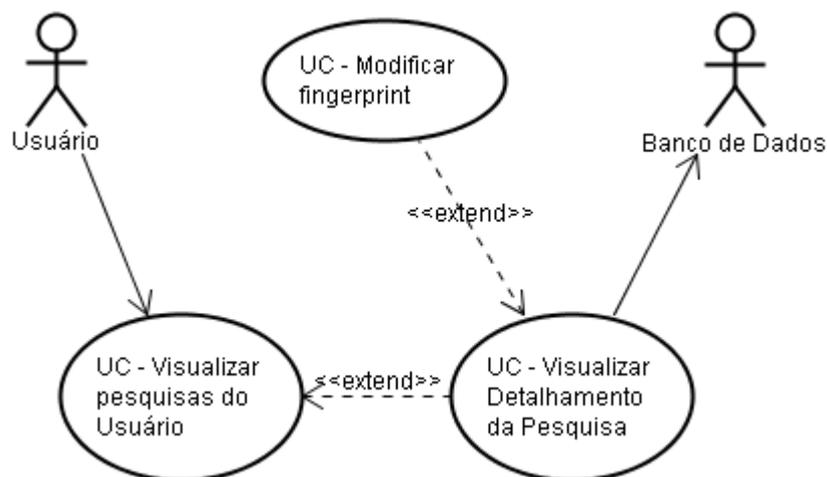


Figura A-19 UCD - Modificar *Fingerprint*

### A.4.11.5Principais Cenários

**Cenário 1: Modificando um *fingerprint* com sucesso através da visualização da pesquisa**

1. O usuário visualiza detalhes da pesquisa.
2. O usuário seleciona a opção ‘Editar’ no *fingerprint* que deseja modificar.
3. O usuário preenche o formulário com as modificações que deseja efetuar.
4. O usuário seleciona a opção ‘salvar’.
5. Os dados da *fingerprint* são validados de acordo com a regra “Validar *fingerprint*”.
6. O formulário é enviado para o banco de dados com sucesso.
7. O caso de uso é encerrado com sucesso.

#### **A.4.11.6 Regras**

Não existem regras específicas para este caso de uso.

### **A.4.12. Remover *fingerprint***

#### **A.4.12.1 Descrição Detalhada**

Este caso de uso será utilizado para remover um *fingerprint* criado previamente. Caso existam diversos conjuntos de dados de *fingerprint*, cada um deles deverá ser removido de forma independente.

#### **A.4.12.2 Atores**

- Usuário
- Banco de Dados

#### **A.4.12.3 Premissas / Pré-Condições**

O usuário precisa estar logado, e ter uma pesquisa com um *fingerprint* criado previamente.

#### A.4.12.4 Diagrama de Caso de Uso

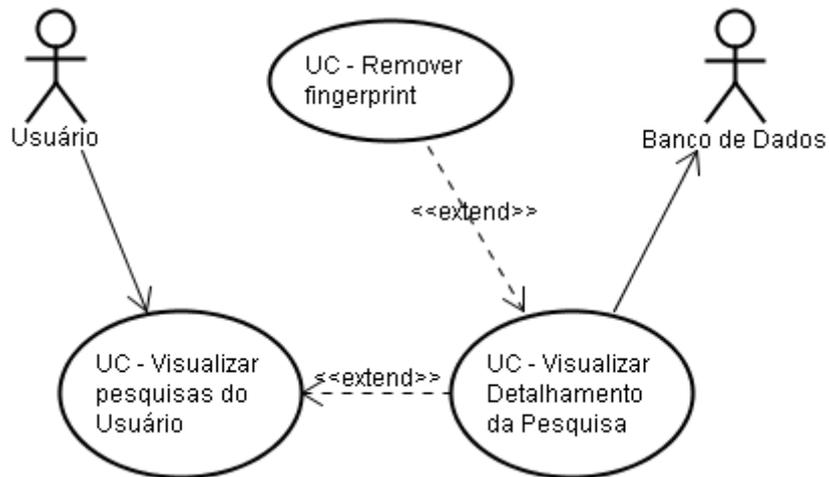


Figura A-20 UCD - Remover *fingerprint*

#### A.4.12.5 Principais Cenários

##### Cenário 1: Removendo um *fingerprint* com sucesso

1. O usuário visualiza o detalhamento da pesquisa
2. O usuário seleciona a opção REMOVE no *fingerprint* desejado.
3. O conjunto de dados de *fingerprint* é removido do banco de dados com sucesso.
4. O usuário é encaminhado para uma tela de informando o sucesso da operação.
5. O caso de uso é encerrado com sucesso.

#### A.4.12.6 Regras

Não existem regras para este caso de uso.

## A.4.13. Adicionar *sequence data*

### A.4.13.1 Descrição Detalhada

Seqüenciamento é uma técnica para identificação de proteínas com base na determinação de sua estrutura primária. Essa técnica é baseada na obtenção de dados de seqüências de uma proteína, ou de peptídeos obtidos de sua clivagem. No cadastro de *sequence data*, são informados os dados da seqüência e, em caso de clivagem, do peptídeo, são também cadastrados os métodos experimentais utilizados para fazer o mapeamento.

### A.4.13.2 Atores

- Usuário
- Banco de Dados

### A.4.13.3 Premissas / Pré-Condições

O usuário deve estar logado no sistema e estar editando uma pesquisa pré-cadastrada ou deve estar criando uma nova pesquisa.

### A.4.13.4 Diagrama de Caso de Uso

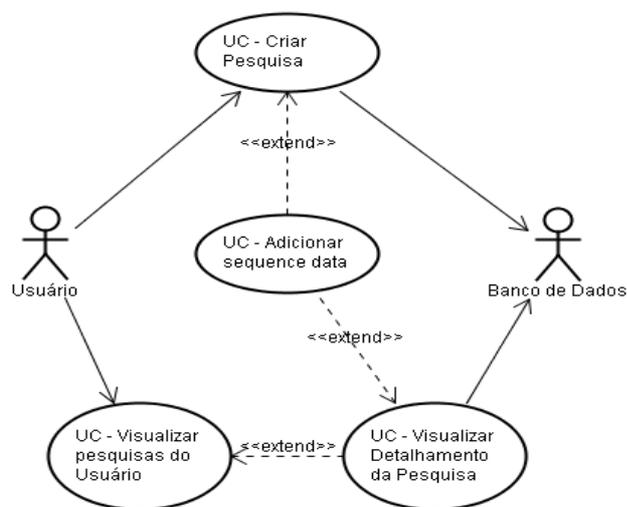


Figura A-21 UCD - Criar *sequence data*

#### **A.4.13.5 Principais Cenários**

##### **Cenário 1: Criando *sequence data* dentro da visualização detalhada da pesquisa**

1. O usuário visualiza os detalhes da pesquisa;
2. O usuário seleciona a opção “Criar *sequence data*”;
3. O usuário é direcionado para uma tela de cadastro de *sequence data*.
4. O usuário preenche os dados solicitados e seleciona a opção salvar formulário.
5. Os dados do formulário são validados pela regra “Validação de *Sequence data*” com sucesso.
6. O usuário é encaminhado para uma tela de sucesso na criação de *sequence data*.

#### **A.4.13.6 Regras**

##### **RN1: Validação de *Sequence data***

Para ser considerado válido, o *sequence data* deve:

- Ter cadastrado exclusivamente:
  - A seqüência obtida utilizando a representação padrão de aminoácidos;
  - OU a lista de massas dos fragmentos do peptídeo.
  - OU dados de seqüenciamento obtidos diretamente do espectrômetro de massa;
- Se for fornecida a lista de fragmentos, é obrigatório o preenchimento dos campos referentes ao precursor íon: peptide mass, unit; range; unit;

monoisotopic ou average; peptide charge e também os campos ms/ms tolerance; unit e monoisotopic/average

#### A.4.13.7 Telas e interfaces

##### Sequence (basic form):

N-terminal    C-terminal    Internal (From:  To:  )    Complete

Save this form

Clear this form

**Figura A-22 Formulário para inserir dados de seqüência de aminoácidos**

#### A.4.14. Modificar sequence data

##### A.4.14.1 Descrição Detalhada

Este caso de uso será utilizado para modificar um *sequence data* criado previamente.

##### A.4.14.2 Atores

- Usuário
- Banco de Dados

##### A.4.14.3 Premissas / Pré-Condições

O usuário precisa estar logado, e a pesquisa e o *sequence data* modificado devem ter sido criados previamente.

#### A.4.14.4 Diagrama de Caso de Uso

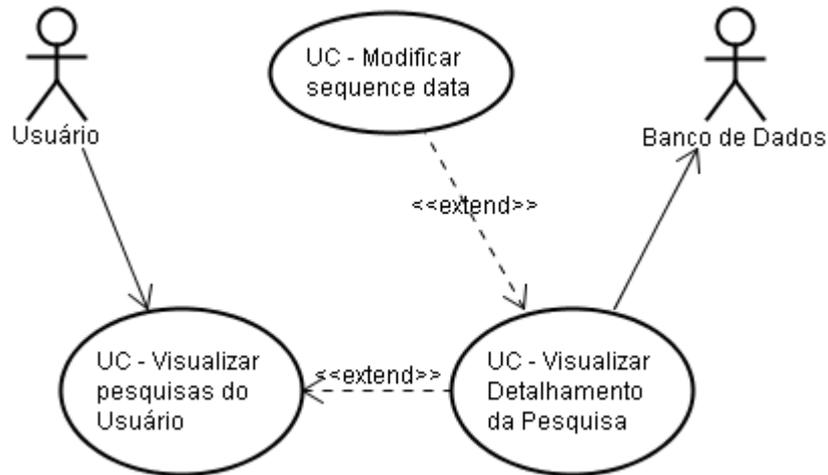


Figura A-23 UCD - Modificar *sequence data*

#### A.4.14.5 Principais Cenários

**Cenário 1: Modificando um *sequence data* com sucesso através da visualização da pesquisa**

1. O usuário visualiza detalhes da pesquisa.
2. O usuário seleciona a opção 'Editar' no *sequence data* que deseja modificar.
3. O usuário preenche o formulário com as modificações que deseja efetuar.
4. O usuário seleciona a opção 'salvar'.
5. Os dados da *sequence data* são validados de acordo com a regra "Validar *fingerprint*" no item 3.8.63.9.6.
6. O formulário é enviado para o banco de dados com sucesso.
7. O caso de uso é encerrado com sucesso.

#### A.4.14.6 Regras

Não existem regras específicas para este caso de uso.

#### A.4.15. Remover *sequence data*

##### A.4.15.1 Descrição Detalhada

Este caso de uso será utilizado para remover um *sequence data* criado previamente.

##### A.4.15.2 Atores

- Usuário
- Banco de Dados

##### A.4.15.3 Premissas / Pré-Condições

O usuário precisa estar logado, e ter uma pesquisa com um *sequence data* criada previamente.

##### A.4.15.4 Diagrama de Caso de Uso

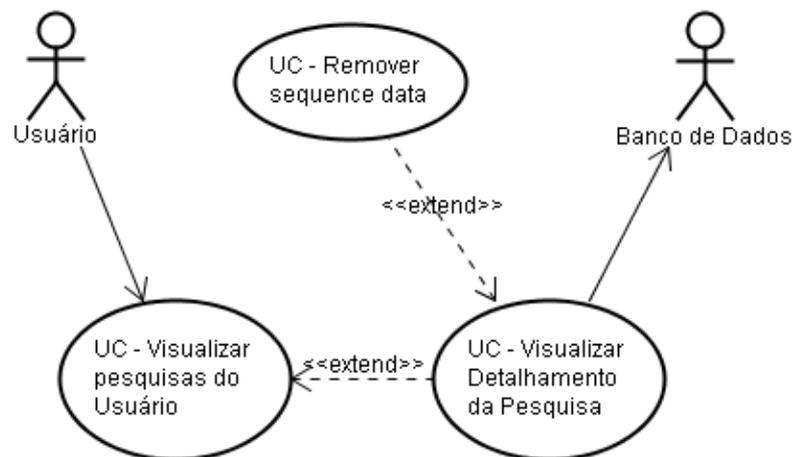


Figura A-24 UCD - Remover *sequence data*

#### **A.4.15.5 Principais Cenários**

##### **Cenário 1: Removendo um *sequence data* com sucesso**

1. O usuário visualiza o detalhamento da pesquisa
2. O usuário seleciona a opção REMOVE no *sequence data* desejado.
3. A *sequence data* é removida do banco de dados com sucesso.
4. O usuário é encaminhado para uma tela de informando o sucesso da operação.
5. O caso de uso é encerrado com sucesso.

#### **A.4.15.6 Regras**

Não existem regras para este caso de uso.

#### **A.4.16. Avaliar possíveis buscas de uma pesquisa**

##### **A.4.16.1 Descrição Detalhada**

O sistema Protein Locator deve ser capaz de avaliar a compatibilidade de uma pesquisa com os principais serviços de pesquisa proteômica (BLAST, MASCOT, AACompident e FASTA). A verificação de compatibilidade será feita baseada nas informações cadastradas na pesquisa.

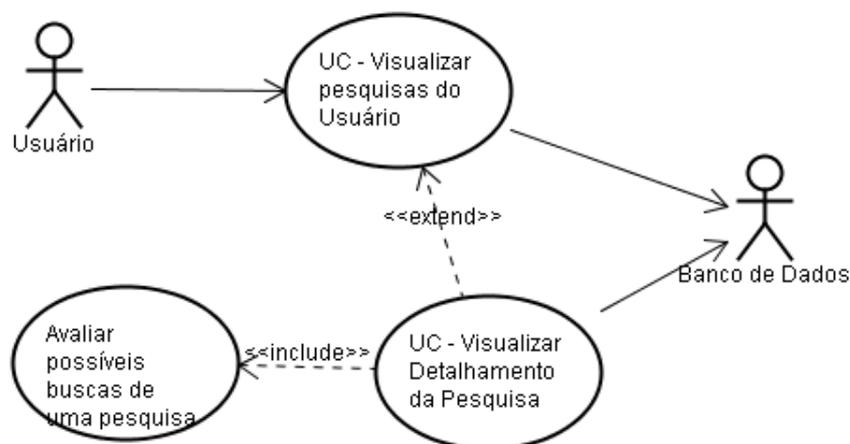
##### **A.4.16.2 Atores**

- Usuário
- Banco de Dados

##### **A.4.16.3 Premissas / Pré-Condições**

O usuário precisa estar logado, e ter uma pesquisa criada previamente.

#### A.4.16.4 Diagrama de Caso de Uso



**Figura A-25 UCD - Avaliar possíveis buscas de uma pesquisa**

#### A.4.16.5 Principais Cenários

Cenário Avaliando possíveis buscas de uma pesquisa

1. O usuário solicita a avaliação das possíveis buscas de uma pesquisa.
2. O sistema carrega a pesquisa no banco de dados.
3. O sistema verifica a compatibilidade da pesquisa com o serviço AACompident de acordo com a regra “Compatibilidade AACompident”.
4. O sistema verifica a compatibilidade da pesquisa com o serviço Blast de acordo com a regra “Compatibilidade Blast”.
5. O sistema verifica a compatibilidade da pesquisa com o serviço FASTA de acordo com a regra “Compatibilidade FASTA”
6. O sistema verifica a compatibilidade da pesquisa com o serviço MASCOT de acordo com a regra “Compatibilidade MASCOT”

7. O sistema retorna ao usuário a informação de quais mecanismos de busca são compatíveis com os dados da pesquisa selecionada.

8. O caso de uso é terminado com sucesso.

#### A.4.16.6 Regras

##### RN Compatibilidade AACompident

O sistema irá considerar a pesquisa compatível com o serviço AACompident se possuir composição de aminoácidos.

##### RN Compatibilidade Blast

O sistema irá considerar a pesquisa compatível com o serviço Blast se possuir pelo menos um *sequence data* cadastrado contendo pelo menos uma seqüência em formato FASTA.

##### RN Compatibilidade FASTA

O sistema irá considerar a pesquisa compatível com o serviço FASTA se possuir pelo menos um *sequence data* cadastrado contendo pelo menos uma seqüência em formato FASTA.

##### RN Compatibilidade MASCOT

O sistema irá considerar a pesquisa compatível com o serviço MASCOT se a pesquisa possuir pelo menos um *fingerprint* ou uma seqüência ou ainda um conjunto de fragmentos para seqüenciamento cadastrado.

#### A.4.16.7 Telas e interfaces

Please, choose the program(s) you would like to use in the search:

FASTA for digestion # 1

Blast for digestion # 1

Mascot for PMF # 1 (Your data from PMF # 1 could not be used in Mascot. To use this program, please edit the following data: **Enzyme;Fixed Modification;**)

Search

Figura A-26 Exibir compatibilidades e incompatibilidades para o usuário

## **A.4.17.Submeter Pesquisa para serviço de busca Proteômica**

### **A.4.17.1 Descrição Detalhada**

O Protein Locator deve ser capaz de submeter dados de uma pesquisa cadastrada para os serviços de busca proteômica, a saber:

- MASCOT
- BLAST
- AACompident
- FASTA

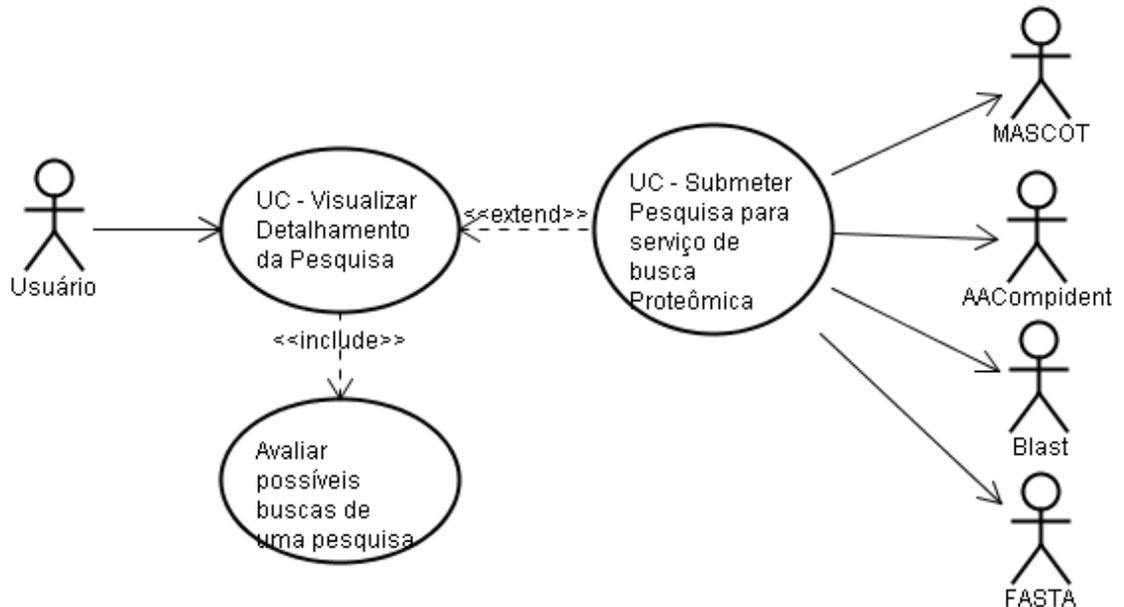
### **A.4.17.2 Atores**

- Usuário
- Banco de dados
- MASCOT
- BLAST
- AACompident
- FASTA

### **A.4.17.3 Premissas / Pré-Condições**

O usuário precisa estar logado na aplicação, possuir uma pesquisa cadastrada, e esta pesquisa deve ter sido considerada compatível com os serviços de busca (vide caso de uso Avaliar possíveis buscas de uma pesquisa).

#### A.4.17.4 Diagrama de Caso de Uso



**Figura A-27 UCD - Submeter Pesquisa para serviço de busca Proteômica**

#### A.4.17.5 Principais Cenários

##### **Cenário 1 Submetendo uma pesquisa para todos os serviços de busca**

1. O usuário seleciona quais serviços deverão receber os dados e solicita o envio da pesquisa.
2. O sistema verifica a compatibilidade da pesquisa com o primeiro serviço a receber os dados de acordo com a regra específica para esse serviço descrita no caso de uso “Avaliar possíveis buscas de uma pesquisa”.
3. Caso seja compatível, o PL carrega o formulário de submissão do serviço testado.
4. O sistema verifica se existem dados armazenados no banco de dados que precisem de processamento para o preenchimento de campos no formulário (ex:

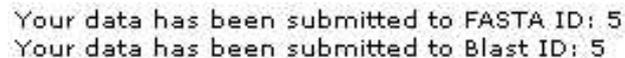
conversão de unidades, cálculos de composição, etc). Se existirem, tais processamentos são realizados.

5. O PL preenche o formulário com os dados da pesquisa.
6. O formulário é submetido ao serviço de busca.
7. Os itens 2, 3, 4 e 5 são repetidos para cada serviço solicitado no item 1
8. O caso de uso é encerrado com sucesso.

#### **A.4.17.6Regras**

Não existem regras de negócios específicas associadas a este caso de uso.

#### **A.4.17.7Telas e interfaces**



Your data has been submitted to FASTA ID: 5  
Your data has been submitted to Blast ID: 5

**Figura A-28 Aviso de submissão com sucesso**

#### **A.4.18.Receber resposta de pesquisa via WEB**

##### **A.4.18.1Descrição Detalhada**

O PL deve ser capaz de receber o resultado de uma pesquisa submetida, cuja resposta é dada através de uma página *WEB*. Neste caso, o sistema deve fazer um *parsing* do HTML recebido, e atualizar o banco de dados com o resultado.

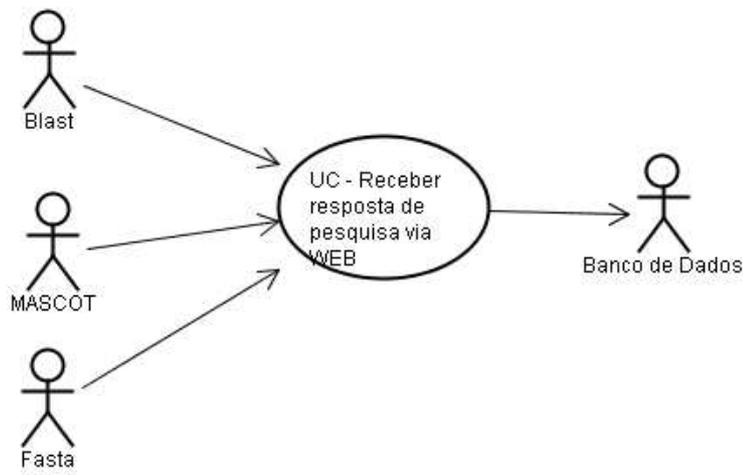
##### **A.4.18.2Atores**

- MASCOT
- BLAST
- FASTA
- Banco de Dados

#### A.4.18.3 Premissas / Pré-Condições

A pesquisa deve ter sido submetida previamente.

#### A.4.18.4 Diagrama de Caso de Uso



**Figura A-29 UCD - Receber resposta de pesquisa via WEB**

#### A.4.18.5 Principais Cenários

##### **Cenário 1 Recebendo a resposta do serviço Blast**

1. O Blast encaminha o PL para a pagina HTML com o resultado da busca.
2. O Sistema efetua o parse do HTML recebido.
3. As informações retiradas do formulário são persistidas no banco de dados.
4. O caso de uso é encerrado com sucesso.

#### A.4.18.6 Regras

Não existem regras de negócios específicas associadas a este caso de uso.

#### A.4.19. Receber resposta de pesquisa via *E-MAIL*

##### A.4.19.1 Descrição Detalhada

O PL deve ser capaz de receber o resultado de uma pesquisa submetida, cuja resposta é enviada através de *E-MAIL*. Neste caso, o sistema deve fazer uma filtragem dos dados do *E-MAIL* recebido, e atualizar o banco de dados com o resultado.

##### A.4.19.2 Atores

- AACompident
- Banco de Dados

##### A.4.19.3 Premissas / Pré-Condições

A pesquisa deve ter sido submetida previamente.

##### A.4.19.4 Diagrama de Caso de Uso



**Figura A-30 UCD - Receber resposta de pesquisa via *E-mail***

##### A.4.19.5 Principais Cenários

###### Cenário 1 Recebendo a resposta do serviço

1. O Serviço de busca submete o *e-mail* para o endereço da caixa que o Protein Locator está escutando.

2. O Protein Locator recebe o *e-mail*.
3. O PL efetua o parse do *e-mail*.
4. As informações retiradas do *e-mail* são persistidas no banco de dados.
5. O caso de uso é encerrado com sucesso.

#### **A.4.19.6 Regras**

Não existem regras de negócios específicas associadas a este caso de uso.

#### **A.4.20. Exibir Resultados consolidados**

##### **A.4.20.1 Descrição Detalhada**

O PL deve ser capaz de calcular o resultado consolidado com a probabilidade de erro de cada resultado apresentado pelos programas utilizados para busca.

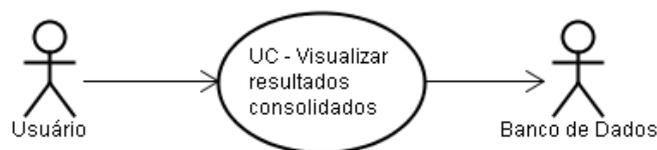
##### **A.4.20.2 Atores**

- Banco de Dados
- Usuário

##### **A.4.20.3 Premissas / Pré-Condições**

A pesquisa deve ter sido submetida previamente.

##### **A.4.20.4 Diagrama de Caso de Uso**



**Figura A-31 UCD – Exibir resultados consolidados**

#### A.4.20.5 Principais Cenários

##### Cenário 1 Exibindo resultados consolidados para o usuário

1. O usuário seleciona a opção resultados.
2. O sistema procura no banco de dados os resultados disponíveis para o usuário.
3. O PL calcula os resultados consolidados para cada uma das pesquisas submetidas do usuário.
4. As informações de identificação da proteína e o resultado consolidado são exibidas para o usuário.
5. O caso de uso é encerrado com sucesso.

#### A.4.20.6 Regras

Não existem regras de negócios específicas associadas a este caso de uso.

#### A.4.20.7 Telas e interfaces

20080528165054		
Protein Identifier at NCBI	Expect value	PL Score
P80674	4.3447121767802E-59	0.03176768073078
B0XAP8	1.0E-12	0.100000000001
Q7Q3Q1	4.0E-12	0.100000000004

**Figura A-32 Tabela com os resultados para cada pesquisa**

#### A.4.21. Exibir Resultados originais

##### A.4.21.1 Descrição Detalhada

O PL deve ser capaz de exibir os resultados originais apresentados pelos programas utilizados para busca.

#### A.4.21.2Atores

- Banco de Dados
- Usuário

#### A.4.21.3Premissas / Pré-Condições

A pesquisa deve ter sido submetida previamente.

#### A.4.21.4Diagrama de Caso de Uso



**Figura A-33 UCD – Exibir resultados originais**

#### A.4.21.5Principais Cenários

##### **Cenário 1 Exibindo resultados originais para o usuário**

1. O usuário, na tela de visualização de todas as pesquisas, seleciona o resultado que deseja consultar.
2. O sistema procura no banco de dados o disponível para o usuário.
3. As informações são exibidas para o usuário, conforme o programa utilizado para busca as enviou.
4. O caso de uso é encerrado com sucesso.

#### A.4.21.6Regras

Não existem regras de negócios específicas associadas a este caso de uso.

## A.4.21.7 Telas e interfaces

- Help
- General Help
- Formats
- Gaps
- Matrix
- References
- Fasta Help
- MView Help
- VisualFasta Help
- Database Information
- UniProt
- UniParc

### Fasta Summary Table

SUBMISSION PARAMETERS			
Title	45 89 aa	Database	swissprot
Sequence length	89	Sequence type	p
Program	fasta	Version	35.02 Feb. 18, 2008
Expectation upper value	10	Matrix	BL50
Sequence range	1-	Number of scores	10
Number of alignments	10	Word size	2
Open gap penalty	-2	Gap extension penalty	-1
Histogram	false		

Alignment	DB:ID	Source	Length	Identity%	Similar%	Overlap	E()
1 <input type="checkbox"/>	<a href="#">SW:RS20_PARDP</a>	30S ribosomal protein S20.	89	100.0	100.0	89	0.00054
2 <input type="checkbox"/>	<a href="#">SW:RS20_DINSH</a>	30S ribosomal protein S20.	88	80.0	90.0	90	0.027
3 <input type="checkbox"/>	<a href="#">SW:RS20_SILST</a>	30S ribosomal protein S20.	87	79.1	90.7	86	0.047
4 <input type="checkbox"/>	<a href="#">SW:RS20_RHOS4</a>	30S ribosomal protein S20.	92	76.3	88.2	93	0.047
5 <input type="checkbox"/>	<a href="#">SW:RS20_RHOS1</a>	30S ribosomal protein S20.	92	76.3	88.2	93	0.047
6 <input type="checkbox"/>	<a href="#">SW:RS20_JANSC</a>	30S ribosomal protein S20.	88	78.7	88.8	89	0.052
7 <input type="checkbox"/>	<a href="#">SW:RS20_SILPD</a>	30S ribosomal protein S20.	87	77.9	90.7	86	0.058

**Figura A-34 Resultado original do programa FASTA**

## **B.– DOCUMENTAÇÃO DO BANCO DE DADOS**

### **B.1 VISÃO GERAL DO DOCUMENTO**

Este documento visa detalhar o banco de dados da aplicação “Protein Locator”, definindo sua estrutura, suas entidades e seus relacionamentos.

A documentação foi dividida da seguinte forma:

- Modelo Entidade-Relacionamento;
- Entidades, cada uma com uma descrição, uma visão e detalhamento de seus atributos.

### **B.2 ABREVIATURAS UTILIZADAS**

BD – Banco de Dados

FK – Chave estrangeira

MER – Modelo Entidade Relacionamento

PK – Chave primária

PL – Protein Locator



## B.4 ENTIDADES

### B.4.1.Users

#### B.4.1.1Descrição Detalhada

Tabela com os dados referentes aos pesquisadores usuários do PL. Contém o nome completo do usuário, seu *e-mail* (que é utilizado com identificador do usuário para *login*) e sua senha.

#### B.4.1.2Visão da tabela

users		
ID	int(11)	<pk>
name	varchar(60)	
password	varchar(255)	
email	varchar(60)	

**Figura B-2 Visão da tabela “users”**

#### B.4.1.3Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (11), auto incrementável.
- name
  - descrição: nome completo do usuário.
  - tipo: varchar (60)
- *password*
  - descrição: hash (utilizando o algoritmo SHA-1) da senha do usuário.
  - tipo: varchar (255)

- *e-mail*
  - descrição: endereço de *e-mail* do usuário, utilizado como *login* no sistema PL.
  - tipo: varchar (60)

## B.4.2.Generic

### B.4.2.1Descrição Detalhada

Tabela com os dados do formulário generic. Cada entrada nesta tabela é associada ao usuário que a criou e é chamada de “pesquisa” (query) pelo sistema. Cada usuário pode criar uma ou mais pesquisas. Cada pesquisa contém os dados gerais de uma amostra de proteínas, como o ponto isoelétrico, a massa, classificação taxonômica da espécie que produziu a amostra, palavras-chaves e comentários.

### B.4.2.2Visão da tabela

generic		
ID	int(11)	<pk>
user_id	int(11)	<ak,fk1>
query_name	varchar(30)	
pi	float	
pi_range	float	
pi_range_unit	varchar(8)	
pi_weight	float	
mw	float	
mw_range	float	
mw_range_unit	varchar(8)	
mw_weight	float	
taxon	text	
taxon_mascot	varchar(255)	<fk2>
keywords	text	
comments	text	

Figura B-3 Visão da tabela “generic”

### B.4.2.3Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (11), auto-incrementável

- user\_id
  - descrição: chave estrangeira, relacionando à tabela “users” a fim de identificar o usuário da pesquisa.
  - tipo: int (11)
  
- query\_name
  - descrição: identificador personalizado da pesquisa, pode-se inserir qualquer texto desejado para identificar a pesquisa do usuário.
  - tipo: varchar (30)
  
- pi
  - descrição: ponto isoelétrico da proteína. Pode ser usado como um pré-filtro das proteínas nos bancos de dados utilizados para identificação.
  - tipo: float
  
- pi\_range
  - descrição: faixa de tolerância para o ponto isoelétrico.
  - tipo: float
  
- pi\_range\_unit
  - descrição: unidade de medida utilizada para a variação do ponto isoelétrico, que pode ser percentual ou em unidades de pI.
  - tipo: varchar (8)
  
- pi\_weight
  - descrição: peso estatístico dado ao ponto isoelétrico
  - tipo: float

- mw
  - descrição: molecular weight (massa molecular) da proteína em Daltons. Pode ser usado como um pré-filtro das proteínas nos bancos de dados utilizados para identificação.
  - tipo: float
- mw\_range
  - descrição: faixa de tolerância para a massa molecular
  - tipo: float
- mw\_range\_unit
  - descrição: unidade de medida utilizada para a variação da massa molecular.
  - tipo: varchar (8)
- mw\_weight
  - descrição: peso estatístico dado à massa molecular
  - tipo: float
- taxon
  - descrição: classificação taxonômica da espécie que esteja sendo pesquisada, de acordo com a classificação taxonômica do NCBI (<http://www.ncbi.nlm.nih.gov/Taxonomy/CommonTree/wwwcmt.cgi>)
  - tipo: text

- taxon\_mascot
  - descricao: classificação taxonômica da espécie pesquisada, de acordo com a classificação utilizada pelo programa Mascot.
  - Tipo: varchar(255)
  
- keywords
  - descrição: palavras-chaves utilizadas para caracterizar genericamente uma pesquisa de identificação da proteína no banco de dados
  - tipo: text
  
- comments
  - descrição: campo para informações úteis que não se enquadram em nenhum dos outros campos.
  - tipo: text

### **B.4.3.Submitted**

#### **B.4.3.1Descrição Detalhada**

Tabela com os dados referentes às pesquisas já submetidas pelos usuários do PL. Ao submeter uma pesquisa para identificação de proteínas, é criada uma entrada na tabela “submitted”, marcando esta entrada como não processada. Os robôs de submissão e recepção de resultados utilizam esta tabela para fazer a submissão e persistir os dados recebidos dos programas de identificação no banco de dados, associando-os à pesquisa do usuário. Portanto, nesta tabela são armazenados todas as informações relacionadas a submissão de dados para identificação, sendo utilizada durante a identificação e depois, para armazenar os resultados originais.

### B.4.3.2 Visão da tabela

submitted		
ID	int(8)	<pk>
generic_id	int(8)	<ak1, fk1>
service_id	int(8)	<ak2, fk2>
query_identifier	varchar(50)	<ak3>
id_reference_table	int(8)	
submission_date	bigint(14)	
returned_file	longblob	
query_status	varchar(15)	

**Figura B-4 Visão da tabela “submitted”**

### B.4.3.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (8), auto-incrementável
- generic\_id
  - descrição: chave estrangeira, relacionando à tabela “generic” a fim de identificar a pesquisa que foi submetida.
  - tipo: int (8)
- service\_id
  - descrição: chave estrangeira, relacionando à tabela “services” a fim de identificar qual o serviço a ser utilizado na busca.
  - tipo: int(8)
- query\_identifier
  - descrição: identificação da pesquisa, contendo data de submissão e identificador da pesquisa na tabela “submitted”.

- tipo: varchar (50)
- id\_reference\_table
  - descrição: identificador na tabela que contém os dados a serem submetidos.
  - tipo: int (8)
- submission\_date
  - descrição: data de submissão da pesquisa.
  - tipo: bigint (14)
- returned\_file
  - descrição: arquivo original retornado pelo programa utilizado para identificação.
  - tipo: longblob
- query\_status
  - descrição: identificador da situação da busca. Opções possíveis: not processed, processing, processed, error.
  - tipo: varchar(15)

#### **B.4.4.Aminoacid**

##### **B.4.4.1Descrição Detalhada**

Tabela com os dados referentes Esta composição é utilizada por alguns programas para identificar proteínas de acordo com a semelhança com a composição teórica de proteínas nos bancos de dados. São utilizados dados da composição em percentual ou peso molecular de aminoácidos das amostras. Se for utilizada um proteína de calibração nos procedimentos de análise da composição de aminoácidos, a composição desta proteína pode ser utilizada para compensar os erros inerentes à busca

resultados com os dados obtidos. Para tanto, é necessário especificar o nome da proteína de calibração (por meio de seu Swiss-Prot ID name) e preencher os dados obtidos experimentalmente de sua composição de aminoácidos.

#### B.4.4.2 Visão da tabela

aminoacid		
ID	int(11)	<pk>
generic_id	int(11)	<ak1,ak2,fk>
composition_type	char(8)	
calibration_protein	char(40)	
frag_window	int(6)	
ala_comp	decimal(5,2)	
arg_comp	decimal(5,2)	
asn_comp	decimal(5,2)	
asp_comp	decimal(5,2)	
asx_comp	decimal(5,2)	
cys_comp	decimal(5,2)	
gln_comp	decimal(5,2)	
glu_comp	decimal(5,2)	
glx_comp	decimal(5,2)	
gly_comp	decimal(5,2)	
his_comp	decimal(5,2)	
ile_comp	decimal(5,2)	
leu_comp	decimal(5,2)	
lys_comp	decimal(5,2)	
met_comp	decimal(5,2)	
phe_comp	decimal(5,2)	
pro_comp	decimal(5,2)	
ser_comp	decimal(5,2)	
thr_comp	decimal(5,2)	
trp_comp	decimal(5,2)	
tyr_comp	decimal(5,2)	
val_comp	decimal(5,2)	
ala_weight	decimal(5,3)	
arg_weight	decimal(5,3)	
asn_weight	decimal(5,3)	
asp_weight	decimal(5,3)	
asx_weight	decimal(5,3)	
cys_weight	decimal(5,3)	
gln_weight	decimal(5,3)	
glu_weight	decimal(5,3)	
glx_weight	decimal(5,3)	
gly_weight	decimal(5,3)	
his_weight	decimal(5,3)	
ile_weight	decimal(5,3)	
leu_weight	decimal(5,3)	
lys_weight	decimal(5,3)	
met_weight	decimal(5,3)	
phe_weight	decimal(5,3)	
pro_weight	decimal(5,3)	
ser_weight	decimal(5,3)	
thr_weight	decimal(5,3)	
trp_weight	decimal(5,3)	
tyr_weight	decimal(5,3)	
val_weight	decimal(5,3)	
ala_calibr	decimal(5,2)	
arg_calibr	decimal(5,2)	
asn_calibr	decimal(5,2)	
asp_calibr	decimal(5,2)	
asx_calibr	decimal(5,2)	
cys_calibr	decimal(5,2)	
gln_calibr	decimal(5,2)	
glu_calibr	decimal(5,2)	
glx_calibr	decimal(5,2)	
gly_calibr	decimal(5,2)	
his_calibr	decimal(5,2)	
ile_calibr	decimal(5,2)	
leu_calibr	decimal(5,2)	
lys_calibr	decimal(5,2)	
met_calibr	decimal(5,2)	
phe_calibr	decimal(5,2)	
pro_calibr	decimal(5,2)	
ser_calibr	decimal(5,2)	
thr_calibr	decimal(5,2)	
trp_calibr	decimal(5,2)	
tyr_calibr	decimal(5,2)	
val_calibr	decimal(5,2)	

Figura B-5 Visão da tabela “aminoacid”

#### B.4.4.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (11), auto-incrementável
- generic\_id
  - descrição: chave estrangeira, relacionando à tabela “generic” a fim de identificar a pesquisa do usuário.
  - tipo: int (11)
- composition\_type
  - descrição: tipo de composição de aminoácido: percentual molar ou número de resíduos por seqüência.
  - tipo: char (8)
- calibration\_protein
  - descrição: nome da proteína de calibração, que pode ter sido utilizada no procedimento de análise dos aminoácidos, de acordo com o Swiss-Prot ID.
  - tipo: char (40)
- frag\_window
  - descrição: tamanho da janela em que serão identificados os fragmentos.
  - tipo: int (6)

- ala\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da alanina identificada na amostra.
  - tipo: varchar (10)
  
- arg\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da arginina identificada na amostra.
  - tipo: varchar (10)
  
- asn\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da asparagina identificada na amostra.
  - tipo: varchar (10)
  
- asp\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) de aspartato identificada na amostra.
  - tipo: varchar (10)
  
- asx\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da asparagina e/ou aspartato identificada na amostra.
  - tipo: varchar (10)

- cys\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da cisteína identificada na amostra.
  - tipo: varchar (10)
  
- gln\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da glutamina identificada na amostra.
  - tipo: varchar (10)
  
- glu\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) de glutamato identificada na amostra.
  - tipo: varchar (10)
  
- glx\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da glutamina e/ou glutamato identificada na amostra.
  - tipo: varchar (10)
  
- gly\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da glicina identificada na amostra.
  - tipo: varchar (10)

- his\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da histidina identificada na amostra.
  - tipo: varchar (10)
  
- ile\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da isoleucina identificada na amostra.
  - tipo: varchar (10)
  
- leu\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da leucina identificada na amostra.
  - tipo: varchar (10)
  
- lys\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da lisina identificada na amostra.
  - tipo: varchar (10)
  
- met\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da metionina identificada na amostra.
  - tipo: varchar (10)

- phe\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da fenilalanina identificada na amostra.
  - tipo: varchar (10)
  
- pro\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da prolina identificada na amostra.
  - tipo: varchar (10)
  
- ser\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da serina identificada na amostra.
  - tipo: varchar (10)
  
- thr\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da treonina identificada na amostra.
  - tipo: varchar (10)
  
- trp\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da triptofano identificada na amostra.
  - tipo: varchar (10)

- tyr\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da tirosina identificada na amostra.
  - tipo: varchar (10)
  
- val\_comp
  - descrição: quantidade (em percentual molar ou número de resíduos) da valina identificada na amostra.
  - tipo: varchar (10)
  
- ala\_weight
  - descrição: peso estatístico dado à composição de alanina.
  - tipo: varchar (10)
  
- arg\_weight
  - descrição: peso estatístico dado à composição de arginina.
  - tipo: varchar (10)
  
- asn\_weight
  - descrição: peso estatístico dado à composição de asparagina.
  - tipo: varchar (10)
  
- asp\_weight
  - descrição: peso estatístico dado à composição de aspartato.
  - tipo: varchar (10)

- `asx_weight`
  - descrição: peso estatístico dado à composição de aspartato/aparagina.
  - tipo: `varchar (10)`
  
- `cys_weight`
  - descrição: peso estatístico dado à composição de cisteína.
  - tipo: `varchar (10)`
  
- `gln_weight`
  - descrição: peso estatístico dado à composição de glutamina.
  - tipo: `varchar (10)`
  
- `glu_weight`
  - descrição: peso estatístico dado à composição de glutamato.
  - tipo: `varchar (10)`
  
- `glx_weight`
  - descrição: peso estatístico dado à composição de glutamato/glutamina.
  - tipo: `varchar (10)`
  
- `gly_weight`
  - descrição: peso estatístico dado à composição de glicina.
  - tipo: `varchar (10)`

- his\_weight
  - descrição: peso estatístico dado à composição de histidina.
  - tipo: varchar (10)
  
- ile\_weight
  - descrição: peso estatístico dado à composição de isoleucina.
  - tipo: varchar (10)
  
- leu\_weight
  - descrição: peso estatístico dado à composição de leucina.
  - tipo: varchar (10)
  
- lys\_weight
  - descrição: peso estatístico dado à composição de lisina.
  - tipo: varchar (10)
  
- met\_weight
  - descrição: peso estatístico dado à composição de metionina.
  - tipo: varchar (10)
  
- phe\_weight
  - descrição: peso estatístico dado à composição de fenilalanina.
  - tipo: varchar (10)
  
- pro\_weight
  - descrição: peso estatístico dado à composição de prolina.

- tipo: varchar (10)
- ser\_weight
  - descrição: peso estatístico dado à composição de serina.
  - tipo: varchar (10)
- thr\_weight
  - descrição: peso estatístico dado à composição de treonina.
  - tipo: varchar (10)
- trp\_weight
  - descrição: peso estatístico dado à composição de triptofano.
  - tipo: varchar (10)
- tyr\_weight
  - descrição: peso estatístico dado à composição de tirosina.
  - tipo: varchar (10)
- val\_weight
  - descrição: peso estatístico dado à composição de valina.
  - tipo: decimal (5,3)
- ala\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de alanina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)

- arg\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de arginina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- asn\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de asparagina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- asp\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de aspartato obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- asx\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de asparagina/aspartato obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- cys\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de cisteína obtida experimentalmente para a proteína usada como calibrante.

- tipo: decimal (5,3)
- gln\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de glutamina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- glu\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de glutamato obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- glx\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de glutamina/glutamato obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- gly\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de glicina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)

- his\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de histidina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- ile\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de isoleucina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- leu\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de leucina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- lys\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de lisina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- met\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de metionina obtida experimentalmente para a proteína usada como calibrante.

- tipo: decimal (5,3)
- phe\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de fenilalanina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- pro\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de prolina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- ser\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de serina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- thr\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de treonina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)

- trp\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de triptofano obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- tyr\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de tirosina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)
- val\_calibr
  - descrição: quantidade (em percentual molar ou número de resíduos) de valina obtida experimentalmente para a proteína usada como calibrante.
  - tipo: decimal (5,3)

### **B.4.5.Fingerprint**

#### **B.4.5.1Descrição Detalhada**

Tabela com os dados referentes aos experimentos de PMF (*Peptide Mass Fingerprint*). Nesta técnica, a proteína desconhecida, que será analisada, sofre clivagens por meio de uma protease, comumente a tripsina. O conjunto resultante de peptídeos forma uma identificação única da proteína desconhecida. As massas desses peptídeos podem ser obtidas por meio de uma análise em espectrômetro de massa e devem ser comparadas com as massas de peptídeos em bancos de dados de proteínas para serem identificadas.

### B.4.5.2 Visão da tabela

fingerprint		
ID	int(11)	<pk>
generic_id	int(11)	<ak,fk>
clv_agt_type	varchar(30)	
ord_clv_agt	varchar(30)	
user_clv_term	char(6)	
user_clv_site	varchar(30)	
user_clv_excl	varchar(30)	
coupled_modif	varchar(30)	
missed_clv	int(3)	
fixed_modif	varchar(50)	
variable_modif	varchar(50)	
user_defined_site1	varchar(30)	
user_defined_modif1	varchar(30)	
user_defined_site2	varchar(30)	
user_defined_modif2	varchar(30)	
peptide_state	char(3)	
mass_tolerance	decimal(6,4)	
mass_tolerance_unit	varchar(4)	
instrument	varchar(15)	
mass_list_monoiso	text	
mass_file_monoiso	mediumblob	
mass_list_avg	text	
mass_file_avg	mediumblob	
contaminant_masses	text	
search	char(1)	
dna_frame_trans	int(1)	
pep_required_match	int(3)	
pep_shift	varchar(5)	
da	decimal(6,4)	
min_matches	int(3)	
mass	int(1)	
intensity	int(1)	
charge	int(1)	
nterm	int(1)	
separator	char(1)	

Figura B-6 Visão da tabela “*fingerprint*”

### B.4.5.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (11), auto-incrementável
- generic\_id
  - descrição: chave estrangeira, relacionando à tabela “generic” a fim de identificar a pesquisa do usuário.
  - tipo: int (11)

- `clv_agt_type`
  - descrição: categoria de agente de clivagem (enzima utilizada para clivagem da proteína em peptídeos e obtenção de seu *fingerprint*). Pode ser uma enzima pré-definida, dentre uma lista de enzimas, ou uma enzima definida pelo usuário.
  - tipo: `varchar (30)`
  
- `ord_clv_agt`
  - descrição: nome da enzima utilizada para clivagem, caso a opção tenha sido por escolher na lista pré-definida.
  - tipo: `varchar (30)`
  
- `user_clv_term`
  - descrição: tipo de clivagem da enzima utilizada e definida pelo usuário (clivagem N-terminal ou C-terminal).
  - tipo: `char (3)`
  
- `user_clv_site`
  - descrição: pontos (aminoácidos) em que a enzima definida pelo usuário faz a clivagem.
  - tipo: `varchar (30)`
  
- `user_clv_excl`
  - descrição: pontos (aminoácidos) que impedem a clivagem da enzima definida pelo usuário faz a clivagem.
  - tipo: `varchar (30)`

- coupled\_modif
  - descrição: modificações acopladas a modificação definida pelo usuário.
  - tipo: varchar (30)
  
- missed\_clv
  - descrição: quantidade de clivagens perdidas. Quanto maior a quantidade, menor a chance de sucesso do experimento.
  - tipo: int (3)
  
- fixed\_modif
  - descrição: modificações pós-traducionais fixas, ou seja, que sempre acontecem com aqueles dados do experimento.
  - tipo: varchar (50)
  
- variable\_modif
  - descrição: modificações pós-traducionais variáveis, ou seja, que nem sempre acontecem com aqueles dados do experimento.
  - tipo: varchar (50)
  
- user\_defined\_site1
  - descrição: ponto em que ocorrem modificações pós-traducionais, que podem ser fixas ou variáveis, definidas pelo usuário.
  - tipo: varchar (30)

- user\_defined\_modif1
  - descrição: modificações pós-traducionais, que podem ser fixas ou variáveis, definidas pelo usuário.
  - tipo: varchar (30)
  
- user\_defined\_site2
  - descrição: ponto em que ocorrem modificações pós-traducionais, que podem ser fixas ou variáveis, definidas pelo usuário.
  - tipo: varchar (30)
  
- user\_defined\_modif2
  - descrição: modificações pós-traducionais, que podem ser fixas ou variáveis, definidas pelo usuário.
  - tipo: varchar (30)
  
- peptide\_state
  - descrição: identifica se os dados do *fingerprint* incluem a massa do íon, MH+, MH-, ou apenas o neutro, M.
  - tipo: char (3)
  
- mass\_tolerance
  - descrição: janela para tolerância de erros nos valores de massa dos peptídeos.
  - tipo: decimal(6,4)

- mass\_tolerance\_unit
  - descrição: unidade de medida para a tolerância de massa. Pode ser em Da, ppm, %, Th, mmu.
  - tipo: varchar (4)
  
- instrument
  - descrição: instrumento utilizado para o experimento de espectrometria de massa. Pode ser escolhido algum dentro da lista de equipamentos, ou a opção de outro equipamento.
  - tipo: varchar (15)
  
- mass\_list\_monoiso
  - descrição: lista com os valores de massa monoisotópica dos peptídeos, um valor por linha. Opcionalmente, pode-se adicionar os valores de intensidade, carga ou “N-Terminal”
  - tipo: text
  
- mass\_list\_avg
  - descrição: lista com os valores de massa média dos peptídeos, um valor por linha. Opcionalmente, pode-se adicionar os valores de intensidade, carga ou “N-Terminal”.
  - tipo: text
  
- contaminant\_masses
  - descrição: Massa dos contaminantes presentes na amostra.
  - tipo: text

- search
  - descrição: define se a busca será realizada por apenas uma proteína, ou mistura de duas, três ou quatro.
  - tipo: char (1)
  
- dna\_frame\_trans
  - descrição: Refere-se à janela de tradução do DNA, pois pode-se começar a tradução em pontos diferentes do códon de 3 nucleotídeos.
  - tipo: int (1)
  
- pep\_required\_match
  - descrição: Número de peptídeos necessários para que seja considerado um match
  - tipo: varchar (3)
  
- pep\_shift
  - descrição: Refere-se ao tipo de desvio esperado na massa de peptídeos caso seja usado o modo de homologia.
  - tipo: varchar (5)
  
- da
  - descrição: Refere-se à massa de desvio esperado na massa de peptídeos caso seja usado o modo de homologia.
  - tipo: decima (6,4)

- `min_matches`
  - descrição: Número mínimo de matches sem substituição de aminoácidos par que seja considerado um acerto no modo de homologia do Prospector.
  - tipo: `int (3)`
  
- `mass`
  - descrição: indicador de que o valor de massa está presente na informação do campo “`mass_list`”.
  - tipo: `int (1)`
  
- `intensity`
  - descrição: indicador de que a intensidade está presente na informação do campo “`mass_list`”.
  - tipo: `int (1)`
  
- `charge`
  - descrição: indicador de que a carga está presente na informação do campo “`mass_list`”.
  - tipo: `int (1)`
  
- `nterm`
  - descrição: indicador de que os resíduos da seqüência amino-terminal estão presente na informação do campo “`mass_list`”
  - tipo: `int (1)`

- separator
  - descrição: caracter utilizado como separador entre os dados informados no campo de lista de massas.
  - tipo: char (1)

## B.4.6. Protein

### B.4.6.1 Descrição Detalhada

Tabela com os dados referentes a seqüência de proteínas. Para cada uma das pesquisas cadastradas, o usuário pode inserir informações de seqüência de proteína.

### B.4.6.2 Visão da tabela

protein		
ID	int(11)	<pk>
generic_id	int(11)	<ak, fk>
fixed_modif	varchar(255)	
variable_modif	varchar(255)	
sequence	varchar(12)	
internal_from	varchar(4)	
internal_to	varchar(4)	
seq_list	text	
gap_penalty_open	varchar(10)	
gap_penalty_extended	varchar(10)	
open_query	varchar(10)	
matrix	varchar(8)	
expect	varchar(10)	
word_size	int(2)	
dist_deviation	varchar(10)	
conserved_domain	int(1)	
filter_low	int(1)	
filter_mask	int(1)	
filter_mask_lower	int(1)	
max_mismatched_aas	int(4)	

Figura B-7 Visão da tabela “protein”

### B.4.6.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (11), auto-incrementável
- generic\_id
  - descrição: chave estrangeira, relacionando à tabela “generic” a fim de identificar a pesquisa que foi submetida.

- tipo: int (11)
- fixed\_modif
  - descrição: modificações pos-traducionais que a proteína possa ter passado, fixas, ou seja, sempre ocorrem.
  - tipo: varchar (255)
- variable\_modif
  - descrição: modificações pos-traducionais que a proteína possa ter passado, variáveis, ou seja, que nem sempre ocorrem.
  - tipo: varchar (255)
- sequence
  - descrição: tipo de sequência, se N-terminal ou C-terminal
  - tipo: varchar (12)
- internal\_from
  - descrição: parâmetro inicial para pegar uma sub-sequência dentro da sequência completa inserida no campo de sequência.
  - tipo: varchar (4)
- internal\_to
  - descrição: parâmetro final para pegar uma sub-sequência dentro da sequência completa inserida no campo de sequência.
  - tipo: varchar (4)
- seq\_list
  - descrição: sequência de aminoácidos da amostra.

- tipo: text
- gap\_penalty\_open
  - descrição: penalidade para o primeiro residuo encontrado em um espaço (gap).
  - tipo: decimal(4,2)
- gap\_penalty\_extended
  - descrição: penalidade para os resíduos adicionais encontrados em um espaço (gap).
  - tipo: decimal(4,2)
- open\_query
  - descrição: modificações pos-traducionais que a proteína possa ter passado, variaveis, ou seja, que nem sempre ocorrem.
  - tipo: decimal(4,2)
- matrix
  - descrição: matriz de substituição para ser usada pelo algoritmo do programa de identificação.
  - tipo: varchar (8)
- expect
  - descrição: valor limite para o e-value.
  - tipo: decimal(5,4)

- word\_size
  - descrição: parâmetro que influencia na sensibilidade das buscas no Blast, definindo o tamanho da palavra que será utilizada na busca.
  - tipo: int (2)
  
- dist\_deviation
  - descrição: distancia percentual máxima aceitável entre segmentos do peptídeo.
  - tipo: decimal(5,4)
  
- conserved\_domain
  - descrição: característica da seqüência que possui funções próprias, podendo ser separada da proteína.
  - tipo: int (1)
  
- filter\_low
  - descrição: opção para habilitar filtro para remover regiões de baixa complexidade da seqüência, aumentando a sensibilidade da busca..
  - tipo: int (1)
  
- filter\_mask
  - descrição: opção para habilitar filtro para remover regiões com repetição na seqüência, aumentando a sensibilidade da busca.
  - tipo: int (1)

- filter\_mask\_lower
  - descrição: opção para habilitar os filtros de baixa complexidade e de repetição na seqüência, aumentando a sensibilidade da busca.
  - tipo: int (1)
- mass\_mismatched\_aa
  - descrição: quantidade máxima de erros em aminoácidos.
  - tipo: int (4)

## B.4.7.Errors

### B.4.7.1Descrição Detalhada

Tabela com casos de falha no sistema, utilizada como log de erros retornados durante a submissão e recepção de resultados de identificação de proteínas.

### B.4.7.2Visão da tabela

errors		
<u>ID</u>	int(8)	<pk>
submitted_id	int(8)	<ak,fk>
error	varchar(20)	

**Figura B-8 Visão da tabela “errors”**

### B.4.7.3Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (8), auto-incrementável

- submitted\_id
  - descrição: chave estrangeira, relacionando à tabela “submitted” a fim de identificar a que pesquisa o erro está associado.
  - tipo: int (8)
- error
  - descrição: descrição do erro retornado do sistema.
  - tipo: varchar (20)

## B.4.8.Services

### B.4.8.1Descrição Detalhada

Tabela com os nomes dos programas de identificação de proteína disponíveis para uso pelo sistema. Esta tabela visa facilitar a inclusão de novos serviços, pois é chave estrangeira na tabela “submitted”.

### B.4.8.2Visão da tabela

services		
ID	int(8)	<pk>
service_name	varchar(50)	<ak>

**Figura B-9 Visão da tabela “services”**

### B.4.8.3Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados
  - tipo: int (8), auto-incrementável

- service\_name
  - descrição: descrição do nome do serviço.
  - tipo: varchar (50)

## **B.4.9.Mascot\_taxon**

### **B.4.9.1Descrição Detalhada**

Tabela com a classificação taxonômica utilizada pelo programa Mascot. O Mascot possui uma lista com algumas classificações taxonômicas, que são organizadas de maneira própria do programa, com base na lista de classificações disponibilizada pelo NCBI. Esta tabela é utilizada para relacionar a classificação utilizada pelo Mascot com a classificação do NCBI, de acordo com a taxonomia inserida pelo usuário no formulário “generic”.

### **B.4.9.2Visão da tabela**

mascot_taxon	
mascot_option	varchar(255) <pk>
options_clean	varchar(255)

**Figura B-10 Visão da tabela “mascot\_taxon”**

### **B.4.9.3Atributos**

- Mascot\_option
  - descrição: chave primária da tabela, contém o valor exato da classificação utilizada pelo Mascot (a forma de exibição é composta por vários “...”).
  - tipo: varchar (255)
- option\_clean
  - descrição: classificação taxonômica do Mascot, porem apresentada de forma mais simples, sem os “...”.

- o tipo: varchar (255)

## B.4.10.Statistical\_data

### B.4.10.1Descrição Detalhada

Tabela com os dados estatísticos fornecidos pelo Blast utilizados para calcular e-value. Esta tabela armazena as informações fornecidas pelo programa Blast que são utilizadas para calcular a probabilidade de erro dos resultados fornecidos pelos programas que o usuário elegeu para sua pesquisa.

### B.4.10.2Visão da tabela

statistical_data	
<u>query_identifier</u>	varchar(50) <pk>
K	float
lambda	float
MxN	bigint(40)
status	varchar(15)

**Figura B-11 Visão da tabela “statistical\_data”**

### B.4.10.3Atributos

- Query\_identifier
  - o descrição: chave primária da tabela e chave estrangeira, identificador da pesquisa do usuário, relacionando os dados à tabela submitted .
  - o tipo: varchar (50)
- K
  - o descrição: valor da constante K retornada pelo Blast.
  - o tipo: float
- lambda
  - o descrição: valor da constante lambda retornada pelo Blast.

- tipo: float
- MxN
  - descrição: valor do espaço amostral de sequencias utilizadas para identificação retornada pelo Blast.
  - tipo: bigint(40)
- status
  - descrição: valor da constante K retornada pelo Blast.
  - tipo: varchar(15)

**B.4.11.Result\_aa\_aacomp**  
**B.4.11.1Descrição Detalhada**

Tabela com os resultados retornados pelo programa AACompident, buscados por *e-mail*. Esta tabela armazena os resultados enviados por *e-mail* pelo programa AACompident e recebidos pelo robô do sistema Protein Locator. É realizado um tratamento no arquivo original para se obter separadamente as informações que serão utilizadas no calculo do resultado consolidado da busca.

**B.4.11.2Visão da tabela**

result_aa_aacomp		
ID	int(8)	<pk>
generic_id	int(8)	<ak1, fk>
submitted_id	int(8)	<ak2>
query_identifier	varchar(50)	<ak3>
rank	int(2)	
score	int(4)	
url	varchar(255)	
DB_id	varchar(255)	
pl	decimal(4,2)	
mw	int(8)	
description	varchar(255)	
result_type	varchar(10)	
evaluate	varchar(150)	

**Figura B-12 Visão da tabela “result\_aa\_aacomp”**

### B.4.11.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados.
  - tipo: int (8)
- generic\_id
  - descrição: chave estrangeira, relacionando o resultado ao generic da pesquisa.
  - tipo: int (8)
- submitted\_id
  - descrição: chave estrangeira, relacionando a tabela ao “submitted” que enviou a busca.
  - tipo: int (8)
- query\_identifier
  - descrição: identificação única da pesquisa, contém data de submissão e submitted\_id.
  - tipo: varchar (50)
- rank
  - descrição: posição da proteína na lista de resultados de possíveis proteínas identificadas.
  - tipo: int (2)

- score
  - descrição: escore da proteína após a identificação, indicando a potencialidade do resultado.
  - tipo: int (4)
  
- url
  - descrição: link para a proteína no banco de dados de proteínas (Uniprot).
  - tipo: varchar (255)
  
- DB\_id
  - descrição: ID da proteína no banco de dados de proteínas (Uniprot), utilizado para comparação entre as proteínas identificadas por outros programas.
  - tipo: varchar (255)
  
- pI
  - descrição: ponto isoelétrico da proteína identificada.
  - tipo: decimal (4,2)
  
- mw
  - descrição: massa da proteína identificada.
  - tipo: int (8)
  
- *description*
  - descrição: descrição da proteína identificada.
  - tipo: varchar (255)

- result\_type
  - descrição: tipo de resultado, que pode ser: somente para a taxonomia selecionada ou para todas as taxonomias.
  - tipo: varchar (10)
  
- evalue
  - descrição: e-value da proteína identificada (probabilidade da proteína ter sido tomada ao acaso no banco de dados).
  - tipo: varchar (150)

## B.4.12.Result\_pmf\_mascot

### B.4.12.1Descrição Detalhada

Tabela com os dez melhores resultados retornados pelo programa Mascot. Esta tabela armazena os resultados enviados por *e-mail* pelo programa AACompident e recebidos pelo robô do sistema Protein Locator. É realizado um tratamento no arquivo original para se obter separadamente as informações que serão utilizadas no calculo do resultado consolidado da busca.

### B.4.12.2Visão da tabela

result_pmf_mascot		
<u>ID</u>	int(8)	<pk>
generic_id	int(8)	<ak1,fk>
submitted_id	int(8)	<ak2>
query_identifier	varchar(50)	<ak3>
url	varchar(255)	
DB_id	varchar(50)	
mass	int(8)	
score	int(4)	
evalue	varchar(150)	
matches	int(10)	
description	varchar(255)	

**Figura B-13 Visão da tabela “result\_pmf\_mascot”**

### B.4.12.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados.
  - tipo: int (8)
- generic\_id
  - descrição: chave estrangeira, relacionando o resultado ao generic da pesquisa.
  - tipo: int (8)
- submitted\_id
  - descrição: chave estrangeira, relacionando a tabela ao “submitted” que enviou a busca.
  - tipo: int (8)
- query\_identifier
  - descrição: identificação única da pesquisa, contém data de submissão e submitted\_id.
  - tipo: varchar (50)
- url
  - descrição: link para a proteína no banco de dados de proteínas (Uniprot).
  - tipo: varchar (255)

- DB\_id
  - descrição: ID da proteína no banco de dados de proteínas (Uniprot), utilizado para comparação entre as proteínas identificadas por outros programas.
  - tipo: varchar (255)
- mass
  - descrição: massa da proteína identificada.
  - tipo: int (8)
- score
  - descrição: escore da proteína após a identificação, indicando a potencialidade do resultado.
  - tipo: int (4)
- evaluate
  - descrição: e-value da proteína identificada (probabilidade da proteína ter sido tomada ao acaso no banco de dados).
  - tipo: varchar (150)
- matches
  - descrição: número de matches que a proteína obteve com a amostra submetida.
  - tipo: int (10)
- description
  - descrição: descrição da proteína identificada.

- o tipo: varchar (255)

### B.4.13.temp\_pmf\_mascot

#### B.4.13.1Descrição Detalhada

Tabela temporária com todos os resultados retornados pelo programa Mascot, após o *parsing* do resultado completo deste programa, utilizada para organizar os dez primeiros, de acordo com a ordem inversa de e-value.

#### B.4.13.2Visão da tabela

temp_pmf_mascot		
<u>ID</u>	int(8)	<pk>
generic_id	int(8)	<ak,fk>
submitted_id	int(8)	
query_identifier	varchar(50)	
url	varchar(255)	
DB_id	varchar(50)	
mass	int(8)	
score	int(4)	
evaluate	float	
matches	int(10)	
description	varchar(255)	

Figura B-14 Visão da tabela “temp\_pmf\_mascot”

#### B.4.13.3Atributos

- ID
  - o descrição: chave primária da tabela, utilização interna ao banco de dados.
  - o tipo: int (8)
- generic\_id
  - o descrição: chave estrangeira, relacionando o resultado ao generic da pesquisa.
  - o tipo: int (8)

- submitted\_id
  - descrição: chave estrangeira, relacionando a tabela ao “submitted” que enviou a busca.
  - tipo: int (8)
  
- query\_identifier
  - descrição: identificação única da pesquisa, contém data de submissão e submitted\_id.
  - tipo: varchar (50)
  
- url
  - descrição: link para a proteína no banco de dados de proteínas (Uniprot).
  - tipo: varchar (255)
  
- DB\_id
  - descrição: ID da proteína no banco de dados de proteínas (Uniprot), utilizado para comparação entre as proteínas identificadas por outros programas.
  - tipo: varchar (255)
  
- mass
  - descrição: massa da proteína identificada.
  - tipo: int (8)

- score
  - descrição: escore da proteína após a identificação, indicando a potencialidade do resultado.
  - tipo: int (4)
  
- evaluate
  - descrição: e-value da proteína identificada (probabilidade da proteína ter sido tomada ao acaso no banco de dados).
  - tipo: varchar (150)
  
- matches
  - descrição: número de matches que a proteína obteve com a amostra submetida.
  - tipo: int (10)
  
- description
  - descrição: descrição da proteína identificada.
  - tipo: varchar (255)

#### **B.4.14.Result\_ptn\_fasta**

##### **B.4.14.1Descrição Detalhada**

Tabela com os resultados retornados pelo programa FASTA, após *parsing* do HTML retornado pelo programa.

#### B.4.14.2 Visão da tabela

result_ptn_fasta		
<u>ID</u>	int(8)	<pk>
generic_id	int(8)	<ak1, fk>
submitted_id	int(8)	<ak3>
query_identifier	varchar(50)	<ak2>
url	varchar(255)	
DB_id	varchar(50)	
source	varchar(50)	
length	int(8)	
identity	varchar(10)	
similar	varchar(10)	
overlap	int(8)	
evaluate	varchar(150)	

Figura B-15 Visão da tabela “result\_ptn\_fasta”

#### B.4.14.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados.
  - tipo: int (8)
- generic\_id
  - descrição: chave estrangeira, relacionando o resultado ao generic da pesquisa.
  - tipo: int (8)
- submitted\_id
  - descrição: chave estrangeira, relacionando a tabela ao “submitted” que enviou a busca.
  - tipo: int (8)
- query\_identifier
  - descrição: identificação única da pesquisa, contém data de submissão e submitted\_id.

- tipo: varchar (50)
- url
  - descrição: link para a proteína no banco de dados de proteínas (Uniprot).
  - tipo: varchar (255)
- DB\_id
  - descrição: ID da proteína no banco de dados de proteínas (NCBI), utilizado para comparação entre as proteínas identificadas por outros programas.
  - tipo: varchar (50)
- source
  - descrição: descrição da fonte da proteína.
  - tipo: varchar (50)
- length
  - descrição: tamanho da proteína.
  - tipo: int (8)
- identity
  - descrição: identidade da proteína com a amostra.
  - tipo: varchar (10)
- similar
  - descrição: índice de similaridade da proteína com a amostra.

- tipo: varchar (10)
- overlap
  - descrição: sobreposição de sequencias.
  - tipo: int (8)
- evalue
  - descrição: probabilidade da proteína ter sido escolhida ao acaso no banco de dados.
  - tipo: varchar (150)

#### B.4.15.Result\_ptn\_blast

##### B.4.15.1Descrição Detalhada

Tabela com os resultados retornados pelo programa Blast, após *parsing* do HTML retornado pelo programa.

##### B.4.15.2Visão da tabela

result_ptn_blast		
<u>ID</u>	<u>int(8)</u>	<u>&lt;pk&gt;</u>
generic_id	int(8)	<ak1,fk>
submitted_id	int(8)	<ak2>
query_identifier	varchar(50)	<ak3>
url	varchar(255)	
DB	varchar(50)	
DB_id	varchar(50)	
short	varchar(50)	
description	varchar(255)	
aa	varchar(10)	
score	varchar(20)	
evalue	varchar(150)	
identities	varchar(20)	
positives	varchar(20)	
gaps	varchar(50)	

**Figura B-16 Visão da tabela “result\_ptn\_blast”**

### B.4.15.3 Atributos

- ID
  - descrição: chave primária da tabela, utilização interna ao banco de dados.
  - tipo: int (8)
- generic\_id
  - descrição: chave estrangeira, relacionando o resultado ao generic da pesquisa.
  - tipo: int (8)
- submitted\_id
  - descrição: chave estrangeira, relacionando a tabela ao “submitted” que enviou a busca.
  - tipo: int (8)
- query\_identifier
  - descrição: identificação única da pesquisa, contém data de submissão e submitted\_id.
  - tipo: varchar (50)
- url
  - descrição: link para a proteína no banco de dados de proteínas (Uniprot).
  - tipo: varchar (255)

- DB
  - descrição: banco de dados onde foi encontrada a proteína.
  - tipo: varchar (50)
- short
  - descrição: ID da proteína no banco de dados de proteínas (Uniprot), utilizado para comparação entre as proteínas identificadas por outros programas.
  - tipo: varchar (50)
- *description*
  - descrição: descrição da proteína.
  - tipo: varchar (255)
- aa
  - descrição: tamanho da sequência (em número de aminoácidos).
  - tipo: varchar (10)
- score
  - descrição: escore da proteína após a identificação, indicando a potencialidade do resultado.
  - tipo: varchar (20)
- *evalue*
  - descrição: e-value da proteína identificada (probabilidade da proteína ter sido tomada ao acaso no banco de dados).
  - tipo: varchar (150)

- identities
  - descrição: percentual de identidade entre a amostra e a proteína.
  - tipo: varchar (20)
  
- positives
  - descrição: quantidade de positivos na comparação da seqüência da amostra com a proteína no banco de dados.
  - tipo: varchar (20)
  
- gaps
  - descrição: espaços vazios na identificação da amostra.
  - tipo: varchar (50)

## **C. MANUAL DO ADMINISTRADOR**

### **C.1 VISÃO GERAL DO DOCUMENTO**

Este manual visa apresentar ao administrador do sistema as informações necessárias para possibilitar a instalação e correta utilização deste projeto.

O documento foi dividido nas seguintes seções:

- Requisitos do equipamento
- Requisitos de programas
- Estrutura de diretórios
- Robôs de submissão e recepção de resultados

### **C.2 REQUISITOS DO EQUIPAMENTO**

O equipamento utilizado como servidor do sistema deve ser dimensionado para suportar acessos simultâneos ao servidor *web* e ao servidor de banco de dados. Não é exigida muita memória RAM do equipamento, porém é necessário espaço em disco rígido para armazenar o banco de dados com todas as informações dos usuários e suas pesquisas, incluindo os resultados originais advindos dos programas de identificação de proteínas. O sistema operacional pode ser tanto Linux quanto Microsoft Windows.

### **C.3 REQUISITOS DE PROGRAMAS**

Por se tratar de um sistema cliente-servidor, é indispensável a disponibilização de um programa servidor de página *web* pela rede. No projeto, o servidor *web* utilizado foi o Apache versão 2, por ser amplamente documentado e integrado com os demais programas utilizados pelo Protein Locator.

Para a interpretação dos *scripts*, é indispensável o programa PHP. A versão utilizada foi a 5, com todas as atualizações de segurança disponibilizadas. É necessário adicionar a extensão para acesso ao programa MySQL, servidor de bancos de dados.

Também é necessária a extensão cURL, utilizada pelos robôs de submissão automática de buscas nos programas de identificação de proteínas. Para receber os resultados de identificação por meio de *e-mail*, é necessária utilização de um socket de conexão POP3, disponível com o conjunto de módulos PEAR, chamado de Net\_POP3.

Outro programa utilizado pelo sistema é o sistema gerenciador de bancos de dados MySQL. Este programa foi escolhido pela sua simplicidade de integração com *scripts* em PHP, por sua ampla documentação e pela disponibilidade de uma versão livre para a comunidade. Não foi utilizada nenhuma característica específica do MySQL, sendo portanto possível a utilização de outro sistema gerenciador de bancos de dados, desde que realizadas as devidas alterações nas funções de acesso ao banco (em PHP). Para proteger o banco de dados do projeto, foi criado um usuário com acesso exclusivo a este banco, com poderes limitados a inserir, apagar e atualizar dados, sendo vedada a criação de usuários ou alteração da estrutura das tabelas, além de ser possível apenas conexões realizadas da própria estação servidora (*localhost*) ao banco de dados.

## C.4 ESTRUTURA DE DIRETÓRIOS

O sistema Protein Locator está estruturado em um diretório dentro do “*DocumentRoot*”, diretório raiz dos *sites* definido no arquivo de configuração do Apache, um diretório com os robôs, localizado um nível abaixo do diretório “*DocumentRoot*” descrito. Esta estrutura permite ao Apache acesso irrestrito às páginas do sistema e impede o acesso de qualquer outro usuário, que não o super-administrador, aos robôs de submissão.

O diretório de páginas do sistema contém outros dois subdiretórios, um para as imagens das páginas e outro para os arquivos incluídos em outros *scripts*. Os arquivos contendo as páginas estão todos no diretório raiz do sistema. Dentre os arquivos incluídos, destacam-se o arquivo de configurações para acesso ao banco de dados, *script* para cálculo da probabilidade consolidada dos resultados, *script* para obtenção de informações de taxonomia, que faz a adaptação entre opções do NCBI e do Mascot.

No diretório dos robôs, encontram-se os *scripts* executáveis, em *Shell script*, que acionam os *scripts* em PHP. Estes, também funcionam com include das funções principais, que se conectam nos programas de submissão e no *e-mail* do programa. A nomenclatura padrão para estes robôs é da forma: robo\_”nome da tabela”\_”nome da ferramenta”, por exemplo: robo\_ *fingerprint* \_mascot. Os arquivos com as funções são da forma func\_”nome da função”, por exemplo, func\_mascot.

## C.5 ROBÔS DE SUBMISSÃO E RECEPÇÃO DE RESULTADOS

Os robôs de submissão e recepção de resultados são executados a cada dez minutos pelo programa “*Cron*” do Linux. Em Windows, deveriam ser adicionados às “tarefas agendadas”. Cada robô é responsável pela submissão a um determinado programa, dentre os utilizados pelo sistema. Dessa forma, a cada intervalo de dez minutos cada um dos robôs lê a tabela de dados a submeter e, utilizando a extensão cURL, submetem os dados.

Esta extensão realiza a submissão e permite fazer a recepção dos resultados via *web*. Após a recepção, é armazenada a página com o resultado original e é realizado um *parsing* para obter as informações utilizadas pelo sistema. Para esta análise das página e obtenção dos resultados necessários, é necessário utilização de expressões regulares que contemplem as informações requisitadas em cada página de resultados. Muitas vezes, torna-se necessário a análise de uma página intermediária que contém o endereço para a página final com os resultados. Esta análise também é feita com expressões regulares e PHP.

No caso de programas que só enviam os resultados via *e-mail*, é utilizado o robô de recepção de *e-mails*. Este robô estabelece uma conexão POP3 com o servidor de *e-mail*, recupera as mensagens que foram enviadas pelo programa utilizado para identificação de proteínas, armazena o resultado original no banco de dados e apaga o resultado, depois disso, apaga todas as demais mensagens da caixa de *e-mail*. As mensagens que contêm informações válidas para o sistema possuem um remetente

específico (no caso dos programas atualmente utilizados neste projeto, as mensagens provêm do Expasy). Além do filtro anti-*spam* do próprio *webmail* utilizado, o robô de leitura de *e-mails* faz uma nova filtragem. As mensagens válidas são analisadas também por expressões regulares, que recuperam as informações convenientes, que são armazenadas no banco de dados.

Qualquer erro desses robôs é armazenado na tabela de erros do sistema para controle do log de atividades. Alguns erros mais comuns ocorrem com a perda da conexão com programa (alguma queda do link de internet de algum dos lados da rede) ou algum dado inválido inserido pelo usuário que possa ter passado pelos filtros do sistema (este tipo de erro é pouco provável, porem existe). Outra possibilidade é a mudança do formato da pagina do programa, que pode exibir os resultados de forma diferente. Este tipo de erro deve ser observado pelo administrador e sua correção requer novas expressões regulares para os robôs.