

Universidade Federal de Minas Gerais  
Instituto de Ciências Exatas  
Departamento de Ciência da Computação

**SISTEMA DE TEMPO REAL PARA PREVENÇÃO DE COLISÃO EM  
CRUZAMENTOS**

RELATÓRIO TÉCNICO DA DISCIPLINA DE SISTEMAS DE TEMPO REAL

Prof.: Sergio Vale Aguiar Campos

Kattiana Constantino  
kattiana@gmail.com

Samir P. Khalifa  
khalifa@dcc.ufmg.br

Vitor Venâncio  
vitorvd@dcc.ufmg.br

Belo Horizonte – MG  
2013 / 2º semestre

## **Resumo**

Este trabalho apresenta o desenvolvimento de um simulador de trânsito com características em tempo real. O objetivo principal do simulador é ajudar os motoristas de carros e os motoristas de ambulâncias a se deslocarem pelas vias de maneira segura e sem causar acidentes nos cruzamentos.

**Palavras-chave:** Simulação, Trânsito, Prevenção de Colisão, Tempo Real.

## **Abstract**

This study presents the development of a traffic simulator with real-time characteristics. The main purpose of the simulator is to help car drivers and drivers of ambulances to move by the routes safely and without causing accidents at intersections.

**Keywords:** Simulation, Traffic, Collision Avoidance, Real-time.

## **Lista de Siglas**

UFMG	Universidade Federal de Minas Gerais
ODBC	Open Database Connectivity

## **Lista de Tabelas**

Tabela 3.1	Cronograma de realização das etapas .....	16
------------	---	----

## **Lista de Figuras**

Figura 1.1	Entidades envolvidas na pesquisa: LUAR, HC de Belo Horizonte e ELSA	..	8
------------	---	----	---

# Sumário

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>8</b>
1.1	Motivação . . . . .	8
1.2	Objetivo . . . . .	9
1.3	Organização do Trabalho . . . . .	9
<b>2</b>	<b>CONTEXTUALIZAÇÃO E CONCEITOS RELEVANTES .....</b>	<b>10</b>
2.1	Exame de Eletrocardiograma . . . . .	10
2.2	Gerência Integrada de Sistemas de Diagnóstico Clínico . . . . .	11
2.3	Software Pyramis . . . . .	11
2.4	Código Minnesota . . . . .	11
<b>3</b>	<b>METODOLOGIA .....</b>	<b>13</b>
3.1	Tipo de Pesquisa . . . . .	13
3.2	Especificidades Técnicas . . . . .	13
3.3	Execução de Atividades . . . . .	15
<b>4</b>	<b>DESENVOLVIMENTO DO TRABALHO .....</b>	<b>17</b>
4.1	Características do Simulador . . . . .	17
4.2	Características do Sistema em Tempo Real . . . . .	18
4.3	Implementação . . . . .	19
<b>5</b>	<b>RESULTADOS E DISCUSSÃO .....</b>	<b>20</b>
<b>6</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS .....</b>	<b>21</b>

<b>Referências Bibliográficas .....</b>	<b>22</b>
---	-----------



# 1 INTRODUÇÃO

Esse relatório apresenta o trabalho desenvolvido na disciplina de Sistemas De Tempo Real ofertada no curso de Ciência da Computação da Universidade Federal de Minas Gerais (UFMG).



Figura 1.1: Entidades envolvidas na pesquisa: LUAR, HC de Belo Horizonte e ELSA

## 1.1 Motivação

O cenário que engloba o trânsito em ruas de uma cidade proporciona observações interessantes sobre os acontecimentos envolvidos nesse ambiente. Acidentes entre veículos não são raros e na maioria das vezes envolve erro humano que poderia ser evitado com implementações de comportamentos automáticos nas entidades mecânicas.

Uma situação ideal pode ser ilustrada com carros que agem automaticamente de acordo com as características que o trânsito proporciona, principalmente em cruzamentos ou situações de risco iminente. E, seguindo essa modelagem, um sistema de tempo real deve existir para que os componentes sejam monitorados e comandados de maneira eficaz e correta.

A proposta desse trabalho é desenvolver um simulador de trânsito com características de tempo real. Tal simulador possui um sistema centralizado com a função de orientar os motoristas sobre qual a melhor decisão a ser tomada ao aproximar-se de um cruzamento. O objetivo nesse contexto é evitar colisões com outros veículos e, na medida do possível, minimizar o tempo de espera para atravessar um cruzamento com segurança.

## **1.2 Objetivo**

## **1.3 Organização do Trabalho**

No capítulo 2 são apresentadas as características do simulador, os conceitos do sistema em tempo real, suas restrições, as tarefas do sistema, o escalonamento e suas abordagens, ainda nesse capítulo também são apresentadas a modelagem e as informações técnicas sobre a implementação do sistema. O capítulo 3 apresenta os cenários de testes e os testes realizados para validar o escalonamento das tarefas e o funcionamento do simulador. Concluímos com o capítulo 5 que apresenta uma análise final sobre esse trabalho e também, apresenta propostas de trabalhos futuros.

Para melhor entendimento, o texto desse relatório está organizado da seguinte forma: A Seção 2 contextualiza o assunto e apresenta os conceitos envolvidos. A Seção 3 apresenta a metodologia aplicada à pesquisa. A Seção 4 mostra o desenvolvimento do trabalho. A Seção 5 apresenta os resultados obtidos e as eventuais discussões que surgiram ao longo do desenvolvimento. Por fim, na Seção 6 são expostas as conclusões obtidas e alguns temas para trabalhos futuros.

## 2 CONTEXTUALIZAÇÃO E CONCEITOS RELEVANTES

Essa Seção expõe sucintamente os principais elementos para o desenvolvimento do trabalho proposto. Esclarecemos alguns conceitos envolvidos com a realização e avaliação de um ECG e apresentamos com mais detalhes o Pyramis, que foi a principal ferramenta utilizada para o desenvolvimento do trabalho. Além disso, mostramos o código Minnesota e suas especificidades para que possamos entender que a utilização isolada desse código gerado de maneira automática deve ser desencorajada o que serve de motivação para o objetivo de recuperação das medições brutas do ECG.

### 2.1 Exame de Eletrocardiograma

O ECG é o registro dos fenômenos elétricos que se originam durante a atividade cardíaca por meio de um aparelho denominado eletrocardiógrafo. O eletrocardiógrafo é um galvanômetro, aparelho que detecta a diferença de potencial entre dois pontos, que mede pequenas intensidades de corrente que recolhe a partir de dois eletrodos, que são pequenas placas de metal conectadas a um fio condutor, dispostos em determinados pontos do corpo humano. Ele serve como um auxiliar valioso no diagnóstico de grande número de cardiopatias e outras condições como, por exemplo, os distúrbios hidro-eletrolíticos (1). Como pode ser visto na Figura ?? retirada de (1), o registro das medições do eletrocardiógrafo é feito em um papel quadriculado e dividido em quadrados pequenos de 1mm. Cada grupo de cinco quadrados na horizontal e na vertical compreendem um quadrado maior delimitado por uma linha mais grossa. O tempo fica denotado no eixo horizontal e o registro é realizado em uma velocidade de 25 mm/seg. Sendo assim, cada quadrado equivale a 0,04 segundos. Portanto, cinco quadrados, ou 1 quadrado maior, equivalem a 0,2 segundos. A voltagem é denotada no eixo vertical e cada quadrado equivale a 0,1 mVolt.

## 2.2 Gerência Integrada de Sistemas de Diagnóstico Clínico

Os sistemas de gerência integrada de diagnóstico clínico são uma parte indispensável para o funcionamento de uma instituição de saúde moderna (2). Tais sistemas auxiliam a prática da medicina armazenando dados dos pacientes, captando informações geradas por aparelhos de exame, dentre outras funcionalidades. Dependendo da solução de gerenciamento utilizada pode-se alcançar melhorias significativas na eficiência do fluxo de trabalho clínico e reduzir os custos envolvidos nesses serviços. O software alvo dessa pesquisa se insere nessa categoria e alguns outros exemplos de software desse tipo são o Centricity© da General Electric, o MediTouch© e o Medios EHR©. A integração entre a captação e apresentação dos dados é considerado um dos aspectos mais desafiadores da implementação desse tipo de software (3) e o aprimoramento de tal aspecto é um dos objetivos desse trabalho.

## 2.3 Software Pyramis

O Pyramis é um software comercial desenvolvido para gerência integrada de sistemas de diagnóstico por ECG desde a captação dos resultados até a interpretação e armazenamento dos mesmos. As principais características do sistema são: - Escalabilidade: O sistema suporta uma quantidade arbitrária de dados limitada somente à capacidade de armazenamento física do cliente. Além disso, novos dispositivos podem ser adicionados livremente. - Flexibilidade: Diversos tipos de dispositivos são suportados e adicionados a qualquer tempo. - Modularidade: Existem vários níveis de acesso e separação entre módulos de suporte técnico, manutenção de performance e serviços para o profissional de diagnóstico.

## 2.4 Código Minnesota

O código Minnesota é o sistema de classificação mais amplamente utilizado para avaliar ECG e é útil para exames clínicos e epidemiológicos (4). Ele foi inicialmente proposto para codificar e suprir a falta de padronização na interpretação do ECG provendo um framework para documentar a avaliação de uma forma mais clara e definida. Basicamente essa codificação consiste de um conjunto de regras de medição, um sistema de classificação e um conjunto de regras de exclusão de diagnóstico (5).

Mas é importante ressaltar que Código Minnesota é sujeito a erros e por si só não produz a interpretação de um ECG (6). O que codificação realmente permite é uma classificação da morfologia eletrocardiográfica com base em critérios rígidos baseados em características

proeminentes reconhecidas no ECG. Sendo assim, as medições reais do ECG sempre têm de acompanhar a codificação para que uma interpretação confiável seja feita.

## **3 METODOLOGIA**

### **3.1 Tipo de Pesquisa**

A pesquisa consistiu na exploração e aprimoramento de uma ferramenta e o trabalho proposto consiste na aplicação de conhecimentos adquiridos ao longo do curso de bacharelado em Ciência da Computação do Departamento de Ciência da Computação da Universidade Federal de Minas Gerais. Portanto, a natureza da pesquisa (7) está definida e é considerada uma pesquisa de aplicação tecnológica. No que diz respeito aos procedimentos e desenvolvimento do trabalho, nos inserimos na classificação de trabalho exploratório com resultados práticos implementados e utilizados em um contexto real de ampla aplicação.

### **3.2 Especificidades Técnicas**

O ambiente de desenvolvimento do trabalho possui configurações fixas e utiliza sistema operacional Windows 2003 Server© na sua versão Standard Edition 32-bit. Um problema é que o sistema de gerenciamento integrado proporcionado pelo Pyramis encontra-se em uso pelo HC e pelas unidades participantes no projeto ELSA. Portanto, para realizarmos os testes e desenvolvermos o trabalho de forma a não afetarmos o sistema criamos uma cópia do estado do sistema e virtualizamos a sua execução utilizando o software VMWare©. O VMWare encontra-se disponível no site:

<http://www.vmware.com/>

O software Pyramis (8) utiliza um gerenciador de banco de dados IBM DB2© na versão Workgroup Edition 7.2. Sendo assim, temos a disposição um banco de dados relacional com suporte Open Database Connectivity (ODBC). Tal sistema suporta até mil conexões simultâneas e não possui limitação de armazenamento (limitado somente pela capacidade do hardware). O programa fornece uma interface web que deve ser acessada obrigatoriamente pelo navegador Internet Explorer© nas versões 6, 7 ou 8.

Com o intuito de estendermos as funcionalidades do Pyramis criamos scripts que podem ser acoplados ao sistema. Para acessarmos o banco de dados nesses scripts utilizamos a interface do ODBC que unifica a gerência e acesso de bancos de dados heterogêneos, inclusive o IBM DB2. Os scripts foram escritos na linguagem Python versão 3.3 cujo interpretador para várias plataformas pode ser encontrado em:

<http://www.python.org/getit/>

No nosso caso utilizamos o instalador disponível em:

<http://www.python.org/ftp/python/3.3.2/python-3.3.2.msi>

No ambiente do Windows basta rodarmos esse instalador para que o script funcione. Para fazer o acesso à interface ODBC precisamos instalar o módulo PyODBC cujo instalador se encontra em:

<http://code.google.com/p/pyodbc/downloads/detail?name=pyodbc-3.0.7.win-amd64-py2.6>.

A linguagem de programação Python é uma linguagem de alto nível, interpretada, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte. Uma das maiores vantagens dessa linguagem é que sua sintaxe é simples e elegante, ideal para scripts e desenvolvimento de aplicações rápidas (9). Outras referências que foram úteis para o desenvolvimento desse trabalho podem ser encontradas na API Python (10).

Para auxiliar no redesenho do banco de dados e no entendimento de suas relações utilizamos o software MySQL Workbench®. O MySQL Workbench é uma ferramenta gráfica para modelagem de dados, integrando criação e designer. A ferramenta possibilita trabalhar diretamente com objetos schema, além de fazer a separação do modelo lógico do catálogo de banco de dados. Toda a criação dos relacionamentos entre as tabelas pode ser baseado em chaves estrangeiras. Outro recurso que a ferramenta possibilita é realizar a engenharia reversa de esquemas do banco de dados, bem como gerar todos os scripts em SQL. O programa encontra-se disponível em:

[www.mysql.com/products/workbench/](http://www.mysql.com/products/workbench/)

### 3.3 Execução de Atividades

O trabalho foi realizado em um semestre e as tarefas foram divididas em várias etapas. O cronograma a seguir apresenta as etapas desenvolvidas durante o semestre de trabalho.

Etapa 1: **Levantamento bibliográfico.** De 19/08/13 a 30/08/13.

As tarefas dessa etapa consistiram de um levantamento bibliográfico de todo o material relacionado ao Pyramis e seus subsistemas além de um estudo sobre os conceitos básicos envolvidos na realização de um ECG e na geração do código Minnesota.

Etapa 2: **Familiarização com o Pyramis.** De 31/08/13 a 16/09/13.

Nessa etapa estudamos a utilização do Pyramis e exploramos suas funcionalidades. Essa etapa foi importante para determinarmos o desenho do banco de dados interno do programa. Como resultado, identificamos as limitações e problemas do softwares e conseguimos acessar as informações guardadas em seu banco de dados.

Etapa 3: **Levantamento de Requisitos com os Interessados.** De 16/09/13 a 30/09/13.

Passamos a uma fase de interação mais próximas como as partes interessadas do HC na qual determinamos com mais exatidão os requisitos do trabalho ao mesmo tempo em que ajustamos o rumo que a implementação da solução proposta tomou.

Etapa 4: **Implementação e Extensão Efetiva do Sistema.** De 30/09/13 a 12/11/13.

Nessa etapa passamos a trabalhar na implementação efetiva da nossa proposta para extensão do sistema. Construímos scripts que foram acoplados ao Pyramis adicionando novas funcionalidades.

Etapa 5: **Identificação e Recuperação de Dados Perdidos.** De 12/11/13 a 26/11/13.

Devido ao fato de que os pesquisadores reportaram a perda de dados de exames armazenado pelo programa essa etapa se fez necessária. Nessa etapa identificamos em que proporção as perdas estavam ocorrendo, os motivos dessas perdas e propusemos mecanismos de recuperação para esses dados.

Etapa 6: **Escrita da Monografia.** De 19/08/13 a 31/11/13.

Essa etapa ocorreu concomitantemente com as outras etapas do trabalho. Ao final, geramos o texto definitivo da monografia.



A Tabela 3.1, apresentada a seguir, mostra o cronograma de execução das etapas realizadas. A informação contida na tabela marca as semanas ocupadas com as tarefas de cada etapa.

<b>Etapas / Semana</b>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	X	X													
2			X	X											
3					X	X									
4							X	X	X	X	X	X			
5													X	X	
6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Tabela 3.1: Cronograma de realização das etapas

## 4 DESENVOLVIMENTO DO TRABALHO

O desenvolvimento do trabalho ocorreu seguindo as etapas do cronograma e em alguns momentos novas demandas surgiram e adicionaram mais tarefas ao trabalho inicialmente proposto. Nessa Seção iremos descrever os achados e a evolução desse desenvolvimento bem como as principais ferramentas complementares desenvolvidas. Nesse capítulo são apresentadas as requisitos/características que o simulador precisa ter e as suas restrições. Também, são apresentadas todas as características de um sistema em tempo real, as restrições temporais, o escalonamento de tarefas e suas abordagens. Por fim, são apresentados os detalhes sobre a implementação do simulador e do servidor do sistema.

### 4.1 Características do Simulador

As principais características do simulador são: Mapa: o mapa contém toda lógica de trânsito funcionando em tempo real. O mapa é composto de quarteirões cujas dimensões estão definidas em um arquivo de entrada. Ele está modelado como uma grade onde cada célula será um dos seguintes componentes: Via; Cruzamento; Calçada; Radar: Em cada via, próximo a um cruzamento, há um radar que envia informações para o sistema. O radar detecta a direção de cada veículo e o tempo para este chegar ao cruzamento.

Via: Cada via é de mão dupla e contínua, o fim de uma via denota o começo dessa mesma via. Cada veículo tem tráfego livre pelas vias e uma velocidade constante. Ao passar por um cruzamento, o veículo decide aleatoriamente a próxima direção a ser seguida.

Carro: Os estados de um carro são: “em movimento” ou “parado esperando”. O comportamento padrão de um carro é parar impreterivelmente antes de cada cruzamento e esperar até que o sistema autorize-o a continuar a sua trajetória. Nesse caso, essa tarefa apresenta uma característica soft real-time (quando as consequências de uma falha devida ao tempo é da mesma ordem de grandeza que os benefícios do sistema em operação normal[5]), ou seja, o carro fica ocioso, mas não há colisão/acidente. Ambulância: O estado de uma ambulância é sempre em movimento. Ela tem prioridade sobre os carros e não esperam pela decisão do sis-

tema, ou seja, sempre que uma ambulância se aproximar de um cruzamento ela continua a sua trajetória sem paradas ou esperas. Um dos cenários de testes desse trabalho envolve a observar como o sistema se comporta quando há carros e ambulâncias nas vias. Esse caso se caracteriza com uma situação hard real-time (quando as consequências de pelo menos uma falha temporal excedam em muito os benefícios normais do sistema (falha catastrófica)[5]), pois se o sistema não responder no tempo correto, há colisão/acidente.

Nesse trabalho, são avaliados pontos de quebra do sistema nos quais uma quantidade excessiva de veículos (carros e ambulâncias) seja colocada no simulador. Quando isso ocorre o sistema falha e consequentemente as colisões/acidentes acontecem. Porém, os limites do sistema foram determinados com experimentos previstos para a etapa posterior a implementação e estão apresentados no capítulo 4 desse trabalho.

## 4.2 Características do Sistema em Tempo Real

O Sistema em Tempo Real é regido por um relógio cujo iterador é incrementado a cada ciclo. Todas as referências temporais feitas nessa seção devem ser interpretadas como ciclos de relógio (quando no contexto de intervalo de tempo) ou valor do iterador do relógio (quando no contexto de um valor de tempo).

No simulador todos os veículos praticam uma velocidade constante de percorrer uma célula do mapa por ciclo de relógio. Tarefa do Sistema O sistema trata diferentes tipos de tarefas, são elas: processamento de informação e envio ou recebimento de sinais. Todas essas tarefas são aperiódicas e estão listadas a seguir: O professor recomendou fortemente que umas das tarefas do radar seja periódica

TratarInformacaoRadar: essa tarefa é responsável por receber as informações do radar quando esse detecta um veículo passando por ele. A informação é composta pelos seguintes dados: O identificado do veículo; A direção corrente do veículo (norte, sul, leste, oeste); O tempo para a chegada do veículo ao próximo cruzamento. O tempo para chegar ao cruzamento é determinado pela distância para o cruzamento, já que todos os veículos praticam uma velocidade constante. Tendo como base essas informações o sistema calcula a possibilidade de colisão nos cruzamentos e assim decide qual informação enviar para o veículo. O tempo de sistema necessário para realização dessa tarefa será de 1 a 10 ciclos de relógio. SinalizarCarro: essa tarefa é responsável por informar a um veículo parado no cruzamento que ele pode seguir a sua trajetória, passando pelo cruzamento, a partir de um determinado período de tempo. O tempo de sistema necessário para realização dessa tarefa será de 1 a 5 ciclos de relógio. Sinali-

zarAmbulância: essa tarefa é responsável por informa a uma ambulância que ela deve parar por um período de tempo. O tempo de sistema necessário para realização dessa tarefa será de 1 a 5 ciclos de relógio. Nesse caso, se o tempo de resposta for superior ao deadline da tarefa (tempo em que a ambulância chega ao cruzamento) essa tarefa não passa no teste de aceitação e não é tratada pelo sistema. Essa é uma decisão de que será discutida posteriormente.

O tratamento de tarefas que envolver as ambulâncias (veículos impetuosos) deve ter uma prioridade maior do que as dos demais veículos (veículos cautelosos) para que o sistema priorize a prevenção das colisões.

As tarefas têm início quando um veículo é detectado pelo radar. Nesse instante, o sistema recebe as informações sobre a direção que esse veículo está seguindo, verifica qual o cruzamento mais próximo dele e então calcula o tempo que o veículo levará para chegar a esse cruzamento.

O tempo de sistema necessário para a execução de cada tarefa é definido aleatoriamente entre um intervalo de valores, isso simula aspectos da realidade onde o atraso na troca de mensagens e no processamento pode sofrer alguma variação. Outro motivo para essa decisão é que isso torna o escalonamento de tarefas mais heterogêneo e também, mais interessante.

### 4.3 Implementação

A implementação foi feita na linguagem Java de forma colaborativa entre os participantes do grupo. O repositório do projeto pode ser encontrado em: <https://github.com/khalifa/RTCollisionAvoid>

Uma das frentes de implementação foi criar um mapa na forma de um grafo onde os vértices são os radares e as arestas são as vias. Os veículos passaram a ter origem e destino, então foi desenvolvido o algoritmo de Busca em Largura, para encontrar o menor caminho entre os dois pontos. Cada veículo agora sabe o seu percurso e quando chegar a um radar (vértice do grafo), esperará ou não para seguir em frente. Ainda nessa etapa, foi criada uma fila com todos os radares para que esse seja tratado pelo servidor.

As próximas etapas dessa frente é criar as filas para os veículos cautelosos e para os veículos de ambulância e o algoritmo de escalonamento de tarefas para o tratamento das filas. Uma extensão desse cenário é ativar ou desativar algum(s) radares, assim as vias para chegar ao(s) radar(es) serão disponibilizadas ou interditadas, respectivamente; e em tempo real um novo percurso deverá ser recalculado.

## **5 RESULTADOS E DISCUSSÃO**

Os objetivos propostos nessa pesquisa foram alcançados de forma satisfatória e são apresentados nessa Seção.

## 6 CONCLUSÕES E TRABALHOS FUTUROS

Como proposto, foi desenvolvido um simulador em tempo real com o objetivo principal de minimizar as colisões entre carros e ambulâncias em um cruzamento. Através das características propostas para o simulador e as tarefas definidas para o sistema, foi possível acompanhar em tempo real o comportamento do fluxo de veículos pelas vias de um mapa e o funcionamento dos radares considerando alguns cenários de testes, tais como, tráfego leve (poucos carros), tráfego pesado (muitos carros). Cada situação foi oriunda de demanda sem horários específicos, onde a lógica de comando dos radares no modelo computacional foi acionada pelo controlador lógico programável.

Foi possível observar e coletar em tempo real os dados do modelo de simulação e do sistema de controle. Assim, através de tabelas e gráficos, pôde-se avaliar o sistema e constatar que o modelo de simulação atendeu aos diferentes estímulos das variáveis relativas à dinâmica do sistema. Da mesma forma, o modelo permitiu avaliar o comportamento de cada uma das vias com radares e dos comandos de ativação/desativação oriundos do controlador.

Portanto, espera-se que .....  
também possam ser analisadas e testadas com a abordagem aqui apresentada.

## Referências Bibliográficas

- 1 RAMOS Ângela P.; SOUSA, B. S. Eletrocardiograma: princípios, conceitos e aplicações. *Centro de Estudos de Fisiologia do Exercício Universidade Federal de São Paulo*, 2007.
- 2 MCCORMACK, M. Compare medical practice management software. *Software Selection Advice*, 2013.
- 3 ASSOCIATION, A. M. How to select a practice management system. *AMA-ASSN*, 2013.
- 4 KORS, J. A.; CROW, R. S. Comparison of computer-assigned minnesota codes. *American Journal of Epidemiology*, 2000.
- 5 KORS, J. A.; WU, J. G. Validation of a new computer program for minnesota coding. *J Electrocardiol*, 1996.
- 6 MACFARLANE, P. Minnesota coding and the prevalence of ecg abnormalities. *US National Institutes of Health*, 2000.
- 7 JUNG, C. F. *Metodologia para pesquisa e desenvolvimento: aplicada a novas tecnologias, produtos e processos*. 1. ed. [S.l.]: Axcel Books do Brasil Editora. Rio De Janeiro/RJ, 2004.
- 8 CARDIAC Science Corporation. Pyramis ECG Management. Junho 2013. Disponível em: <<http://www.cardiacscience.com/assets/008/5959.pdf>>.
- 9 MARTELLI, A. R. e. D. A. A. *Python Cookbook*. 3. ed. [S.l.]: Axcel Books do Brasil Editora. Rio De Janeiro/RJ, 2013.
- 10 PYTHON 3.3 API Reference Manual. Junho 2013. Disponível em: <<http://docs.python.org/3.3/c-api/>>.