



IGOR HORTA CORREA

UMA FERRAMENTA DE *E-COMMERCE*
PARA ELABORAÇÃO E COMPRA DE
COMPUTADORES COM *HARDWARE*
CUSTOMIZADO

LAVRAS - MG

2015

IGOR HORTA CORREA

**UMA FERRAMENTA DE *E-COMMERCE* PARA
ELABORAÇÃO E COMPRA DE COMPUTADORES COM
HARDWARE CUSTOMIZADO**

Monografia apresentada ao Colegiado do
Curso de Sistemas de Informação, para
a obtenção do título de Bacharel em
Sistemas de Informação.

Orientador

MSc. Renato Resende Ribeiro de Oliveira

Coorientador

Dr. Denilson Alves Pereira

LAVRAS - MG

2015

**Ficha Catalográfica Elaborada pela Coordenadoria de Produtos
e Serviços da Biblioteca Universitária da UFLA**

Uma ferramenta de *e-commerce* para elaboração e compra de computadores com *hardware* customizado/ Igor Horta Correa. – Lavras : UFLA, 2015.

71 p. : il.

Monografia (graduação) – Universidade Federal de Lavras, 2016.

Orientador: .

Bibliografia.

1. Redes de sensores sem fio. 2. Programação genética. 3. Algoritmo genético. 4. Middlewares. I. Universidade Federal de Lavras. II. Título.

CDD – 004.6

IGOR HORTA CORREA

**UMA FERRAMENTA DE *E-COMMERCE* PARA
ELABORAÇÃO E COMPRA DE COMPUTADORES COM
HARDWARE CUSTOMIZADO**

Monografia apresentada ao Colegiado do
Curso de Sistemas de Informação, para
a obtenção do título de Bacharel em
Sistemas de Informação.

APROVADA em 24 de janeiro de 2016.

Dr. Denilson A. Pereira UFLA

Dr. Ahmed A. A. Esmin UFLA

Dr. Raphael W. de Bettio UFLA

MSc. Renato Resende Ribeiro de Oliveira
Orientador

LAVRAS - MG

2015

Dedico ao turing

AGRADECIMENTOS

Agradeço aos nbs.

Agradeço também aos turings.

RESUMO

Resumo da monografia

Palavras-chave: E-Commerce. Tecnologia da Informação. Extração de Dados. Mineração de Dados.

ABSTRACT

The english translation.

Keywords: E-Commerce. Information Technology. Data Extraction. Data Mining.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

HTML	Hypertext Markup Language
JS	JavaScript

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Contextualização e Motivação	12
1.2	Objetivos Gerais e Específicos	12
1.3	Metodologia.....	12
1.4	Organização do Trabalho	13
2	REFERENCIAL TEÓRICO	14
2.1	Redes de Sensores sem Fio.....	14
2.2	Simuladores de RSSF	15
2.3	Middlewares	17
2.4	Algoritmos Genéticos	18
2.5	Programação Genética.....	20
3	METODOLOGIA	22
4	RESULTADOS E DISCUSSÃO	23
5	CONCLUSÃO E TRABALHOS FUTUROS.....	24

1 INTRODUÇÃO

1.1 Contextualização e Motivação

Com o aumento dos requisitos mínimos para se utilizar jogos e programas atuais e com o aumento de possíveis combinações de peças de computadores, a tarefa de customizar seu próprio computador para executar tarefas específicas se torna cada vez mais difícil. Visando facilitar esta tarefa, este trabalho propõe um sistema que utilize extração de dados para obter preços de diferentes peças de computadores em diferentes lojas de e-commerce e que então os utilize para auxiliar o usuário montar sua própria máquina visando o melhor custo benefício.

1.2 Objetivos Gerais e Específicos

O objetivo geral deste trabalho é uma aplicação web que realize a comparação de preços de peças de computadores em diferentes sites de e-commerce e que auxilie o usuário a comprar seu próprio desktop.

Visando atingir este objetivo geral, os objetivos específicos são definidos a seguir:

- Desenvolver um algoritmo eficiente que seja capaz de coletar informações de produtos em diferentes sites de e-commerce.
- Desenvolver uma interface para aplicação.
- Modelar o banco de dados da aplicação.

1.3 Metodologia

Breve descrição da metodologia que foi utilizada...

1.4 Organização do Trabalho

O trabalho está organizado da seguinte forma, na Seção 2 será apresentado um referencial teórico acerca dos conceitos utilizados. Na Seção 3 será apresentada a metodologia do trabalho desenvolvido. Na Seção 4 serão apresentadas as simulações computacionais realizadas e os resultados obtidos. As conclusões e os trabalhos futuros serão apresentadas na Seção 5.

2 REFERENCIAL TEÓRICO

2.1 Redes de Sensores sem Fio

As Redes de Sensores sem Fio (RSSF) são uma tecnologia recente que tem crescido muito nos últimos anos. As RSSF são compostas por dispositivos eletrônicos (nós sensores) independentes que possuem capacidade de comunicação sem fio (*wireless*). Os nós sensores possuem capacidade de processamento computacional, comunicação sem fio, sensores que são capazes de realizar medições no ambiente (sismógrafos, termômetros, barômetros, entre outros) e atuadores que podem interagir com o ambiente. Esses nós sensores normalmente são alimentados por baterias, o que torna o consumo energético uma questão chave neste tipo de tecnologia. Esse tipo de tecnologia de sistema embarcado tem sido utilizada como uma forma de computação ubíqua, sendo aplicada em diversas áreas da vida humana, como agricultura, monitoramento, rastreamento e até mesmo para controlar componentes domésticos simples, como luzes, cortinas, eletrodomésticos, entre outros (??). A computação ubíqua é definida como uma forma onipresente de computação, é um conjunto de aparelhos e dispositivos eletrônicos que realizam computação de forma transparente e imperceptível para os usuários (??).

Apesar de cada nó sensor da rede possuir a capacidade de processamento e sensoriamento de forma individual, em aplicações de RSSF um nó sensor não é capaz de atingir o objetivo da rede individualmente. É necessário que os nós sensores cooperem entre si, através de comunicação sem fio, para que consigam alcançar o objetivo da RSSF. Essa característica acarreta em diversas dificuldades no desenvolvimento do software que con-

trolará a operação da RSSF. A flexibilidade e a heterogeneidade das RSSF requerem que o software de controle seja capaz de tratar e abstrair as diferenças e características de cada nó sensor individual. É necessário que este software seja capaz de tratar a RSSF como um todo para que o objetivo seja alcançado.

Devido às estas características especiais das RSSF, diversas pesquisas são realizadas no intuito desenvolver arquiteturas e protocolos que sejam mais adequados às necessidades deste tipo rede. ??) realizaram uma revisão da literatura englobando todas as áreas de pesquisa em RSSF. Diversos protocolos e arquiteturas já desenvolvidas para RSSF são detalhados por ??). Uma revisão mais recente da literatura é realizada por ??), que focam suas análises em protocolos inteligentes para RSSF.

2.2 Simuladores de RSSF

Devido às características das RSSF e por serem sistemas massivamente distribuídos, os experimentos com dispositivos (nós sensores) reais são caros e muito demorados, em alguns casos se tornando impraticáveis (??). Essa dificuldade trouxe a necessidade de se desenvolver simuladores para que protocolos e arquiteturas desenvolvidas em pesquisas pudessem ser previamente testados e avaliados. Essa necessidade existe porque a implantação e construção de arquiteturas em nós sensores reais é difícil e consome muito tempo e recursos. Após essa avaliação preliminar com simuladores, esses protocolos e arquiteturas podem ser implantados e avaliados em RSSF reais, com nós sensores físicos.

Existem vários simuladores de redes de computadores (não apenas RSSF) disponíveis atualmente. Muitos desses simuladores suportam a simu-

lação de redes sem fio e ad hoc, permitindo que sejam usados para simular RSSF. Esses simuladores são complexos e simulam desde o meio físico de comunicação (cabo, ar, entre outros) até as características de software e hardware dos nós da rede, como baterias, rádios de comunicação, sensores e pilhas de protocolos.

Dentre os diversos simuladores de redes disponíveis atualmente, pode-se destacar alguns importantes. O ns-3 (??) é um simulador de redes escrito em C++ e Python muito popular no meio acadêmico. Ele é principalmente utilizado para simulações de redes sem fio, o que contempla o universo das RSSF. O ns-3 é um software gratuito e de código-fonte aberto (*open source*), o que incentiva sua utilização em pesquisas.

Outro simulador de redes de código-fonte aberto é o GNS3 (??), que tem como principal característica a simulação de redes complexas, englobando diversos roteadores, *switches* e vários tipos de dispositivos com capacidade de se conectar à uma rede. O GNS3 possui uma interface gráfica para modelagem e configuração da rede que será simulada, o que facilita estes processos para quem não possui conhecimento de programação de computadores.

Um outro simulador gratuito é o Grubix (??), que é desenvolvido na linguagem de programação Java e é focado em simulações de redes sem fio ad hoc. Este simulador é implementado seguindo uma arquitetura orientada à eventos (*discrete event network simulation*).

Devido ao fato dos simuladores de redes contemplarem quase todas as características de uma rede real, as simulações realizadas com estes softwares demandam muito tempo para serem executadas. No contexto do atual estudo, é necessário que as aplicações geradas pelo *framework* pro-

posto sejam simuladas de forma intensiva (dezenas a centenas de vezes por segundo). Os primeiros experimentos realizados utilizaram o simulador Grubix para avaliar os programas gerados pela PG. Para solucionar um cenário simples com poucos nós sensores, o *framework* necessitava de 3 dias de execução. Portanto é impraticável a utilização de um simulador de redes completo para simular e avaliar as aplicações geradas pela PG. Dessa forma, foi desenvolvido um módulo de simulação simplificado para realizar a avaliação das aplicações geradas.

O módulo simplificado desenvolvido foi utilizado para solucionar o mesmo cenário em que o simulador Grubix foi aplicado, o tempo de execução caiu de 3 dias para 2 minutos. Esse módulo de simulação foi desenvolvido baseado na arquitetura interna do simulador Grubix (??). Essa foi escolha deve-se ao fato de que o Grubix é um simulador específico para simulações de redes sem fio adhoc, o que torna sua arquitetura mais simples e mais adequada às necessidades do *framework* proposto.

2.3 Middlewares

Middlewares são abstrações na camada de aplicação das RSSF que permitem simplificar a programação de uma rede heterogênea, facilitando a gestão de recursos e energia (??). Eles devem prover a coordenação dos nós sensores, através da distribuição de tarefas de baixo nível equivalentes às tarefas de alto nível recebida como entrada, fundir os resultados obtidos pelos nós individualmente e apresentar um *feedback* para o usuário, além de ter mecanismos que promovam a eficiência energética, robustez e escalabilidade (??).

Os middlewares (MW) funcionam entre a camada do Sistema Ope-

racional (SO) dos nós sensores e a camada de aplicação e podem ser classificados considerando duas camadas, horizontal e vertical (??). A camada horizontal define o tipo de MW, ou seja, a sua implementação. Já a camada vertical se preocupa com as propriedades que cada MW possui considerando os ambientes em que estão inseridos.

Existem outras classificações de diversos MWs. O MW proposto se encaixa melhor na categoria de MWs orientados à eventos. O paradigma orientado a eventos é um dos mais recentes tipos de MW existente, embora seja muito promissor há poucos estudos nesta área (??). O MW mais notável na categoria de MWs orientados à eventos é o *Impala* (??). O *Impala* foi criado visando dar suporte à um projeto maior, denominado ZebraNet, que visa o rastreamento de animais e mapeamento do meio-ambiente. O *Impala* possui algumas características similares ao Odin, como a implantação sem fio de novas aplicações, gerenciamento das características específicas de hardware e do rádio de comunicação, entre outras. A principal diferença é a linguagem de programação em *scripts* provida pelo Odin. Essa linguagem foi desenvolvida e especializada para ser utilizada pela PG, visando aspectos específicos da geração automatizada de código-fonte.

2.4 Algoritmos Genéticos

O Algoritmo Genético (ou *Genetic Algorithm*) é um algoritmo de computação evolutiva inicialmente proposto por ??). Essa metaheurística baseia-se principalmente no processo de seleção natural. O conceito de seleção natural foi proposto por Charles Darwin em 1859 e consiste na ideia de que as espécies com características mais adaptadas ao meio onde vivem tendem a sobreviver, enquanto as espécies pouco adaptadas tendem a ser

extintas. O mesmo raciocínio ocorre para as características genéticas de uma espécie. Uma característica bem adaptada ao meio, se mantém nas gerações seguintes, já as características pouco adaptadas ao meio, tendem a desaparecer.

Darwin acreditava que o meio ambiente seleciona naturalmente os seres que nele vivem. De forma resumida, os seres que são adaptados ao meio em que vivem sobrevivem. Já os seres que não se adaptam morrem e são extintos, assim como características genéticas de uma espécie. Esse conceito de seleção natural leva em conta o conceito de hereditariedade, que é a capacidade dos seres vivos de transmitirem características genéticas aos seus descendentes. Isso ocorre, por exemplo, durante a meiose das células humanas para geração dos gametas (espermatozóide e óvulo). Durante o processo de meiose pode ocorrer um fenômeno chamado *crossing-over*. Este fenômeno ocorre quando um par cromossomos de DNA trocam trechos de sua codificação genética. Esse fenômeno pode trazer variabilidade nas características provenientes do progenitor. Além disso, podem ocorrer mutações aleatórias na codificação genéticas dos gametas produzidos na meiose.

Todos esses conceitos são levados em conta no Algoritmo Genético. Inicialmente define-se uma população de indivíduos. Cada indivíduo representa uma solução do problema em questão. Essa população tem seus indivíduos inicializados de forma randômica (ou através de alguma rotina específica para o problema tratado). Em seguida, executa-se o chamado laço de evolução, que “evolui” os indivíduos durante um certo número de gerações. Primeiramente é executada uma rotina de seleção, que seleciona dois indivíduos da população para se realizar o cruzamento (ou recombinação, *crossover*). Um novo indivíduo é gerado nessa recombinação, em seguida

ele é processado por uma rotina que realiza mutações em sua codificação. Esse procedimento de seleção, cruzamento e mutação é executado até que a população da próxima geração seja totalmente gerada.

Para avaliar a aptidão dos indivíduos (qualidade das soluções), o AG utiliza uma função de avaliação (ou função de *fitness*). Essa função de avaliação é uma rotina que analisa a codificação de um indivíduo do AG e calcula um valor que indica a qualidade da solução representada por aquele indivíduo. A função de *fitness* leva em conta características do problema para calcular a aptidão do indivíduo, dessa forma são necessárias diferentes funções de *fitness* para problemas distintos.

Existem diversas variações de AG, ??) propõe um AG que possui múltiplas populações que evoluem independentes. Cada população organiza seus indivíduos em uma estrutura hierárquica baseada em árvores. Periodicamente é executada uma rotina de migração, que copia os melhores indivíduos das populações para a população vizinha. Foram realizados experimentos com este tipo de AG, porém os resultados do AG simples se mostraram mais promissores durante o desenvolvimento do estudo.

2.5 Programação Genética

Programação Genética (PG) é um conjunto de técnicas de computação evolutiva que visa realizar a geração de programas de computador (??). Segundo ??), a programação genética utiliza algoritmos evolutivos para evoluir estruturas (indivíduos) que são programas de computador. A função de avaliação (*fitness*) em programação genética consiste em uma medida relacionada à execução desses programas. Segundo os mesmos autores, trata-se de uma busca evolutiva em um espaço possível de programas de computador

que sejam capazes de produzir os melhores resultados quando executados. Os programas gerado pela PG podem ser programas independentes ou parte de programas maiores.

A PG é utilizada em diversos campos de aplicação, sendo explorada para gerar programas visando solucionar diversos problemas. ??) apresentam uma revisão das pesquisas realizadas nas áreas de PG e máquinas evolutivas em um período de dez anos. São apresentados livros sobre o assunto e mostradas algumas estatísticas acerca das pesquisas em PG e máquinas evolutivas. Já ??) realizam uma revisão bibliográfica acerca de aplicações de PG baseada em gramáticas. O artigo se concentra em mostrar como as gramáticas formais contribuem na solução dos problemas, além de realizar uma análise das prováveis tendências da pesquisa nessa área.

3 METODOLOGIA

4 RESULTADOS E DISCUSSÃO

5 CONCLUSÃO E TRABALHOS FUTUROS