

UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

**Desenvolvimento de um aplicativo para  
iPhone para acesso às informações  
sobre os genes na base da  
Ontocancro.**

TRABALHO DE GRADUAÇÃO

André Lúcio Hahn

Santa Maria, RS, Brasil  
2010

# **Desenvolvimento de um aplicativo para iPhone para acesso às informações sobre os genes na base da Ontocancro.**

**Por**

**André Lúcio Hahn**

Trabalho de Graduação apresentado ao Curso de Ciência da Computação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para a obtenção do grau de **Bacharel em Ciência da Computação**

Orientador: Prof. Giovani Rubert Librelotto

Trabalho de Graduação N. 299  
Santa Maria, RS, Brasil  
2010

**UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

A Comissão Examinadora, abaixo assinada,  
aprova o Trabalho de Graduação

**Desenvolvimento de um aplicativo para iPhone para  
acesso às informações sobre os genes na base da  
Ontocancro.**

elaborado por  
**André Lúcio Hahn**

como requisito parcial para obtenção do grau de  
**Bacharel em Ciência da Computação**

**COMISSÃO EXAMINADORA**

**Prof. Giovani Rubert Librelotto**  
(Presidente/Orientador)

**Prof<sup>a</sup>. Deise De Brum Saccol (UFSM)**

**Prof<sup>a</sup>. Iara Augustin (UFSM)**

Santa Maria, 07 Dezembro de 2010.

## **DEDICATÓRIA**

Dedico este trabalho à minha família, que nunca deixou de acreditar em mim.

## **AGRADECIMENTOS**

Dedico meus agradecimentos a todos que de alguma forma colaboraram com a realização deste trabalho:

- aos meus familiares, pelo apoio em minhas decisões, especialmente os meus pais;
- ao Profº. Dr. Giovani Rubert Librelotto, pela orientação neste trabalho;
- aos colegas de graduação da UFSM, que ajudaram de alguma forma;
- aos demais professores do curso de Ciência da Computação, que colaboraram na minha formação acadêmica.

*“Não vivemos uma era de mudanças.....*

*....vivemos uma mudança de era.”*

Chris Anderson

# RESUMO

Trabalho de Graduação  
Curso de Ciência da Computação  
Universidade Federal de Santa Maria

## **Desenvolvimento de um aplicativo para iPhone para acesso às informações sobre os genes na base da Ontocancro.**

Autor: André Lúcio Hahn

Orientador: Prof. Giovani Rubert Librelotto

Local e data da defesa: Santa Maria, 07 de dezembro de 2010

Com a crescente popularização dos dispositivos móveis e crescente acesso à rede por meio destes dispositivos, tem-se notado o aumento da demanda para a disponibilização de novas funcionalidades e serviços, além dos tradicionais, por meio de dispositivos móveis.

Neste contexto, este trabalho se justifica pelo interesse da Ontocancro disponibilizar o acesso ao seu banco de dados ontológico de genes por tais dispositivos.

Assim o objetivo deste trabalho consiste em criar um aplicativo para acesso às informações sobre os genes na base Ontocancro.

O dispositivo móvel escolhido para ser utilizado na implementação foi o iPhone, telefone celular da Apple. Aplicativos para esta plataforma são criados utilizando-se a linguagem de programação Objective-C em conjunto com as ferramentas disponibilizadas no SDK do próprio iPhone.

**Palavras chaves:** Aplicações móveis, iPhone, Ontocancro, Objective c, Ontologia.

# **ABSTRACT**

Trabalho de Graduação  
Curso de Ciência da Computação  
Universidade Federal de Santa Maria

## **DEVELOPMENT OF AN IPHONE APP TO ACCESS INFORMATION ABOUT THE GENES ON THE BASIS OF ONTOCANCRO.**

Author: André Lúcio Hahn  
Advisor: Prof. Giovani Rubert Librelotto

With the growing popularity of mobile devices and increasing access to the network through these devices has been seeing increased demand for the release of new features and services beyond the traditional, through mobile devices.

In this context, this work is justified by the interest of Ontocancro provide access to its database of ontological genes for such devices.

So our objective is to create an application for access to information about the genes on the basis Ontocancro.

The mobile device chosen to store the applicative developed is the Apple's iPhone. Applications for this platform are created using the Objective-C programming language and the iPhone SDK tools.

**Keywords:** Mobile applications, iPhone, Ontocancro, Objective c, Ontology.



## LISTA DE FIGURAS

Figura 2.1: Página inicial do site da Ontocancro .....	23
Figura 2.2: Arquitetura do sistema para o processamento da Ontocancro.....	23
Figura 2.3: Relação entre Model-View-Controller. ....	18
Figura 3.1: Arquitetura do sistema para processamento da Ontocancro do APP..	26
Figura 3.2: Folha de estilo XSL Parser.....	26
Figura 3.4: Lista dos genes na página da Ontocancro .....	28
Figura 3.5: Informações detalhadas sobre o gene.....	29
Figura 3.6: Estrutura e sequência de acesso as informações no aplicativo.....	29
Figura 3.7 Classes que herdam de UITableViewController.....	31
Figura 3.8: Classes que herdam de UIViewController.....	31
Figura 3.9: Classe gene que é filha de NSObject e a classe app da aplicação .....	32
Figura 3.10: Tela inicial do aplicativo.....	33
Figura 3.11: Lista de genes e com a exibição do teclado auxiliar para buscar e selecionar um gene.....	33
Figura 3.12: Informações detalhadas do gene selecionado .....	34
Figura 3.13: <i>Pathways</i> da ontologia Ontocancro.....	34
Figura 3.14: Listagem dos genes relacionados ao pathway.....	35
Figura 3.15: Informações do gene selecionado relacionados ao pathway.....	35

## LISTA DE ABREVIATURAS E SIGLAS

- DNA** – Ácido Desoxirribonucléico
- GMM** - Genome Maintenance Mechanisms
- J2ME** - Java 2 Micro Edition
- MVC** - Model-View-Controller
- OWL** – Web Ontology Language
- RNA** – Ácido Ribonucléico
- SDK** - Software Development Kit
- UIKit** - User Interface Kit
- XML** – eXtensible Markup Language
- XSL** - (*eXtensible Stylesheet Language*)
- XTM** - XML *Topic Maps*

# SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	<b>12</b>
1.1 Objetivos .....	12
1.2 Estrutura do texto .....	12
<b>2. REVISÃO BIBLIOGRÁFICA</b> .....	<b>14</b>
2.1 Conceitos e definições .....	14
2.1.1 Ontologia e Topic Map .....	14
2.1.2 Gene .....	15
2.2 Projeto Ontocancro .....	15
2.2.1 Descrição da arquitetura original .....	17
2.3 Programação para iPhone .....	19
2.3.1 Linguagem objective-C.....	20
2.3.2 iPhone SDK.....	21
2.3.3 Padrão de arquitetura de software Model-View-Control .....	22
<b>3. DESENVOLVIMENTO DE UM APLICATIVO PARA ACESSO MÓVEL A BASE DE DADOS ONTOCANCRO, UTILIZANDO O IPHONE</b> .....	<b>24</b>
3.1 Objetivos e requisitos do aplicativo .....	24
3.2 Arquitetura da ferramenta .....	25
3.2.1 Descrição da arquitetura para processamento da Ontocancro pelo aplicativo .....	25
3.3 Projeto e modelagem dos dados .....	27
3.4 Descrição das interfaces .....	32
3.5 Teste e análise dos dados .....	36
<b>4. CONCLUSÃO</b> .....	<b>38</b>
<b>5. REFERÊNCIAS</b> .....	<b>39</b>
<b>6 ANEXO A – GERADOR XSL</b> .....	<b>41</b>

# 1 INTRODUÇÃO

Nos últimos anos, a popularidade dos dispositivos móveis tem crescido a ponto de tornarem-se acessórios comuns e totalmente integrados ao dia-a-dia do consumidor. Suas funcionalidades também têm crescido exponencialmente. Hoje em dia, é comum para a maioria dos dispositivos móveis incluírem funções de telefone, agenda de compromissos, alarme, câmera entre outros. Além disso, o crescente acesso a *Internet* por meio destes dispositivos criou uma demanda para oferecer serviços e funcionalidades que, até o momento, somente são oferecidos por terminais estáticos.

Neste contexto, o projeto Ontocancro (Ontocancro, 2010), tem por objetivo disponibilizar aos seus usuários o acesso ao seu banco de dados de genes por meio de dispositivos móveis.

Dentre os dispositivos móveis existentes atualmente, um dos mais atrativos tanto a desenvolvedores quanto a usuários é o telefone celular multifuncional da Apple, o iPhone. Devido a suas características e funcionalidades inovadoras ele será utilizado como objeto alvo do aplicativo implementado neste projeto.

## 1.1 Objetivos

O objetivo geral deste trabalho é a construção de uma ferramenta que possibilite ao portador de um iPhone o acesso às informações sobre os genes na base de dados Ontocancro. Essa ferramenta visa auxiliar o projeto Ontocancro ao providenciar um meio de acesso móvel a sua base de dados sem a necessidade do usuário do sistema procurar um terminal estático para conseguir acesso aos dados.

## 1.2 Estrutura do Texto

Este trabalho está estruturado da seguinte forma: no Capítulo 2 é realizada a revisão bibliográfica sobre conceitos e tecnologias úteis para o desenvolvimento do trabalho. O Capítulo 3 detalha a criação do aplicativo, apresentando as ferramentas

usadas para seu desenvolvimento e as funcionalidades disponíveis. Por último, o capítulo 4 conclui o trabalho.

## **2. REVISÃO BIBLIOGRÁFICA**

### **2.1 Conceitos e definições**

Nesta seção serão apresentados os conceitos importantes para uma melhor compreensão de seções posteriores.

#### **2.1.1 Ontologia e Topic Maps**

Uma ontologia é uma especificação explícita de uma conceitualização [GRUBER, 1993]. Enquanto vocabulários controlados somente restringem as palavras que serão utilizadas em um determinado domínio, as ontologias estendem esta característica simples dos vocabulários controlados e permitem uma especificação formal de termos e seus relacionamentos, tornando possível compartilhar e reutilizar o conhecimento, suportando a interoperabilidade entre os sistemas e também permitem inferências sobre o conhecimento representado.

Topic Maps, de acordo com a ISO (International Organization for Standardization) e a IEC (International Electrotechnical Commission), é uma norma internacional unificada para descrever estruturas de conhecimento e formalizar a sua associação com recursos de informações [BIEZUNSKY; BRYAN; NEWCOMB, 1999].

Nestes termos, referencia-se Topic Maps como uma tecnologia habilitada para a representação e manipulação do conhecimento, proporcionando também uma poderosa forma de navegação sobre recursos de informação. Seu conceito pode ser definido como uma descrição de um ponto de vista sobre uma coleção de recursos, organizado formalmente por tópicos e pela ligação de algumas partes relevantes do conjunto de informação aos tópicos apropriados.

## 2.1.2 Gene

O gene é um código distinto, um seguimento de um cromossomo. Segundo o projeto Genoma [MAGATÃO, 2008] um gene pode ser definido como uma unidade fundamental, física e funcional da hereditariedade, ou seja, uma informação capaz de produzir uma determinada proteína ou atuar no controle de uma característica, como por exemplo, a cor dos olhos.

## 2.2 Projeto Ontocancro

A Ontocancro consiste em uma base de dados que reúne informações de genes e vias envolvidas no processo carcinogênico, onde foram filtrados e catalogados aproximadamente 1428 genes distribuídos em 130 vias (Ontocancro, 2010). Todos estes dados foram extraídos dos principais bancos de dados públicos de genes: *NCI-Nature*, *BioCarta*, *KEGG*, *Reactome*, *Prosite*, *GO* e *STRING*, além dos demais citados anteriormente.

Devido à falta de consenso na definição dos conjuntos de genes e das vias de cada base de dados, o projeto apresenta um padrão próprio para as vias Ontocancro, que estão relacionadas da seguinte forma (Ontocancro,2010):

- Apoptose – 491 genes
- Reparação por Excisão de Base (BER) – 44 genes
- Ciclo Celular (CC) – 286 genes
- Estabilidade Cromossômica (CS) – 76 genes
- Apoptose expandida – 955 genes
- Recombinação homologa (HR) – 34 genes
- Reparação por mau pareamento (MMR) – 28 genes
- Ligação de extremidades não coesivas – 14 genes
- Reparação por Excisão de Nucleotídeos (NER) – 51 genes

A disfunção destas vias pode provocar vários tipos de anomalias celulares, uma delas é o câncer. As vias se dividem de acordo com a sua função. Por exemplo, na via de reparação por excisão de base (BER) encontra-se um conjunto de genes envolvidos diretamente no reparo das bases do DNA; assim como as outras vias de

reparação, tais como, reparação por excisão de nucleotídeos (NER), reparação por mau pareamento (MMR) e recombinação homologa (HR), entre outras, também auxiliam na manutenção e no reparo do DNA.

Em outros casos, aparecem as vias de ciclo celular (CC), responsáveis pelo processo de duplicação do DNA. Outra via importante é a via de estabilidade cromossômica (CS), que coordena todas as proteínas responsáveis pela estabilidade dos cromossomos. Se algumas destas vias não são capazes de corrigir ou atuar de forma adequada, entra em função a via de apoptose, que tem como função destruir a célula ou o DNA. É ela a responsável pelo suicídio celular, também conhecido por morte celular programada.

O acesso a base de dados da Ontocancro é feita a partir do endereço eletrônico <http://www.ontocancro.org> (Figura 2.1), frequentemente utilizado pelos pesquisadores que estudam o comportamento dos genes relacionados ao câncer.

**Ontocancro**

Webmail Resources and Pathways Genes Publications Tools Downloads Contact

Search

**Ontocancro**

 Cancer cells have large and small abnormalities in their genetic material: additional or missing chromosomes, mutated genes and other types of genetic alterations due to dysfunctional pathways including DNA repair and chromosome stability. The loss of Genome Maintenance Mechanisms (GMM) is one of the most important aspects of human carcinogenesis being related to 90% of the deaths from cancer and several genetic syndromes are known to be associated to GMM dysfunction. Ontocancro is a database that compiles gene pathways involved in GMM that comprises several pathways and subpathways of cell cycle, DNA repair mechanisms, apoptosis and chromosome stability whose information is scattered in the scientific literature and in many databases such as NCI-Nature, BioCarta, KEGG, Reactome, Prosite, Gene Ontology and String.

Due to the large heterogeneity of the data available from these databases and the absence of a specialized tool in GMM is difficult to build models involving these pathways. Ontocancro aims to facilitate the GMM network and functional modeling for the integration of genomic, transcriptomic and interatomic data.

Ontocancro project is a collaboration among Universidade Federal de Santa Maria, Centro Universitário Franciscano and Universidade Federal do Rio Grande do Sul from Brazil. Ontocancro is an open source and free on-line resource.

[Contact](#)

**Quick Links**

- [Ontocancro pathways](#)
- [NCI-Nature pathways](#)
- [KEGG pathways](#)
- [BioCarta pathways](#)
- [Gene Ontology pathways](#)
- [Reactome pathways](#)
- [Prosite protein domains](#)

**Figura2.1:** Página inicial do site da Ontocancro



O site também possibilita ao usuário baixar os arquivos textos com todas as informações das vias, que pode ser lido e processado em planilhas eletrônicas ou editores de texto. Isto facilita a extração das informações por outros aplicativos, tornando o acesso aos dados mais dinâmico. Pode-se, ainda, selecionar um gene pertencente a uma determinada via para obter informações detalhadas sobre o mesmo.

A Ontocancro reúne diversas informações sobre cada gene catalogado. Estas informações estão descritas também no arquivo texto da via, que pode ser visualizado e mais facilmente descrito quando aberto em uma planilha eletrônica.

### **2.2.1 Descrição da arquitetura original**

A Ontocancro é composta de informações de bancos de dados biológicos. Como se percebe na Figura 3.1(Ontocancro, 2010), a partir dos acordos com todos os mantenedores dos bancos de dados biológicos utilizados como fonte da Ontocancro, foram obtidos os arquivos texto e XML referentes a cada banco. Os arquivos de texto são posteriormente tratados, para que seus dados estejam em formato XML e, desta forma, possam ser lançados no banco de dados XML da ontologia.

O banco XML da ontologia está armazenado em um sistema gerenciador de banco de dados XML nativo *eXist*. Neste banco, os arquivos XML obtidos de cada fonte são mantidos em sua forma original.

Entretanto, a partir dos bancos de dados biológicos consultados se obtém um arquivo para cada uma de suas vias relacionadas com interatoma e transcriptoma. Assim, o banco de dados *eXist* é, neste momento, composto por mais de cento e trinta arquivos XML. Dentro deste montante, destacam-se trinta e dois arquivos oriundos da *Biocarta* e sessenta e cinco arquivos obtidos do *GeneOntology*; estes são os bancos biológicos que mais contribuem com vias metabólicas à Ontocancro.

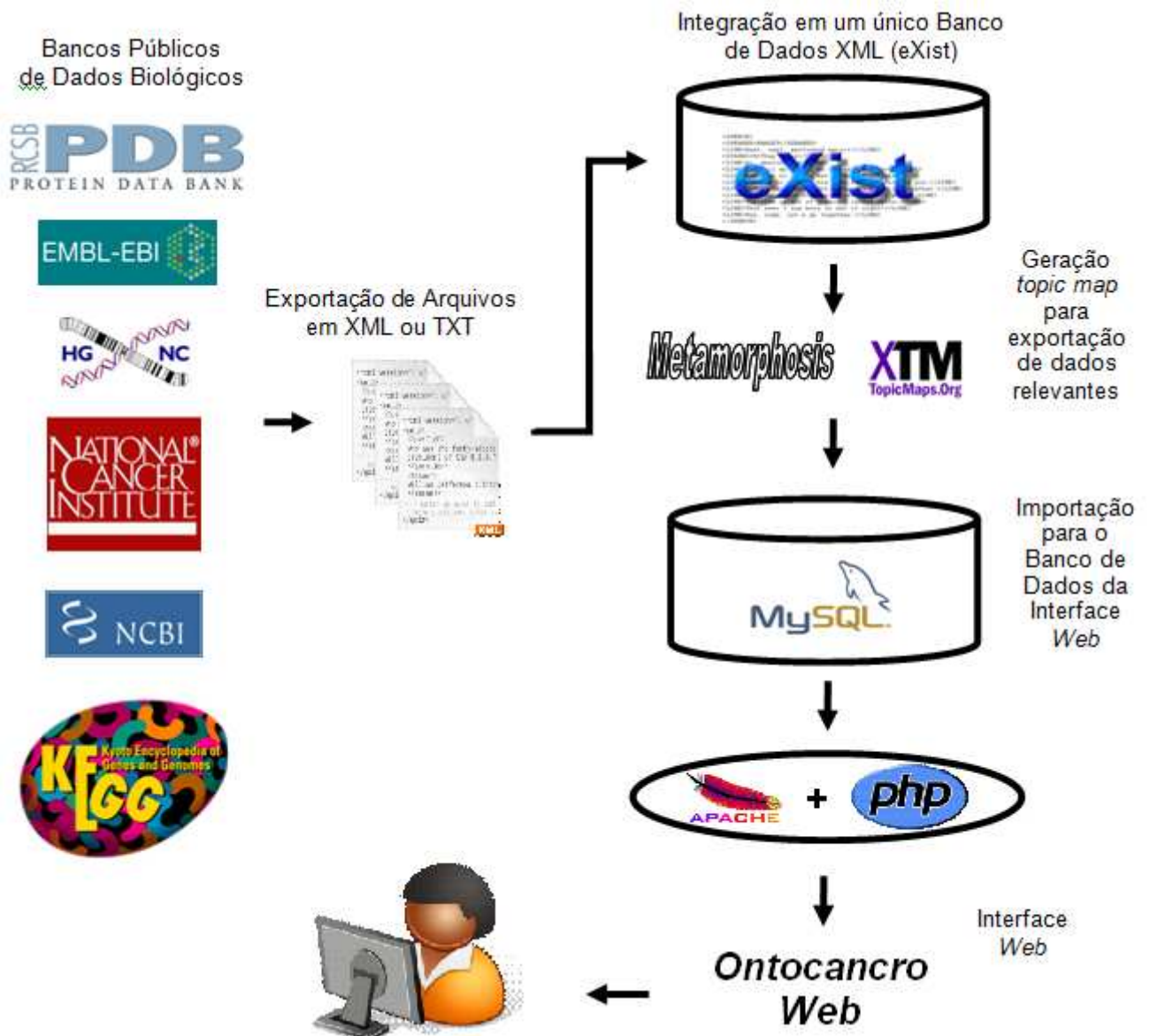


Figura 3.1: Arquitetura do sistema para o processamento da Ontocancro.(Ontocancro)

Os arquivos XML obtidos a partir do mesmo banco biológico são estruturados de acordo com o mesmo esquema XML. Entretanto, os arquivos de bancos distintos possuem esquemas diferentes entre si. Para possibilitar a integração destes documentos com esquemas distintos, usa-se o *Metamorphosis*. Esta ferramenta cria um *topic map* para cada banco de dado biológico, composto pelos dados que são importantes para a Ontocancro.

Para possibilitar uma integração coerente entre os genes e vias metabólicas, encontrados nos diversos bancos, são utilizados principalmente dados como:

- código *EntrezGene* de cada gene, gerenciado pelo *NCBI*;
- código e o nome de acordo com o banco *HGNC*;

➤ identificador no banco *NCI*.

A partir desta integração, o *Metamorphosis* gera um único *topic map* que contém todos os dados oriundos dos diversos bancos de dados biológicos consultados. Este *topic map* único contém a ontologia chamada Ontocancro.

A necessidade de ter-se criado um banco de dados relacional *MySQL* paralelo ao *eXist* deve-se ao fato de que a atualização dos bancos de dados biológicos se dá, geralmente, a partir de documentos XML. Desta forma, mantém-se o banco de dados XML para permitir uma atualização permanente da Ontocancro, enquanto que o banco de dados relacional é utilizado para a geração das páginas Web e para o seu motor de busca, em sua versão mais atualizada.

## 2.3 Programação para iPhone

Antes mesmo de ser lançado, o dispositivo móvel nomeado iPhone (APPLE IPHONE, 2010) já era considerado revolucionário e, após o seu lançamento, as expectativas vieram a se confirmar, ameaçando a aparente hegemonia da programação com J2ME. Mesclando os telefones celulares comuns com tocadores de MP3, além de oferecer suporte à rede sem fio, bússola e GPS integrado, acelerômetros embutidos, telas multi toque, entre outras características, o dispositivo possui atualmente, em sua melhor configuração, um processador de 600MHz e 256MB de memória RAM e até 32GB de armazenamento, algo considerável frente às configurações da maioria dos telefones celulares existentes no mercado.

A criação de aplicações para este dispositivo mostra-se um desafio para inciantes, pois além da expectativa na possibilidade de explorar todos os recursos disponíveis no mesmo e criar aplicações cada vez mais complexas, não há uma cultura muito difundida de utilização de aparelhos da Apple no Brasil, tornando assim escassos os materiais necessários para tal, pois o Kit de Desenvolvimento de Software (SDK) para o iPhone funciona somente em sistemas operacionais Mac OS X (APPLE, 2010).

### 2.3.1 Linguagem Objective-C

A programação de aplicativos para o iPhone é feita utilizando-se a linguagem de programação Objective-C (APPLE,2010). Atualmente, na versão 2.0 (adicionando o recurso de coletor de lixo), esta linguagem é considerada uma fina camada acima da linguagem ANSI C (ANSI C, 2009), ou seja, incorpora ao C os conceitos de orientação a objetos e reflexão além de troca de mensagens.

O modelo de programação orientada a objetos desta linguagem é baseado na troca de mensagens entre as instâncias de objetos. Em Objective-C não são chamados métodos e sim realizadas mensagens. A diferença entre estes dois conceitos está em como o código referido pelo nome do método ou mensagem é executado. No primeiro caso, o nome do método é ligado a uma sessão do código da classe de destino pelo compilador. Já com o conceito de mensagens, o nome da mesma é apenas um nome, e o alvo deste é resolvido somente em tempo de execução, fazendo com que o objeto receptor da mensagem se preocupe em como tratá-la. Como consequência, o sistema de troca de mensagens não possui verificação de tipos e caso o receptor não possua alguma maneira para tratar a mensagem recebida, ele simplesmente a ignora, retornando um ponteiro nulo.

Outra característica da linguagem é que ela é inteiramente baseada em objetos. Todos os tipos primitivos como *Strings* e inteiros, até estruturas mais elaboradas como vetores, são representados como objetos, *NSString*, *NSNumber* e *NSArray*, respectivamente nos casos citados. Todos objetos herdam suas características do tipo padrão da linguagem: a classe *NSObject*.

São adicionados os tipos:

- id: ponteiro para um objeto qualquer. Permite a criação de objetos dinamicamente tipados.
- nil: ponteiro para objetos nulos.
- SEL: *selector*, pode ser considerado similar a um ponteiro para função.
- YES e NO: os tipos booleanos, representando verdadeiro e falso.

Suas classes são referidas como interfaces e estas são declaradas em um arquivo de extensão ".h" abaixo do *token @interface*. Uma classe em Objective-C possui métodos de classe (iniciados com um sinal de "+" durante sua declaração na interface), equivalentes a métodos estáticos em C++, e métodos de instância

(iniciados com o sinal de "-"). A função da interface é informar o protótipo do que será criado. Os métodos são verdadeiramente implementados em arquivos com a extensão ".m" logo após o *token @implementation*. Uma das vantagens presentes em Objective-C é o conceito de *@synthesize* que, ao ser utilizado durante a implementação, localizado antes a uma variável da classe, automaticamente cria os métodos de requisição e alteração de valor desta variável.

### 2.3.2 iPhone SDK

O SDK do iPhone disponibiliza ferramentas e recursos necessários para a criação de aplicações para o iPhone que aparecem como ícones na tela principal do usuário. Essas aplicações executam de forma autônoma e possuem acesso a todas as funcionalidades que tornam o iPhone interessante, como os acelerômetros, serviço de localização e interface multitoque. Aplicações também podem salvar dados no sistema de arquivos local e se comunicar com outras aplicações instaladas no dispositivo.

Para a criação de aplicativos, deve-se utilizar a API Cocoa Touch (APPLE,2010) que consiste de uma camada de abstração do sistema operacional presente em um iPhone. Esta é dividida em duas partes *UIKit* e *Foundation*.

Todo aplicativo desenvolvido para esta arquitetura é construído utilizando-se o *UIKit Framework* e, portanto, possui a mesma arquitetura núcleo. Este *framework* fornece objetos necessários para se executar o aplicativo e coordenar as eventuais interrupções das entradas do usuário e a exibição de conteúdo na tela.

Um aplicativo em execução recebe interrupções continuamente do sistema e precisa responder a estas interrupções. Receber estas interrupções é papel do objeto *UIApplication*, porém respondê-las é de responsabilidade do código a ser desenvolvido. Para uma melhor compreensão de onde se deve responder a certas interrupções, é necessário entender um pouco sobre o ciclo de vida de um aplicativo para iPhone.

Na inicialização e término, o *framework UIKit* envia mensagens específicas para a o objeto *Delegation* do aplicativo, responsável por gerenciar o funcionamento geral do mesmo, por informá-lo sobre qualquer acontecimento ocorrido durante o

ciclo de vida do aplicativo. Durante o ciclo de eventos, o *framework UIKit* despacha as interrupções para o manipulador das mesmas, presente no aplicativo criado.

As ferramentas que compõem o SDK do iPhone são o Xcode, onde todos os arquivos necessários à criação de um aplicativo são manipulados, e também onde o código fonte é escrito. A ferramenta *Interface Builder* é responsável pela manipulação dos arquivos com extensão ".nib" onde são definidas as interfaces visuais a serem manipuladas pelo usuário. E, por fim, ao término da criação de um novo programa é necessário testá-lo, e para isso a última aplicação presente no SDK é o *iPhone Simulator*, aplicação que emula um iPhone dentro do computador do desenvolvedor.

### **2.3.3 Padrão de arquitetura de software Model-View-Control**

O desenvolvimento de aplicativos para iPhone baseia-se no padrão de arquitetura de *software Model-View-Controller (MVC)*, que especifica que, com o aumento da complexidade dos programas criados, torna-se fundamental a separação entre os dados (*Model*) e o leiaute (*View*). A adoção desse padrão traz ao programador uma maior segurança, seja na manipulação dos dados, garantindo que isto não afetará o leiaute, seja na eventual reorganização do leiaute, garantindo que não resulte em um efeito colateral aos dados.

Para realizar a ligação entre dados e leiaute é criada a figura do controlador (*Controller*), que processa e responde a ações do usuário, através das interfaces e também pode realizar modificações diretas nos dados. Esta relação é exibida na Figura 2.4 onde é apresentada a figura do controlador, baseada em um objeto modelo, e somente ele troca informações com o que é exibido na tela do aplicativo.



**Figura 2.4:** Relação entre Model-View-Controller. (Apple iPhone App Programming Guide)

No caso específico do iPhone, a utilização de MVC representa, na grande maioria dos casos, que cada tela de exibição existente possuirá uma classe controladora responsável pela lógica da mesma, bem como por indicar todos os dados a serem exibidos.

Devido à variada gama de elementos de interface já inclusos no SDK, muitos destes também trazem consigo um controlador "cru", ou seja, disponibiliza-se o esqueleto de uma classe controladora para o tipo de elemento escolhido. Tendo acesso a essa classe, necessita-se acrescentar o código para tratamento de interrupções, inserção de dados, entre outras funcionalidades, nos seus respectivos locais.

Dentre os elementos com classes controladoras disponíveis convêm citar:

- *View Controller*: controlador responsável por gerenciar uma tela genérica, com qualquer tipo de elemento inserido;
- *Navigation Controller*: controlador que gerencia a navegação hierárquica entre diversas telas;
- *Tab Bar Controller*: controlador que gerencia um conjunto de telas que representam itens/ícones numa barra guia, normalmente, localizada na parte inferior do aplicativo;
- *Table View Controller*: controlador que administra uma tela contendo uma exibição em forma de tabela;
- *Image Picker Controller*: controlador que administra uma tela para escolha e navegação entre imagens.

### 3. O DESENVOLVIMENTO DE UM APLICATIVO PARA ACESSO MÓVEL A BASE DE DADOS ONTOCANCRO, UTILIZANDO O IPHONE

Neste capítulo é detalhado o processo de desenvolvimento do aplicativo para acesso a base de dados de genes cancerígenos Ontocancro, utilizando o dispositivo móvel iPhone (APPLE, 2010). O aplicativo desenvolvido provê uma ferramenta para que o usuário, tenha acesso às informações relacionadas a cada gene pertencente a base de dados Ontocancro, bem como exibir os *pathways* e as relações existentes entre eles. Primeiramente, é descrito o objetivo do aplicativo desenvolvido além de seus requisitos. A seguir, é detalhado o projeto do mesmo e, para concluir o capítulo, é descrita a implementação em si, além de uma exibição das interfaces finais obtidas.

#### 3.1 Objetivos e requisitos do aplicativo

Todos os aplicativos desenvolvidos para terem êxito precisam cumprir alguns objetivos e requisitos previamente definidos. O objetivo geral é possibilitar o acesso móvel, através do iPhone, a base de dados da ontologia Ontocancro. A escolha do dispositivo móvel da Apple se justifica devido as suas características e funcionalidades inovadoras.

Para cumprir com este objetivo geral, alguns objetivos mais específicos consistem de:

1. Buscar bancos, genes e *pathways*;
2. Visualizar genes e *pathways*;
3. Visualizar os genes que interagem entre si;
4. Visualizar o grafo de *pathways*;
5. Efetuar downloads da ontologia (em OWL e XTM);
6. Efetuar downloads das listas de genes dos *pathways*;



Tendo estes objetivos definidos, os seguintes requisitos precisam ser observados:

- O usuário deverá ter a possibilidade de acessar o sistema de qualquer lugar, através de seu dispositivo móvel;
- A interface de visualização deve ser adequada ao dispositivo da Apple para informações referentes à ontologia Ontocancro.

Após a definição do objetivo principal e dos objetivos mais específicos, bem como dos requisitos a serem implementados, partiu-se para a criação do projeto do aplicativo.

## 3.2 Arquitetura da Ferramenta

Originalmente, a Ontocancro está baseada em uma ontologia que é tida como a fonte dos dados que estão presentes no site [www.ontocancro.org](http://www.ontocancro.org). A arquitetura original foi descrita na seção 2.1.1. A seguir será descrito arquitetura para o processamento da Ontocancro pelo aplicativo deste trabalho.

### 3.2.1 Descrição da arquitetura para o processamento da Ontocancro pelo aplicativo.

Para permitir o acesso às informações sobre os genes da Ontocancro foi necessário transformar o arquivo XTM para um formato de arquivo XML para Iphone, como mostrado na Figura 3.1. A folha de estilos XSL (*eXtensible Stylesheet Language*) tem a função de receber a ontologia Ontocancro em um formato XML *Topic Maps* (XTM) e gerar um arquivo XML com as informações que a aplicação para iPhone necessita (Figura 3.2).

Esta folha de estilos XSL desenvolvida é composta por 10 *templates*, onde cada um trata um dos itens da ontologia que interessa para a aplicação. Por exemplo, há um *template* específico para todos os genes, enquanto outros *templates* para processar cada *pathway* originário de um banco de dados biológico específico. Por fim, outro *template* processa todas as associações existentes entre genes e

pathways da ontologia. O template XSL desenvolvido se encontra no Anexo A (Librelotto, 2010).

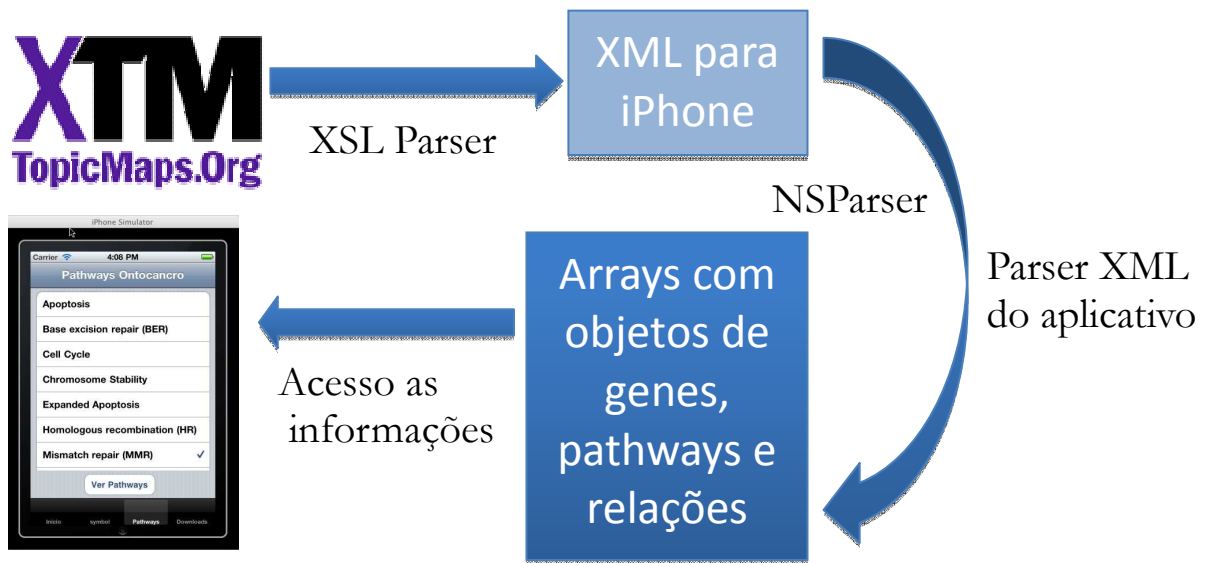


Figura 3.1 – Arquitetura do sistema para processamento da Ontocancro do APP.

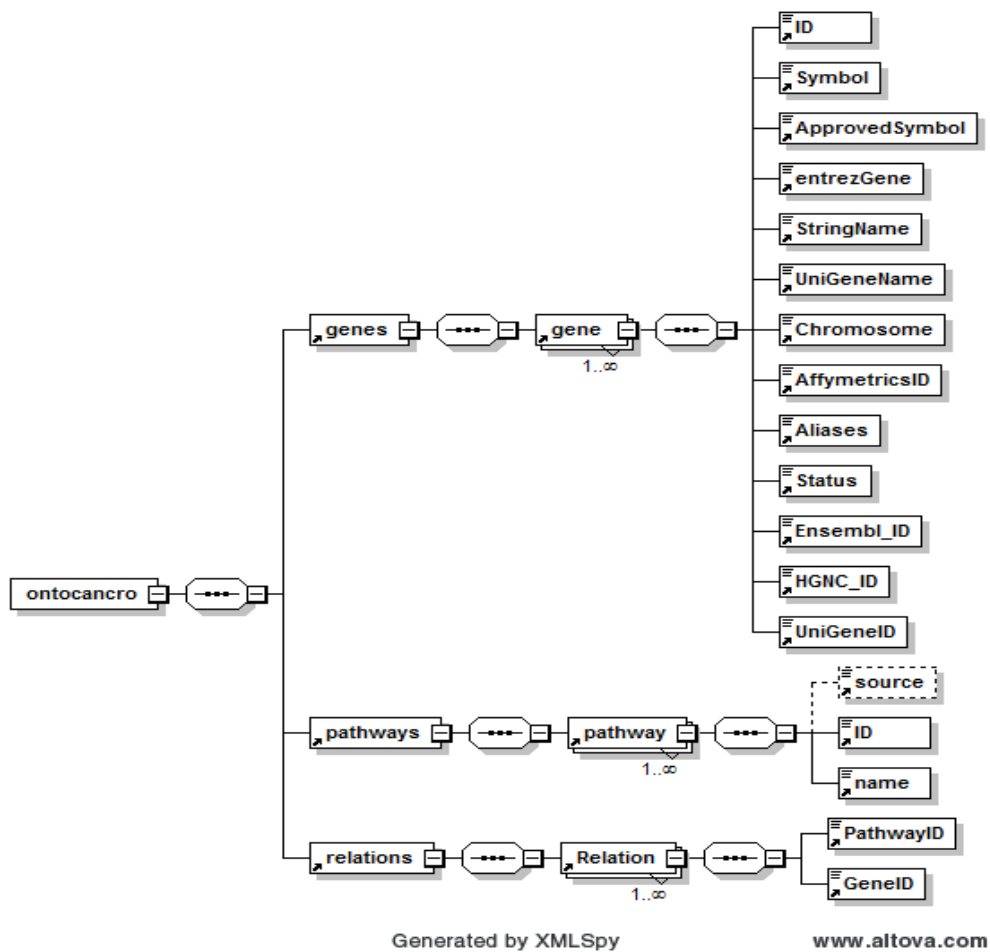


Figura 3.2 – Folha de estilo XSL Parser

Devido ao fato de que a aplicação foi gerada a partir do conhecimento expresso na ontologia Ontocancro, os dados no arquivo XML não necessitam estar representados na forma ontológica original. Sendo assim, o arquivo XML gerado possui um tamanho significativamente inferior, o que facilita o processamento por um dispositivo móvel. A ontologia Ontocancro ocupa 12 MB em um arquivo XTM, enquanto que o arquivo gerado para a aplicação aqui desenvolvida ocupa apenas 1.77 MB.

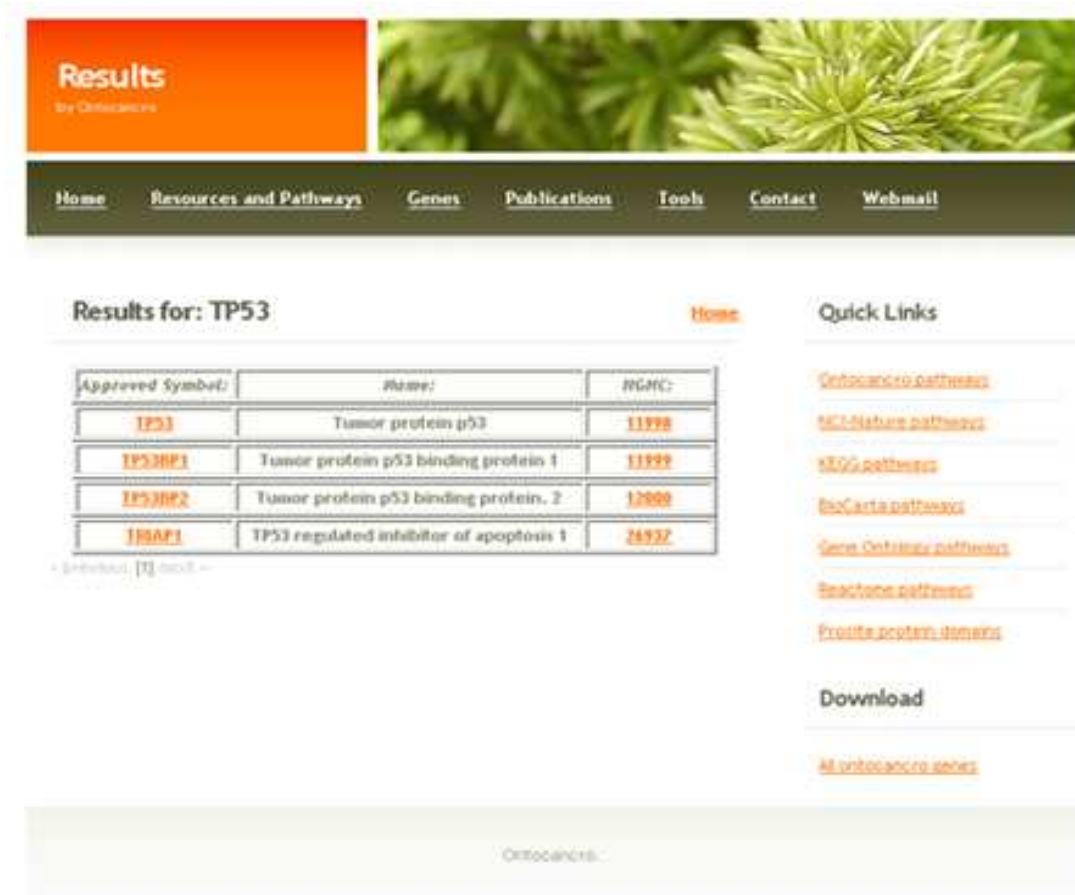
A partir do arquivo XML gerado, com o auxílio da biblioteca *NSParser*, disponibilizada pela próprio SDK para iPhone, é realizada a leitura de arquivo XML para a utilização pelo aplicativo desenvolvido. A biblioteca *NSParser* disponibiliza os métodos necessário para a leitura do arquivo XML, ou seja, somente é necessário repassar ao método ( *didStartElement* ) a tag de início e para outro método a tag de finalização ( *didEndElement* ), e por fim um terceiro método ( *foundCharacters* ) retorna a informação contida no meio das tag de começo e finalização.

O arquivo XML gerado para iPhone contém as informações sobre todos os genes da Ontocancro, os *pathways* e as suas relações. Por isso, foram criadas três classes distintas para armazenar estas informações e o conjunto destas informações e armazenado em seu respectivo vetor. A aplicação posteriormente utiliza estes *arrays* de genes, *pathways* e relações para mostrar as informações sobre os genes da ontologia Ontocancro.

### 3.3 Projeto e modelagem dos dados

Para o projeto deste aplicativo, convêm citar que já é possível acessar os dados da ontologia Ontocancro via *browser*. A diferença entre o acesso por um navegador de *internet* e o aplicativo desenvolvido neste trabalho é a necessidade do primeiro estar constantemente conectado a rede, enquanto o segundo por possuir um repositório local (arquivo no formato XML) permite acessar as informações sem o acesso a rede, e, além disso, permitir a mobilidade do usuário com o dispositivo móvel.

A Figura 3.4 exibe interface de visualização da lista dos genes da ontologia Ontocancro acessado via *browser*, e a Figura 3.5 mostra informações detalhadas de um determinado gene selecionado.



Results for: TP53

Approved Symbol	Name	NCIC
<a href="#">TP53</a>	Tumor protein p53	<a href="#">11998</a>
<a href="#">TP53BP1</a>	Tumor protein p53 binding protein 1	<a href="#">11999</a>
<a href="#">TP53BP2</a>	Tumor protein p53 binding protein, 2	<a href="#">12000</a>
<a href="#">TBAI1</a>	TP53 regulated inhibitor of apoptosis 1	<a href="#">24932</a>

Quick Links

- [Ontocancro pathways](#)
- [NCI Nature pathways](#)
- [KEGG pathways](#)
- [BioCarta pathways](#)
- [Gene Ontology pathways](#)
- [Reactome pathways](#)
- [Protein-protein domains](#)

Download

[All ontocancro genes](#)

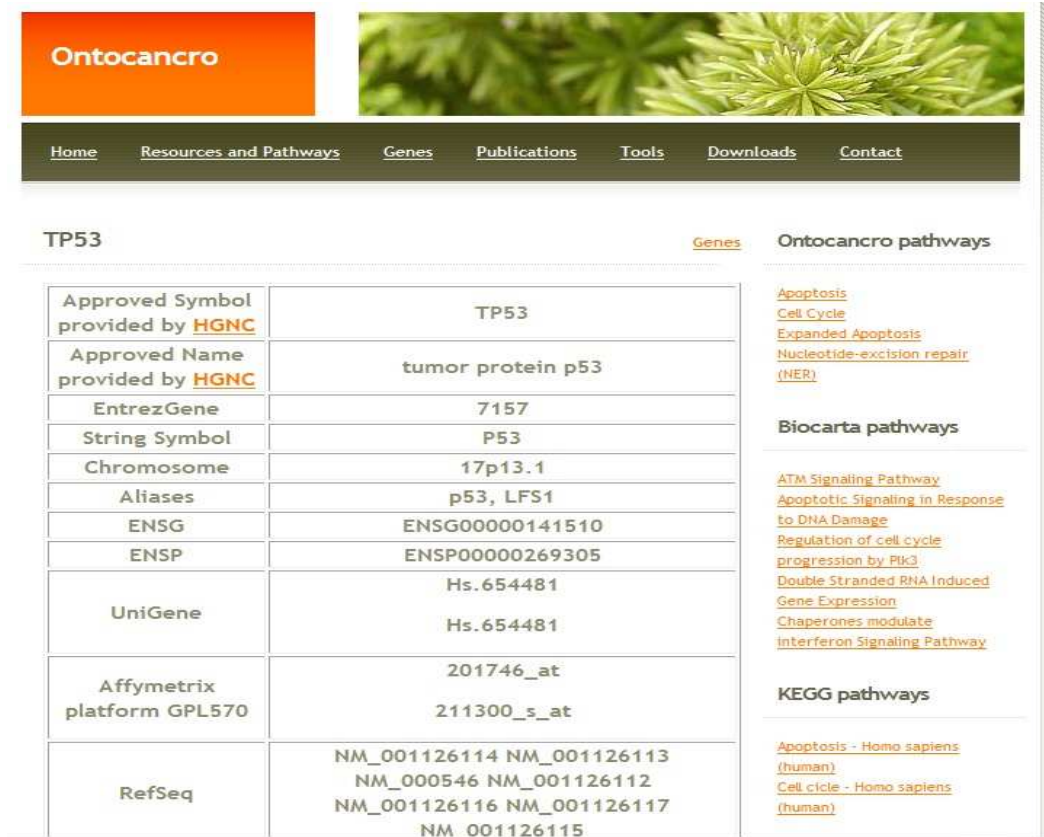
**Figura 3.4:** Lista dos genes na página da Ontocancro

A forma de disponibilização das informações serviu como parâmetro para o desenvolvimento da forma de apresentação das informações no iPhone.

O aplicativo desenvolvido baseia-se em uma sequência de passos definida e exibida na Figura 3.6.

Primeiramente, quando o usuário entrar no sistema será exibido a tela inicial do sistema contendo informações sobre o projeto Ontocancro e formas de contato com os responsáveis. A partir do menu inferior (Tab Bar Controller) o usuário tem a opção de trocar entre as tela (*view*) de genes, *pathways* e imagens. Ao clicar na aba genes serão listados todos os genes, em seguida, permite a seleção de um gene em específico para mostrar seus detalhes. Ao clicar na aba *pathways* serão exibidos os *pathways* da ontologia Ontocancro. Esta permite selecionar um *pathway* para mostrar os seus genes relacionados. Se o usuário quiser poderá exibir os detalhes

de cada gene. Na aba imagem são apresentadas as imagens gráficas para o usuário ver a ontologia Ontocancro.



**TP53** [Genes](#) [Ontocancro pathways](#)

Approved Symbol provided by <a href="#">HGNC</a>	TP53
Approved Name provided by <a href="#">HGNC</a>	tumor protein p53
EntrezGene	7157
String Symbol	P53
Chromosome	17p13.1
Aliases	p53, LFS1
ENSG	ENSG00000141510
ENSP	ENSP00000269305
UniGene	Hs.654481
Affymetrix platform GPL570	201746_at 211300_s_at
RefSeq	NM_001126114 NM_001126113 NM_000546 NM_001126112 NM_001126116 NM_001126117 NM_001126115

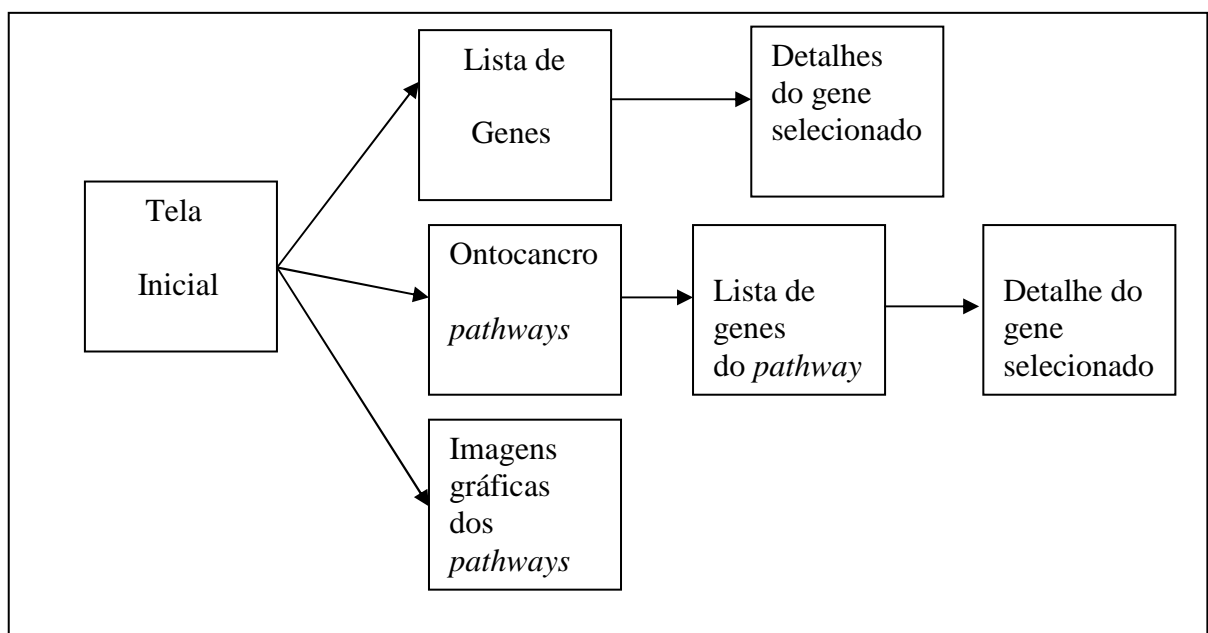
**Biocarta pathways**

- [Apoptosis](#)
- [Cell Cycle](#)
- [Expanded Apoptosis](#)
- [Nucleotide-excision repair \(NER\)](#)

**KEGG pathways**

- [Apoptosis - Homo sapiens \(human\)](#)
- [Cell cycle - Homo sapiens \(human\)](#)

**Figura 3.5:** Mostra informações detalhadas sobre o gene.



**Figura 3.6:** Estrutura e sequência de acesso as informações no aplicativo

Frente a estas especificações, os controladores escolhidos para a implementação do aplicativo foram: um *Tab Bar Controller*, para permitir a troca entre *view* diferentes; dois *Navigation Controller*, para gerenciar a hierarquia entre as telas de exibição; um *View Controller*, para mostrar as informações da tela inicial; e, por fim, alguns *Table View Controllers*, um para a tela de listagem de genes, detalhes dos genes, listagem os *pathways*.

Tanto o *View Controller* quanto os *Table View Controllers* são submissos às ações tomadas pelo *Navigation Controller*, estando neste último toda a lógica de controle de qual tela será exibida em determinado momento. As responsabilidades do *View Controller* e dos *Table View Controllers* consistem em gerenciar as informações presentes nas telas associadas a cada um deles.

As classes necessárias para a implementação do aplicativo desenvolvido nesse trabalho, bem como sua estrutura, são descritas abaixo:

- Gene: armazena os dados referentes ao gene obtidos no *parser* do arquivo XML contendo as informações sobre a ontologia Ontocancro;
- Pathway: armazena os dados referentes a cada *pathway* obtidos no *parser* do arquivo XML contendo as informações sobre a ontologia Ontocancro;
- Relation: armazena os dados referentes da relação dos *pathways* e genes;
- InicioViewController: herda as propriedades de um *View Controller*, portanto, não possui nenhuma característica específica já definida. Ela controla somente uma tela para exibir informações sobre o projeto Ontocancro;
- GeneViewController: herda suas características de um *Table View Controller*, esta é responsável por exibir a lista de genes encontrados no sistema em tabela do iPhone. Também é papel desta classe oferecer um meio de inicialização do *Navigator Controller* e permitir a busca por um gene específico nos genes listados;
- DetalheGeneViewController: exibe os dados detalhados do gene selecionado.
- PathwayViewController: herda suas características de *View controller* e é responsável por mostrar os *pathways* da ontologia Ontocancro e inicializar o *Navigator Controller* para utilização na aba *pathway*;
- PathwayGeneViewController: herda suas características de *Table View Controller* e é responsável pela listagem de todos os genes relacionados ao *pathway*;

- `DetalhePathwayGeneViewController`: exibe as informações sobre o gene selecionado.

A seguir, nas Figuras 3.7, 3.8 e 3.9 são apresentados o diagrama de classes das principais classes utilizadas no projeto .

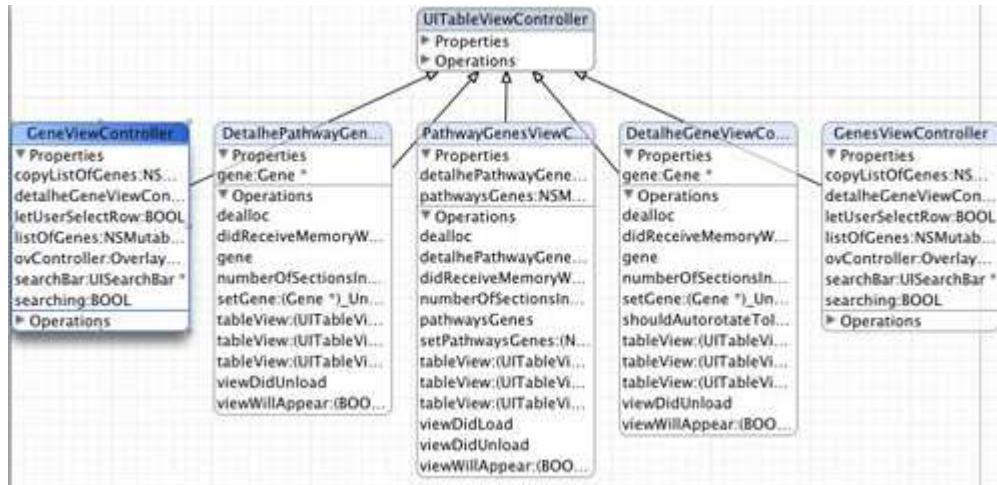


Figura 3.7: Classes que herdam de UITableViewController

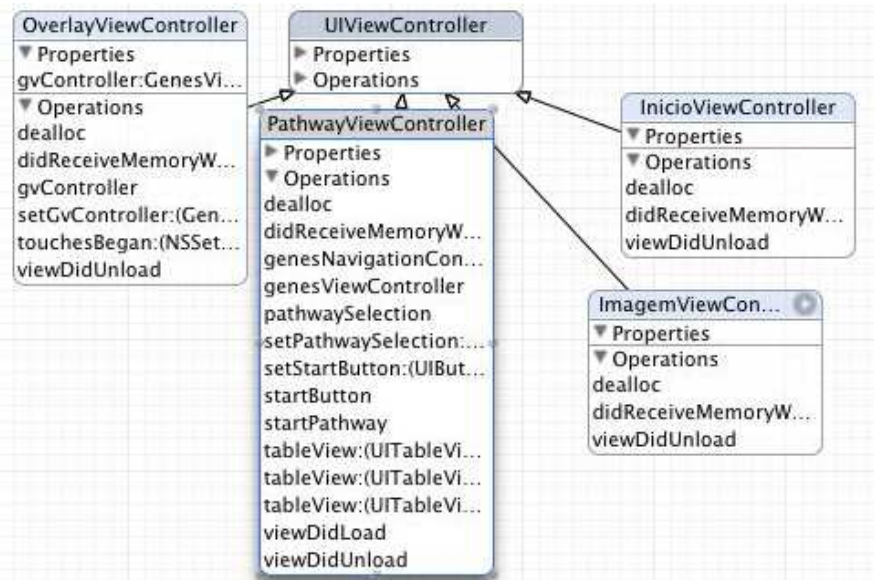
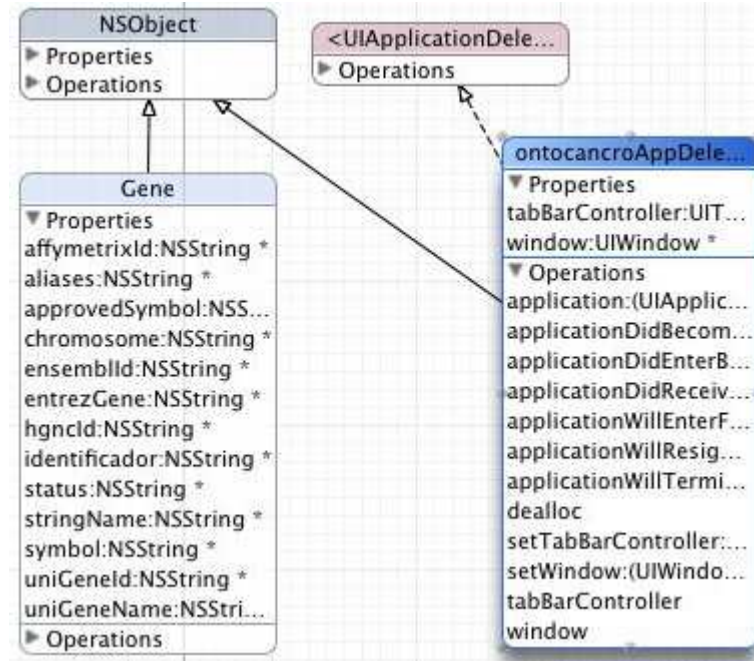


Figura 3.8: Classes que herdam de UIViewController



**Figura 3.9:** Classe gene que é filha de NSObject e a classe app da aplicação

### 3.4 Descrição das Interfaces

Após a finalização da codificação, obteve-se diversas telas como resultado do aplicativo criado e estas serão exibidas e detalhadas nesta sessão.

Na Figura 3.10 mostra a tela inicial do aplicativo, contendo informações iniciais sobre o projeto Ontocancro e forma de contato com os responsáveis pelo projeto. Conforme pode ser visto nas Figuras 3.10, 3.11, 3.12 e 3.13 existe na parte inferior da tela uma aba contendo opções de alteração de tela.

Ao clicar na aba genes serão listados todos os genes da ontologia ontocancro, conforme a Figura 3.11. Na parte superior da tela é oferecida a opção para o usuário buscar o gene listado de um forma mais rápida. Ao selecionar um gene específico será exibido a Figura 3.12 contendo as informações detalhadas de cada gene.

Ao clicar na aba *pathway* serão exibidos os *pathway* da ontologia Ontocancro, como mostrado na Figura 3.13. Após o usuário selecionar uma das vias serão exibidos todos os genes relacionados a esta via, e para em seguida permitir mostrar as informações detalhadas de um gene selecionado, como mostram as Figuras 3.14 e 3.15 respectivamente.



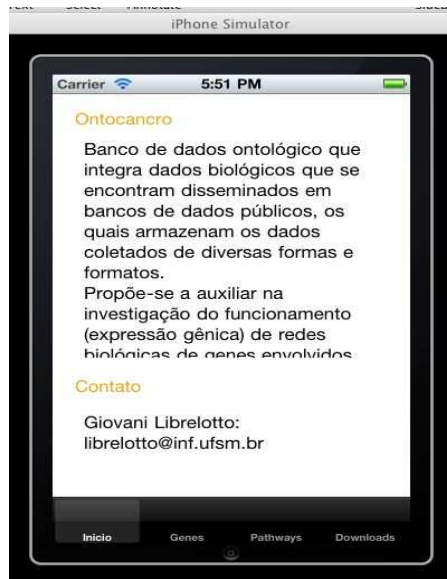


Figura 3.10: Tela inicial do aplicativo



Figura 3.11: Lista de genes e com a exibição do teclado auxiliar para buscar e selecionar um gene.



Figura 3.12: Informações detalhadas do gene selecionado



Figura 3.13: Pathways da ontologia Ontocancro



Figura 3.14: Listagens dos genes relacionados ao pathway



Figura 3.15: Informações detalhadas do gene selecionado relacionados ao pathway

### 3.5 Testes e análise dos dados

Ao inicializar o aplicativo o método *main* da classe *main* é executado. Neste método é inicializado o programa e além disso chama o método *didFinishLaunchingWithOptions* da classe *ontocancroAppDelegate*, responsável pela apresentação da interface, modelada no Interface Builder, mostrada na Figura 3.10. Este método também é responsável pela invocação do método *initXMLParser* da classe *XMLParser*, responsável pelo início da leitura do arquivo XML (comentado na seção 3.2.1). A leitura é realizada com o auxílio da biblioteca NSParser, disponibilizada pela próprio SDK para iPhone. A biblioteca NSParser disponibiliza os métodos necessário para a leitura do arquivo XML, ou seja, somente é necessário repassar ao método ( *didStartElement* ) a tag de início e para outro método a tag de finalização (*didEndElement* ), e por fim um terceiro método (*foundCharacters* ) retorna a informação contida no meio das tag de começo e finalização.

O arquivo XML gerado para iPhone contém as informações sobre todos os genes da Ontocancro, os *pathways* e as suas relações. Por isso, foram criadas três classes distintas para armazenar estas informações e o conjunto destas informações e armazenado em seu respectivo vetor. A aplicação posteriormente utiliza estes *arrays* de genes, *pathways* e relações para mostrar as informações sobre os genes da ontologia Ontocancro.

Após a inicialização do aplicativo ao tocar sobre aba genes, na parte inferior da interface, é invocada a classe *GeneViewController* e apresentada a interface, modelada no Interface Builder, da Figura 3.11. Ela herda as características da classe *UITableViewController*. A classe *UITableViewController* é herdada por outras classes do aplicativo. Esta classe é útil pois disponibiliza métodos para a listagem dos dados em uma tabela e, além disso, disponibiliza um método (*didSelectRowAtIndexPath*) que permite criar uma navegação sequencial entre interfaces para apresentação de dados relacionados. Na Figura 3.11 são listados os genes da ontologia Ontocancro, lidos anteriormente e armazenados em um vetor, e criado uma navegação entre a interface de listagens dos genes e a interface de informação detalhada do gene selecionado. Na parte superior da interface da Figura 3.11 é apresentado também a opção de pesquisa por gene, dado que a listagem

completa da ontologia Ontocancro é formada no momento por 1428 genes. Ao tocar sobre um gene específico (por exemplo ABL1) o método *didSelectRowAtIndexPath* é executado. Este constrói a navegação entre as interfaces de listagens dos genes e a informação detalhada do gene ABL1. Além disso, invoca o *cellForRowAtIndexPath* da classe *DetalheGeneViewController*, responsável pela apresentação das informações sobre o gene ABL1 e controle das ações na interface. Como todas as informações sobre os genes não são visualizadas na tela do iPhone, basta rolar para baixo para obter as outras informações referentes ao gene selecionado. Na parte superior da interface da Figura 3.12 aparece a barra de navegação na qual se encontra o botão rotulado por Genes. Ao tocar sobre este permite retornar a interface de listagens dos genes para selecionar qualquer outro gene e visualizar as informações detalhadas.

A outra opção disponibilizada pelo aplicativo é a visualização dos *pathways* da ontologia Ontocancro. Ao tocar na aba *pathways* é invocada a classe *PathwaysViewController*, responsável pela listagens dos *pathways* da Ontocancro (Figura 3.13) e o controle de ações na interface. Esta classe herda as características de *UIViewController*. A interface da Figura 3.13 permite selecionar um *pathway* (por exemplo Apoptosis) e ao tocar sobre o botão *Ver Pathways* serão listados os genes relacionados ao *pathway* apoptosis da Ontocancro (Figura 3.14). Ou seja, ao tocar no botão *Ver Pathways* é invocada a classe *PathwayGenesViewController*, responsável pela listagens dos genes relacionados ao *pathway* selecionado (no exemplo os genes relacionados ao *pathway* apoptosis) e construção da navegação entre a interface de listagens dos genes do *pathway* e a interface de visualização das informações do gene selecionado. Ao tocar sobre um gene é invocada a classe *DetalhePathwayGeneController* responsável pela apresentação da interface da Figura 3.15.

## 4 CONCLUSÃO

Neste trabalho foram apresentados conceitos sobre o desenvolvimento de aplicativos para dispositivos móveis, mais especificamente para o telefone celular multi-funcional da Apple, o iPhone. Acompanhou-se o desenvolvimento e implementação de um estudo de caso focado no acesso aos dados da ontologia Ontocancro.

Frente aos desafios encontrados, como a dificuldade em obter uma plataforma adequada à programação para iPhone, e apesar de certos imprevistos no momento de implementação, tendo em vista os objetivos primários deste trabalho, seu resultado pode ser considerado satisfatório, pois o aplicativo implementado possibilita ao seu usuário a busca/consulta de genes na ontologia Ontocancro.

A programação para iPhone mostra-se desafiadora e instigante, pois as possibilidades que essa arquitetura disponibiliza são consideráveis. Idéias das mais diversas tem surgido e, com isso, a gama de aplicativos para o dispositivo da Apple tem crescido exponencialmente.

Espera-se que o aplicativo construído sirva de base para a continuação do desenvolvimento de novas funcionalidades para o acesso aos dados da ontologia Ontocancro. Além disto, em um futuro próximo, pretende-se refinar a maneira como os dados são exibidos pelo aplicativo, bem como adicionar novas funcionalidades.

Entre os trabalhos futuros que se pode citar é a edição da ontologia Ontocancro, em modo administrador. Ou seja, possibilitar alterar os genes da base de dados da ontologia. Além disso, permitir a atualização sincronizada do arquivo XML com a versão contida no servidor. E, por fim, para contornar esta situação oferecer uma versão online do aplicativo.

## 5. REFERÊNCIAS

APPLE. The Objective-C 2.0 Programming Language. [S.l.: s.n.], 2010. Disponível: <http://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/ObjectiveC/ObjC.pdf>

APPLE. iPhone Application Programming Guide. [S.l.: s.n.], 2010. Disponível em: <http://developer.apple.com/iphone/library/documentation/iPhone/Conceptual/iPhoneOSProgrammingGuide/iPhoneAppProgrammingGuide.pdf>.

APPLE iPhone. Disponível em <http://www.apple.com/iphone/>.

FERREIRA, Ronaldo Rodrigues. Introdução à Bioinformática. Site da Universidade Federal do Rio Grande do Sul - UFRGS. Disponível em: <http://www.inf.ufrgs.br/~rrferreira/bioinf/Apresentacoes/introducaoBioinf.pdf>. Acesso em: 25 Set 2010.

GIBAS, Cynthia e Jambeck, Per. Desenvolvendo Bioinformática: ferramentas de software para aplicações em biologia. Rio de Janeiro : Campus, 2001.

Librelotto, G. R. ; Ramalho, J. C. ; Henriques, P. R. ; Gassen, J. B. ; Turchetti, R. C. A framework to specify, extract and manage topic maps driven by ontology. In: International Conference on Design of Communication (ACM SIGDOC), 2008, Lisboa. p. 155-162. ISBN:978-1-60558-083-8

Librelotto, G. R. ; Mombach, J. C. M. ; Sinigaglia, M. ; Simão, E. ; Cabral, H. B. ; Castro, M. A. A. ; An Ontology to Integrate Transcriptomics and Interatomics Data involved in Gene Pathways of Genome Stability. BSB 2009, LNBI 5676, pp. 164-167, 2009.

MARCONDES, Roberto C. Jr. Universidade de São Paulo: Instituto de Matemática e Estatística – IME Disponível em: <http://www.ime.usp.br/~cesar/projects/lowtech/setemaiiores/dna.htm>

Mombach, J. C. M. ; Castro, M. A. A. ; Almeida, Rita M. C. de ; Moreira, J. C. F. Impaired expression of NER gene network in sporadic solid tumors. *Nucleic Acids Research*, v. 35, p. 1859-1867, 2007.

Ontocancro: <http://www.ontocancro.org/>

PROSDOCIMI, Francisco; Cerqueira, Gustavo C.; Filho, Fernando C.; Binneck, Eliseu. *Bioinformática: Manual do Usuário*. 2003. pp. 18-31.

ROCHA, Miguel. *Bioinformática: passado, presente e futuro!!* Bragança, Portugal : s.n., 11 de Maio de 2006.

SETUBAL, João Carlos. *ComCiência: A origem e o sentido da bioinformática*. Disponível em: <http://www.comciencia.br/reportagens/bioinformatica/bio10.shtml>. Acesso em: 10 Ago. 2010.

Simão, E. M. ; Cabral, H. C. ; Castro, M. A. A. ; Sinigaglia, M. ; Mombach, J. C. M. ; Librelotto, G. R. Modeling the Human Genome Maintenance network. *Physica A: Statistical Mechanics and its Applications*, Article in Press, 2010.

TEIXEIRA, Mônica. *O Projeto Genoma Humano*. São Paulo : Publifolha, 2000.

WEISER M., 1991. The computer for the 21st century. *Scientific American*, pp. 94-104.



## Anexo A – Template XSL

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xtm="http://www.topicmaps.org/xtm/1.0/">

  <xsl:output method="xml" encoding="UTF-8" indent="yes" omit-xml-declaration="no"/>
    <xsl:template match="text()" priority="-1"/>
    <xsl:template match="text()" priority="-1" mode="genes"/>
    <xsl:template match="text()" priority="-1" mode="pathway"/>
    <xsl:template match="text()" priority="-1" mode="assoc"/>

    <xsl:template match="/">
      <ontocancro>
        <genes>
          <xsl:apply-templates mode="genes"/>
        </genes>
        <pathways>
          <xsl:apply-templates mode="pathway"/>
        </pathways>
        <relations>
          <xsl:apply-templates mode="assoc"/>
        </relations>
      </ontocancro>
    </xsl:template>

    <xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_Molecule']" mode="genes">
      <gene>
        <ID>
          <xsl:value-of select="@id"/>
        </ID>
        <Symbol>
          <xsl:value-of select="xtm:baseName[not(xtm:instanceOf)]/xtm:baseNameString"/>
        </Symbol>
        <ApprovedSymbol>
          <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_ApprovedSymbol']/xtm:resourceData"/>
        </ApprovedSymbol>
        <entrezGene>
          <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_EntrezGeneIDbyNCBI']/xtm:resourceData"/>
        </entrezGene>
        <StringName>
          <xsl:value-of select="xtm:baseName[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_StringName']/xtm:baseNameString"/>
        </StringName>
        <UniGeneName>
          <xsl:value-of select="xtm:baseName[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_UniGeneName']/xtm:baseNameString"/>
        </UniGeneName>
        <Chromosome>
          <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_Chromosome']/xtm:resourceData"/>
        </Chromosome>
        <AffymetricsID>
          <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_AffymetricsID']/xtm:resourceData"/>
        </AffymetricsID>
        <Aliases>
          <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_Aliases']/xtm:resourceData"/>
        </Aliases>
        <Status>

```

```

        <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_Status']/xtm:resourceData"/>
    </Status>
    <Ensembl_ID>
        <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_Ensembl_ID']/xtm:resourceData"/>
    </Ensembl_ID>
    <HGNC_ID>
        <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_HGNC_ID']/xtm:resourceData"/>
    </HGNC_ID>
    <UniGeneID>
        <xsl:value-of select="xtm:occurrence[xtm:instanceOf/xtm:topicRef/@xlink:href =
'#id_UniGeneID']/xtm:resourceData"/>
    </UniGeneID>
</gene>
</xsl:template>

<xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_PathwayBiocarta']"
mode="pathway">
    <pathway>
        <source>Biocarta</source>
        <ID>
            <xsl:value-of select="@id"/>
        </ID>
        <name>
            <xsl:value-of select="xtm:baseName/xtm:baseNameString"/>
        </name>
    </pathway>
</xsl:template>
<xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_PathwayGO']"
mode="pathway">
    <source>GO</source>
    <pathway>
        <ID>
            <xsl:value-of select="@id"/>
        </ID>
        <name>
            <xsl:value-of select="xtm:baseName/xtm:baseNameString"/>
        </name>
    </pathway>
</xsl:template>
<xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_PathwayNCI']"
mode="pathway">
    <pathway>
        <source>NCI</source>
        <ID>
            <xsl:value-of select="@id"/>
        </ID>
        <name>
            <xsl:value-of select="xtm:baseName/xtm:baseNameString"/>
        </name>
    </pathway>
</xsl:template>
<xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_PathwayOntocancro']"
mode="pathway">
    <Pathway>
        <source>Ontocancro</source>
        <ID>
            <xsl:value-of select="@id"/>
        </ID>
        <name>
            <xsl:value-of select="xtm:baseName/xtm:baseNameString"/>
        </name>
    </Pathway>
</xsl:template>

```

```

<xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_PathwayKEGG']"
mode="pathway">
  <Pathway>
    <source>KEGG</source>
    <ID>
      <xsl:value-of select="@id"/>
    </ID>
    <name>
      <xsl:value-of select="xtm:baseName/xtm:baseNameString"/>
    </name>
  </Pathway>
</xsl:template>
<xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_PathwayReactome']"
mode="pathway">
  <Pathway>
    <source>Reactome</source>
    <ID>
      <xsl:value-of select="@id"/>
    </ID>
    <name>
      <xsl:value-of select="xtm:baseName/xtm:baseNameString"/>
    </name>
  </Pathway>
</xsl:template>
<xsl:template match="xtm:topic[xtm:instanceOf/xtm:topicRef/@xlink:href = '#id_PathwayProsite']"
mode="pathway">
  <Pathway>
    <source>Reactome</source>
    <ID>
      <xsl:value-of select="@id"/>
    </ID>
    <name>
      <xsl:value-of select="xtm:baseName/xtm:baseNameString"/>
    </name>
  </Pathway>
</xsl:template>

<xsl:template match="xtm:association[xtm:member[1]/xtm:roleSpec/xtm:topicRef/@xlink:href = '#id_belongs-to']"
mode="assoc">
  <Relation>
    <PathwayID>
      <xsl:value-of select="substring-after(xtm:member[2]/xtm:topicRef/@xlink:href, '#)"/>
    </PathwayID>
    <GeneID>
      <xsl:value-of select="substring-after(xtm:member[1]/xtm:topicRef/@xlink:href, '#)"/>
    </GeneID>
  </Relation>
</xsl:template>

</xsl:stylesheet>

```