

LISTA DE FIGURAS

2.1	Sistema hipermídia representado como um grafo	3
2.2	Formigas em busca de alimento encontram o menor caminho entre alimento e ninho	4
2.3	As regiões de uma cidade com pontos de agências bancárias	6
3.1	Funcionamento do sistema em camadas	9
5.1	Quantidade de trabalho aplicada em cada workflow nas fases do UP	12

LISTA DE TABELAS

6.1	Descrição dos riscos do sistema	18
7.1	Descrição dos limites do sistema	19
9.1	Cronograma das atividades desenvolvidas	24
9.2	Cronograma da metodologia de desenvolvimento	24

LISTA DE ABREVIATURAS E SIGLAS

ACO	<i>Ant Colony Optimization</i>
Ajax	<i>Asynchronous JavaScript and XML</i>
CSS	<i>Cascading Style Sheets</i>
DOM	<i>Document Object Model</i>
FTP	<i>File Transfer Protocol</i>
HA	Hipermídia Adaptativa
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
IMAP	<i>Internet Message Access Protocol</i>
LDAP	<i>Lightweight Directory Access Protocol</i>
MR	Modelo Relacional
NA	Navegação Adaptativa
PDF	<i>Portable Document Format</i>
PDSU	Processo de Desenvolvimento de Softwares Unificado
PEAR	<i>PHP Extension and Application Repository</i>
PHP	<i>Hypertext PreProcessor</i>
PO	Probabilidade de Ocorrência
POP3	<i>Post Office Protocol</i>
UML	<i>Unified Modeling Language</i>
UP	<i>Unified Process</i>
URL	<i>Universal Resource Locator</i>
XML	<i>Extensible Markup Language</i>

SUMÁRIO

LISTA DE FIGURAS.....	i
LISTA DE TABELAS.....	ii
LISTA DE ABREVIATURAS E SIGLAS.....	iii
1 INTRODUÇÃO.....	1
2 REFERENCIAL TEÓRICO.....	3
2.1 TÉCNICAS DE NAVEGAÇÃO ADAPTATIVA	4
2.2 TRABALHOS RELACIONADOS.....	5
2.2.1 AntWeb	5
2.2.2 Tendências espaciais	6
2.2.3 <i>Data mining</i> com otimização por colônia de formigas	6
3 OBJETIVO GERAL	8
3.1 OBJETIVOS ESPECÍFICOS	8
3.1.1 Módulo: Computação de Feromônio.....	8
3.1.2 Módulo: Adaptação.....	8
3.1.3 Módulo: Administração.....	9
4 JUSTIFICATIVA	11
5 METODOLOGIA.....	12
5.1 TECNOLOGIA.....	14
5.1.1 PHP	14
5.1.2 CSS.....	15

5.1.3	JavaScript	15
5.1.4	Ajax	16
5.1.5	UML	16
5.1.6	PEAR.....	16
6	ANÁLISE DE RISCOS.....	18
7	LIMITES DO SISTEMA	19
8	RECURSOS COMPUTACIONAIS	20
8.1	RECURSOS DISPONÍVEIS PARA DESENVOLVIMENTO	20
8.1.1	Recursos de Hardware	20
8.1.2	Recursos de Software	20
8.2	RECURSOS NECESSÁRIOS PARA UTILIZAÇÃO	21
8.2.1	Usuário	21
8.2.2	<i>Website</i> Hospedeiro	21
9	CRONOGRAMAS.....	23
9.1	ATIVIDADES	23
9.2	DISCIPLINAS	23
9.3	ITERAÇÕES	23
	REFERÊNCIAS.....	25

1 INTRODUÇÃO

A Internet representa atualmente um dos maiores meios de comunicação global, por meio da qual milhares de pessoas interagem entre si, seja pela escrita, voz ou vídeo.

Dentre os meios de comunicação disponíveis na Internet, os *websites* são responsáveis por grande parte das informações trafegadas na rede. Um site é um conjunto de páginas, isto é, de hipertextos, sistema para a visualização de informação cujos documentos contém referências internas para outros documentos (chamadas de *hyperlinks* ou, simplesmente, *links*), acessíveis geralmente pelo protocolo *HyperText Transfer Protocol* (HTTP) na Internet. (WONG, 2000)

As páginas num *website* são encontradas a partir de uma *Universal Resource Locator* (URL) inicial (página principal) e geralmente residem no mesmo diretório de um servidor. (WONG, 2000)

A maioria dos websites utiliza recursos da hipermídia, documento hipertexto que incorpora textos, gráficos, sons, imagens e animações, para apresentar as informações aos usuários.

Entretanto, nem sempre as ligações (*links*) entre os documentos estão organizadas de modo que o usuário consiga facilmente atingir seu objetivo. Isso acontece devido a quantidade de páginas e sub-páginas inseridas para representar partes do mesmo contexto, ou ainda pelo fato que maioria das páginas *Web* utiliza mecanismos de representação embutidos em linguagem *HyperText Markup Language* (HTML), tornando a definição da estrutura da página *Web* bastante limitada e dificultando sua análise semântica. (VIEIRA; CAIXETA, 2004)

Apesar dos vários mecanismos de busca existentes, como Google (<<http://www.google.com>>), Yahoo (<www.yahoo.com.br>) entre outros, ainda se percebe a carência de uma mecanismo que oriente o usuário sugerindo ou mostrando qual é o próximo passo para encontrar seu objetivo.

Os termos orientação e navegação sugerem a concepção de hiperdocumentos como espaços de informação onde os leitores podem mover-se de uma parte para outra. Essa idéia certamente contribui para caracterizar o mais citado problema dos leitores de hipertexto, a "desorientação", que ocorre quando os leitores não sabem onde estão, como chegaram lá, ou para onde deveriam ir. (PANSANATO, 2003)

Hipermídia Adaptativa (HA) é a área da ciência da computação que se ocupa do estudo e desenvolvimento de sistemas, arquiteturas, métodos e técnicas capazes de promover a adaptação de hipermídia em geral às expectativas, necessidades, preferências e desejos de seus usuários. (PALAZZO, 2002)

A fim de suprir essas necessidades, a HA, através da navegação adaptativa, procura gerar as ligações (*links*) entre os assuntos do site de forma que o usuário consiga minimizar sua desorientação, formando assim uma rota até seu objetivo. A presente proposta está inserida neste contexto.

No Capítulo 2 estão apresentados os principais fundamentos teóricos, no Capítulo 3 estão apresentados o objetivo geral e os objetivos específicos, no Capítulo 4 estão relacionados os principais motivos pelo qual o sistema deverá ser elaborado, no Capítulo 5 está descrita a metodologia de desenvolvimento do projeto e as tecnologias utilizadas, no Capítulo 6 estão apresentados os possíveis riscos apresentados pelo sistema, no Capítulo 7 estão apresentados os principais tópicos que objetivam o sistema, no Capítulo 8 estão descritos os recursos computacionais disponíveis para desenvolvimento e os recursos computacionais necessários para utilização do sistema e finalmente no Capítulo 9 estão estipulados os cronogramas para desenvolvimento do projeto.

2 REFERENCIAL TEÓRICO

Segundo Vieira e Caixeta (2004), um sistema de hipermídia é um grafo e nele existem vários caminhos possíveis para se chegar à um nodo partindo de outro. Alguns caminhos são menores que outros. A indicação preferencial desses caminhos faz com que os usuários cheguem ao seu objetivo de forma mais rápida.

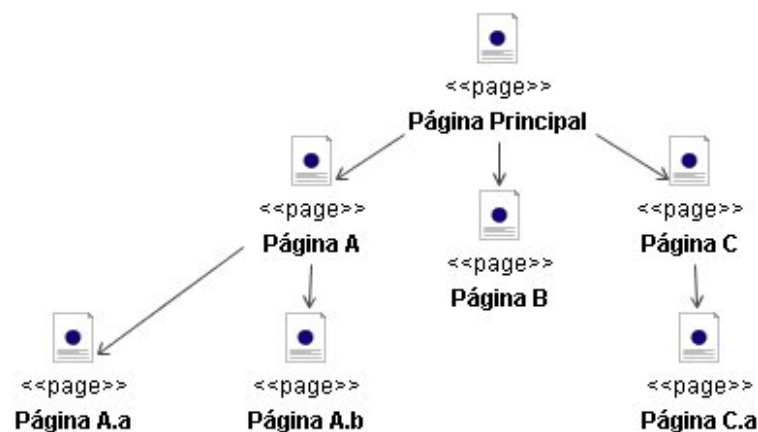


Figura 2.1: Sistema hipermídia representado como um grafo

Observa-se na figura 2.1 a estrutura exemplificado de um diagrama navegacional de um *web-site*, nota-se grande semelhança com a estrutura de um grafo.

Partindo desse princípio, pode-se fazer uma analogia com uma colônia de formigas reais, visto que cada usuário representa uma formiga, o grafo gerado pelas páginas corresponde aos diferentes caminhos do formigueiro, o ponto inicial do usuário (página principal) corresponde ao ninho (ponto de partida das formigas) e o objetivo do usuário corresponde a comida que deve ser encontrada. Ainda pode-se afirmar que como a formiga, o usuário também tem uma visão de curto alcance, ou seja consegue visualizar apenas o próximo passo. (TELES et al., 2003)

Apesar de tanta semelhança, o que causa o sucesso das colônias de formiga em alcançar o objetivo de encontrar o alimento por uma forma cooperativa de trabalho, ainda não ocorre na Internet, pois os usuários não interagem para colaborar entre si na navegação.

As formigas reais se comunicam indiretamente por uma substância chamada feromônio. Essa substância é exalada espontaneamente por todos os membros da colônia. Uma formiga ao se deparar com dois caminhos diferentes faz uma escolha probabilística baseada na quantidade de feromônio presente nestes caminhos. Logo, se há maior concentração de feromônio em determinado

caminho, será esse o caminho escolhido. (LOPES, 2006)

O feromônio evapora com o passar do tempo, assim sendo, um caminho que não é constantemente visitado tende a desaparecer do caminho oficial tomado pelas formigas.

No início do processo, quando um grupo de formigas está em busca de alimento e ainda não há um caminho traçado, o grupo que optar casualmente pelo menor caminho irá chegar ao alimento antes do grupo que optar pelo caminho maior. Desta forma, vão acentuar o menor caminho com feromônio na volta para o ninho. É questão de tempo até que todos os membros da colônia estejam trafegando pelo caminho mais curto. (DORIGO et al., 1996)

A busca por alimento pelas formigas reais é ilustrada na figura 2.2. A figura 2.2-A mostra o caminho traçado entre o ninho e o alimento. A figura 2.2-B mostra o que acontece logo após a inserção de um obstáculo no caminho. O inseto tende a seguir cada uma das direções (esquerda ou direita) com probabilidade igual de 50%. Observa-se a ocorrência análoga no sentido inverso 2.2-C. Com tempo, o feromônio acumula-se mais rapidamente no caminho mais curto ao redor do obstáculo, tornando assim o menor caminho a ligação entre o ninho e o alimento. (LOPES, 2006)

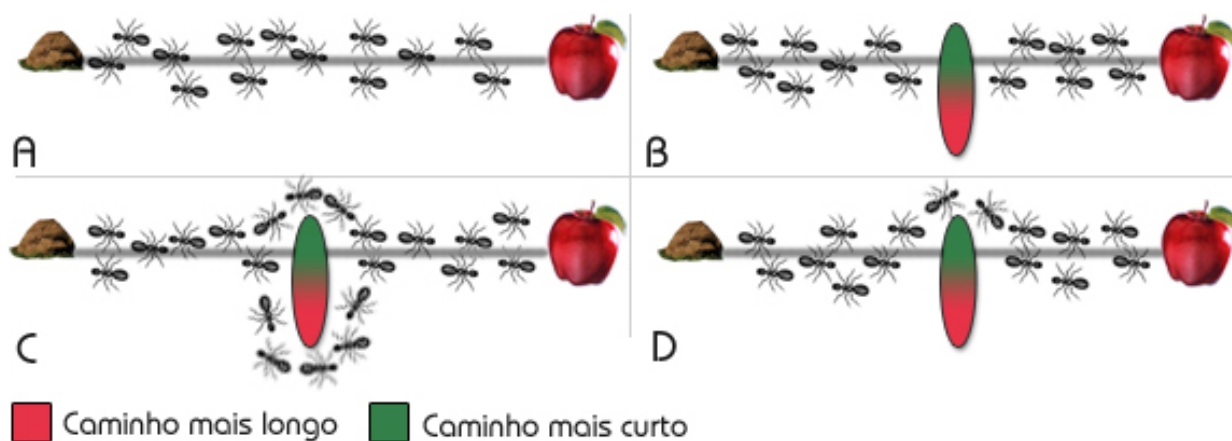


Figura 2.2: Formigas em busca de alimento encontram o menor caminho entre alimento e ninho

Através desta engenhosidade, que no âmbito computacional ficou conhecida como *Ant Colony Optimization* (ACO) muitos problemas puderam ser solucionados como é o caso do problema do caixeiro viajante e roteamento de pacotes. (VIEIRA; CAIXETA, 2004)

2.1 TÉCNICAS DE NAVEGAÇÃO ADAPTATIVA

Segundo Palazzo (2002) as técnicas de Navegação Adaptativa (NA) são os recursos usados pelos sistemas de HA para informar ou influenciar o usuário no seu processo de navegação.

As principais técnicas de NA são:

- **Orientação Direta:** Decide em cada ponto da navegação qual o melhor nodo a ser visitado a seguir. Como desvantagem, essa técnica não oferece suporte no caso do usuário não desejar seguir a sugestão do sistema.
- **Classificação Adaptativa:** Classifica os links de um nodo de acordo com sua relevância.
- **Ocultação Adaptativa:** Consiste em restringir o espaço de navegação, ocultando os links para nodos não-relevantes.
- **Anotação Adaptativa:** Consiste em aumentar a informação existente nos links com alguma forma de anotação ou comentário que podem dizer mais sobre o estado corrente dos nodos a que se conectam.
- **Geração de Links:** Consiste em criar links adicionais automaticamente nos nodos. Uma das desvantagens desta técnica é o aumento no número de links existentes nos nodos, aumentando a sobrecarga cognitiva e tornando o hiperespaço mais complexo.
- **Mapas Adaptativos:** Têm o objetivo de ajudar o usuário a entender a estrutura do hiperespaço.

Para a presente proposta, as técnicas de NA empregadas serão: Orientação Direta e Classificação Adaptativa.

2.2 TRABALHOS RELACIONADOS

2.2.1 AntWeb

O trabalho proposto está relacionado a um trabalho já existente chamado AntWeb (TELES et al., 2003; VIEIRA; CAIXETA, 2004)

O AntWeb é um sistema que proporciona aos usuários de um determinado site uma navegação cooperativa, também baseada no comportamento das formigas reais em busca de alimento. O sistema orienta os visitantes com interesses similares às suas páginas de destino.

Um protótipo do AntWeb foi implantado em uma versão do portal Interlegis (Portal da Comunidade Legislativa Brasileira <<http://www.interlegis.gov.br>>). O portal serviu apenas como um experimento e o sistema AntWeb não foi efetivamente implantado no site oficial.

O AntWeb foi testado apenas com simulações computacionais, que indicaram boa performance, entretanto não é possível prever o comportamento do sistema com usuários reais. (VIEIRA; CAIXETA, 2004)

Apesar dos testes realizados, e do bom desempenho obtido, fica claro que o AntWeb ainda pode ser melhorado, pois é preciso determinar com precisão as categorias de usuários para que o sistema possa recuperar as páginas relevantes para os usuários de mesma categoria (VIEIRA; CAIXETA, 2004)

Nos testes do sistema AntWeb, foram gerados dados para contornar o fato destas funcionalidades não estarem implementadas.

A presente proposta também servirá de parâmetro para trabalhos futuros, em que pretende-se coletar resultados e informar se o sistema realmente funcionará com usuários reais.

2.2.2 Tendências espaciais

Com a otimização por colônia de formigas é possível traçar uma linha que representa qual será a tendência, por exemplo, de crescimento de uma população. Ainda tomando uma população como exemplo, com o estudo de tendências espaciais é possível fazer uma previsão de crescimento que irá indicar para que lugar a população tende a se alastrar. (ZARNANI; RAHGOZAR, 2006).

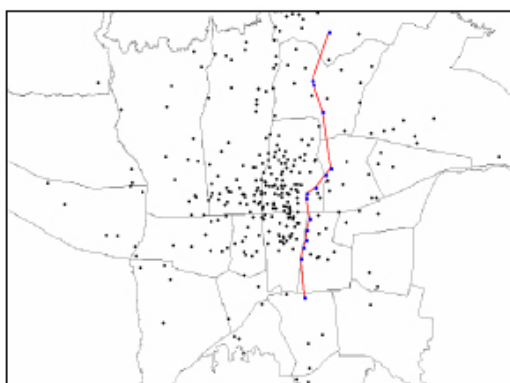


Figura 2.3: As regiões de uma cidade com pontos de agências bancárias

Fonte: (ZARNANI; RAHGOZAR, 2006)

Na figura 2.3 é traçada a tendência de expansão das agências bancárias de uma determinada cidade.

2.2.3 Data mining com otimização por colônia de formigas

O trabalho de Parpinelli et al. (2002) propõe um algoritmo para a mineração de dados denominado *Ant-mining*. O objetivo do algoritmo é extrair regras da classificação de dados. O algoritmo é inspirado na pesquisa sobre o comportamento das colônias de formigas reais e reúne conceitos de mineração de dados. O objetivo do algoritmo é gerar uma lista de regras que consegue uma solução sub-ótima para a resolução de um determinado problema. Num comparativo com outra técnica de

mineração o *Ant-mining* se mostrou mais eficiente e a lista de regras descobertas foi consideravelmente mais simples (menor) do que aquelas descobertas com outra técnica. (PARPINELLI et al., 2002).

3 OBJETIVO GERAL

Desenvolver um mecanismo de navegação adaptativa e cooperativa, com base na sugestão de links selecionados a partir da meta-heurística ACO e implementa-lo em um protótipo.

3.1 OBJETIVOS ESPECÍFICOS

O sistema será dividido em módulos descritos a seguir.

3.1.1 Módulo: Computação de Feromônio

O módulo de computação do feromônio, deve implementar de fato a meta-heurística de ACO. Esse módulo é acionado toda vez que um usuário trafega pelas páginas do *website*.

A cada link acessado, antes do redirecionamento propriamente dito figura 3.1-3, uma camada (camada Ant) é invocada para computar a origem e destino de determinado usuário, atualizando a taxa de feromônio na aresta percorrida figura 3.1-2.

Os algoritmos de ACO serão desenvolvidos em forma de componentes, para que possam ser reutilizados em outros projetos.

3.1.2 Módulo: Adaptação

Antes da devolução da página em HTML, a camada de sugestão de links recolhe as informações relevantes para o usuário solicitante, conforme a figura 3.1-4. Finalmente, a página é retornada com as sugestões relevantes para o usuário.

Os visitantes poderão optar pela utilização do sistema. Ao se escolher por navegar com o sistema habilitado, uma camada (em HTML *iframe*) é adicionada na parte superior do *website* e nela serão efetivamente adicionadas as sugestões de links, caso contrário o site não sofrerá qualquer alteração, porém os links percorridos pelos usuários não serão computados.

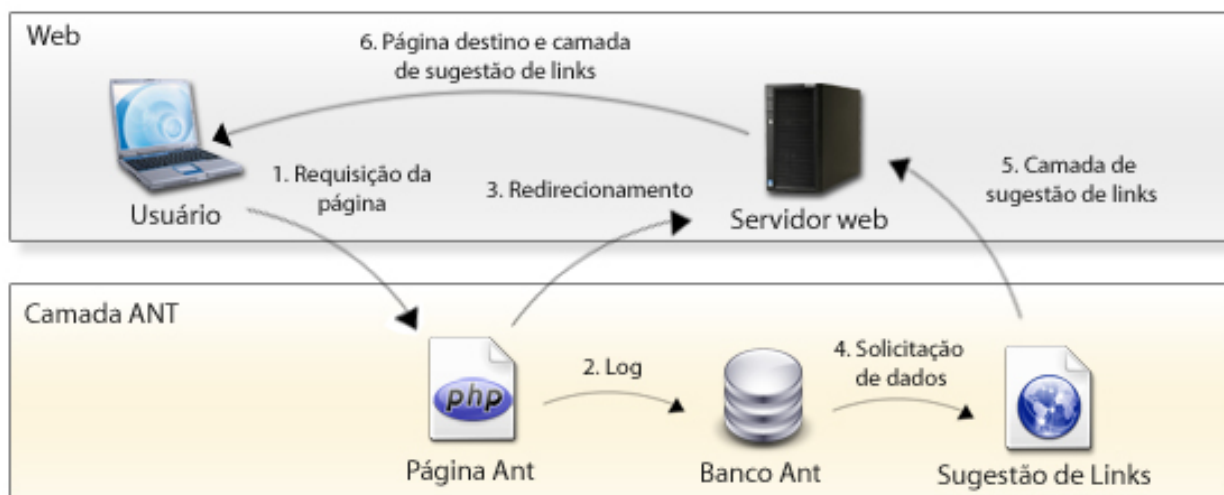


Figura 3.1: Funcionamento do sistema em camadas

3.1.3 Módulo: Administração

É imprescindível que o usuário se auto-classifique em um dos grupos pré-determinados pelo administrador do *website*, pois a relevância dos conteúdos será baseada no histórico de navegação dos usuários de mesmo grupo.

O sistema poderá ser ajustado pelo administrador do *website* hospedeiro, através do qual será possível a personalização de:

- **Grupos de usuários** - Esses grupos deverão ser cadastrados de acordo com o contexto do *website* hospedeiro, sendo possível adicioná-los ou removê-los. Os grupos devem ser bem definidos, para que o sistema tenha um bom aproveitamento.
- **Cores do sistema** - Para de tornar o sistema adaptável ao estilo do *website* hospedeiro, alguns estilos pré-definidos estarão disponíveis. Para usuários avançados, com objetivo de uma personalização específica, a folha de estilos *Cascading Style Sheets* (CSS) ficará disponível para alterações.
- **Cores dos links** - As cores dos links sugeridos pelo sistema poderão ser configuradas de modo que um link mais relevante fique mais destacado que outro.
- **Exibição das sugestões** - Será possível alterar a disposição das sugestões:
 - **Disposição objetiva:** Mostrará qual é a página alvo. Essa é a página que em média foi mais visitada pelos outros elementos de um mesmo grupo. Essa disposição implementa a técnica de orientação direta de Palazzo (2002).

- **Disposição orientada:** Mostrará qual o caminho que se deve seguir para chegar até a página alvo. Esse caminho é traçado com base no histórico dos caminhos percorridos de usuários do mesmo grupo.
- **Disposição por assuntos relacionados:** Mostrará quais os links mais visitados pelos usuários de seu grupo. Além da página alvo outras páginas também podem interessar o usuário de determinado grupo, assim ao selecionar a exibição por assuntos relacionados, obtém-se uma lista das páginas mais relevantes para determinado.
- **Configurações do sistema** - Determinadas configurações de característica técnica, para administradores avançados, também poderão ser configuradas. Estas características são constantes matemáticas utilizadas nas equações para o cálculo de atualização de feromônio ou sugestão de links, por exemplo.

A equação 3.1 (VIEIRA; CAIXETA, 2004) mostra como é feito o cálculo da probabilidade na qual uma formiga escolhe ir da página i para a página $j \in N_i$ na iteração p .

$$p_{ij}^d(p) = \frac{[\tau_j^d(p)]^\alpha \cdot [\eta_j(p)]^\beta}{\sum_{l \in N_i} [\tau_l^d(p)]^\alpha \cdot [\eta_l(p)]^\beta} \forall j \in N_i \quad (3.1)$$

Na qual $\tau_j^d(p)$ é a quantidade de feromônio na página j na iteração p para o destino d ; N_i é o conjunto dos vizinhos da página i . α e β são parâmetros que permitem controlar respectivamente os pesos relativos a trilha de feromônio e o valor da heurística. $\eta_j = 1/\omega t_j$ é uma heurística relacionada ao custo de usar a página j , onde:

$$\omega t_j = lt_j + vt_j \quad (3.2)$$

Onde ωt_j é o tempo estimado na página j , sendo lt_j o tempo estimado para obter a página j através do browser, e vt_j o tempo que o usuário levará para visitar a página.

4 JUSTIFICATIVA

A Internet é um meio de comunicação em constante desenvolvimento, contém diversos tipos de conteúdos e abriga diferentes tipos de usuários. Para usuários experientes, a busca por informações na rede é intuitiva, pois isso faz parte de seu cotidiano. Entretanto nem todos usuários têm a mesma facilidade para encontrar rapidamente o que estão procurando. Isso ocorre devido a desorientação oferecida por *websites* que não contemplam algumas regras básicas de acessibilidade/usabilidade.

Afim de suprir esta desorientação, sistemas de busca conseguem filtrar uma grande quantidade de informação objetivando parte da busca do usuário, mas não conseguem traçar com precisão o caminho que deve ser percorrido até seu objetivo. Ainda nota-se uma carência de alguma forma de assistência navegacional que oriente o usuário até seu objetivo.

Percebe-se que mesmo com os mecanismos de busca, alguns usuários ainda têm dificuldades de encontrar páginas de determinado assunto. Isto ocorre pois, o que é oferecido como resultado se resume em uma lista de *websites* que contém palavras iguais à(s) palavra(s)-chave(s) procurada(s). Apesar da grande ajuda oferecida, uma busca menos detalhada pode trazer resultados irrelevantes para o usuário.

Nota-se a carência de algum mecanismo que oriente o usuário dentro um de determinado *website*, pois fica mais fácil se localizar em um site relevante que procurar por vários sites com conteúdos incoerentes à pesquisa. Outra vantagem do sistema de orientação é que ao se navegar até o objetivo há a possibilidade de se descobrir outros assuntos relevantes que podem estar relacionados ao assunto procurado.

Para minimizar a desorientação, deve-se fornecer recursos para permitir aos leitores a identificação de sua posição corrente em relação à estrutura global, reconstruir o caminho que o levou a esta posição, e distinguir entre as diferentes opções para mover-se a partir desta posição. Por exemplo, a manutenção do histórico de navegação (isto é, o caminho percorrido pelo usuário) auxilia o leitor a reconstruir o caminho até a sua posição atual. (PANSANATO, 2003)

O trabalho proposto pretende criar um mecanismo que, independentemente do conteúdo ou diagramação do *website* hospedeiro, possa oferecer um mecanismo de orientação ajudando os usuários a traçar o caminho até seu objetivo.

5 METODOLOGIA

Neste sistema, o processo de desenvolvimento adotado é baseado no Processo de Desenvolvimento de Softwares Unificado (PDSU), denotado nesse trabalho como *Unified Process* (UP). Esse processo é usado em conjunto com a *Unified Modeling Language* (UML) e o paradigma de orientação a objetos.

A figura 5.1 representa graficamente as fases (concepção, elaboração, construção e transição), tarefas do processo (requisitos, análise, projeto, implantação e testes), e a quantidade de trabalho estimado para tarefas específicas em cada fase.

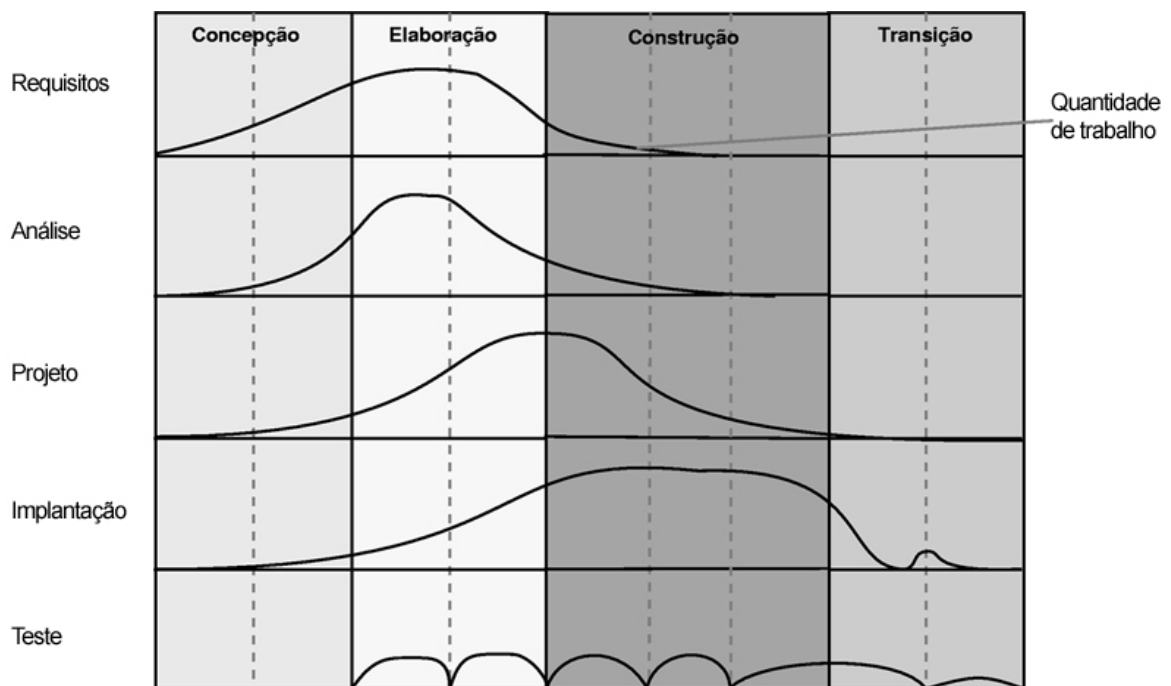


Figura 5.1: Quantidade de trabalho aplicada em cada workflow nas fases do UP

Fonte: (ARLOW, 2002)

O UP é iterativo, incremental, dirigido por casos de uso e dividido em quatro fases: concepção, elaboração, construção e transição (SCOTT, 2003):

- **Concepção:** Estabelece a viabilidade e escopo do sistema, define e estrutura uma arquitetura, identifica os riscos, esclarece questões de custo-benefício. (SCOTT, 2003)
- **Elaboração:** A elaboração deve coletar os requisitos, visando recolher o máximo de informação útil e solidificar a arquitetura. Esta fase visa desenvolver a maioria dos casos de uso e especificar as metas de qualidade (SCOTT, 2003).

- **Construção:** Desenvolve uma versão operacional do sistema que possa ser entregue a usuários beta. Nesta etapa constrói-se maior parte dos modelos de projeto, análise, instalação, implantação e teste (SCOTT, 2003).
- **Transição:** Fase final do projeto, quando é realizado a entrega do sistema aos usuários beta com todas as funcionalidades finalizadas (e futuramente a todos os usuários). A divulgação do software, e preparação dos usuários também fazem parte dessa fase (SCOTT, 2003).

Cada uma dessas fases é dividida em sub-processos chamados fluxos de trabalho ou *workflows*. Cada fluxo de trabalho tem um tema técnico específico, enquanto as fases constituem divisões gerenciais, caracterizadas por atingirem metas bem definidas. (FILHO, 2003) Podem acontecer uma ou mais iterações por fase do processo, cada iteração gerará uma gama de artefatos específicos dos fluxo de trabalho.

É proposto, com o uso desta metodologia e base nos workflows citados, gerar os seguintes artefatos para fins de documentação:

- **Requisitos:** Fluxo que visa a obter um conjunto de requisitos de um produto, aprovado pelo cliente e desenvolvedor (FILHO, 2003).
 - Levantamento dos requisitos funcionais e não-funcionais;
 - Levantamento dos atores e casos de uso;
 - Levantamento dos riscos;
 - Especificação dos casos de uso;
 - Diagrama de caso de uso;
- **Análise:** Fluxo cujo objetivo é detalhar, estruturar, e validar os requisitos, de forma que esses possam ser usados como base para o planejamento detalhado (FILHO, 2003).
 - Diagramas de classes de análise;
 - Refinamento dos casos de uso;
 - Especificação de casos de uso;
 - Diagrama de sequência;
- **Projeto:** Fluxo cujo objetivo é formular um modelo estrutural do produto que sirva de base para a implementação (FILHO, 2003).
 - Diagrama de classe de projeto;

- Modelo Relacional (MR);
- Diagrama de pacotes;
- Manual do usuário;
- Implementação: Fluxo cujo objetivo é realizar o projeto em termos de componentes de código (FILHO, 2003).
 - Codificação do sistema;
 - Implementação do MR;
 - Diagrama de implantação;
 - Diagrama de componentes;
- Teste: Fluxo cujo objetivo é verificar os resultados da implementação (FILHO, 2003).
 - Gerar casos de teste de todas as funcionalidades do sistema;
 - Documento de teste de caixa preta;

5.1 TECNOLOGIA

O trabalho será implementado utilizando as seguintes tecnologias:

5.1.1 PHP

Hypertext PreProcessor (PHP) é uma linguagem de script para uso em servidores. O servidor faz a codificação dos scripts PHP, retornando um resultado em HTML para o navegador. Este tipo de interação permite algumas operações complexas, possibilitando assim a elaboração de páginas dinâmicas. (SILVA et al., 2003)

A escolha da linguagem está fortemente relacionada com o poder que a linguagem oferece e o seu crescimento. A linguagem PHP vem crescendo exponencialmente nos últimos anos. No início de 1999, pouco mais de 50 mil sites utilizavam a linguagem PHP, enquanto que no início de 2003 esse número ultrapassava 10 milhões. (NIEDERAUER, 2004)

Segundo Couto (2004) são vantagens do PHP:

1. Linguagem de fácil aprendizado;
2. Performance e estabilidade excelentes;

3. Código-aberto (não necessitando licença de uso e podendo alterá-lo conforme as necessidades);
4. Suporte nos principais servidores de páginas existentes;
5. Suporte aos principais banco de dados existentes;
6. Multipaltaforma e portátil;
7. Suporte a uma grande variedade de padrões e protocolos, como *Extensible Markup Language* (XML), *Document Object Model* (DOM), *Internet Message Access Protocol* (IMAP), *Post Office Protocol* (POP3), *Lightweight Directory Access Protocol* (LDAP), HTTP, *File Transfer Protocol* (FTP), entre outros.

No projeto, a linguagem PHP (em sua versão 5) será utilizada por implementar os paradigmas da orientação a objetos, tornando possível a implementação efetiva das classes projetadas.

5.1.2 CSS

Cascading Style Sheets (CSS) são folhas estilos para páginas web e possibilitam a mudança da aparência simultânea de todas as páginas relacionadas com o mesmo estilo.

Ao invés de colocar a formatação dentro do código, o programador cria um *link* (ligação) para uma página que contém os estilos, procedendo de forma idêntica para todas as páginas de um portal.

Quando quiser alterar a aparência do portal basta portanto modificar apenas um arquivo. (AMARAL, 2001)

As folhas de estilo CSS serão utilizadas para a customização da camada de apoio à navegação, a fim de tornar a camada semelhante ao estilo do *website* hospedeiro.

5.1.3 JavaScript

JavaScript é uma linguagem de programação que proporciona maior interatividade com os usuários. Os programas feitos em JavaScript permitem que praticamente todas as aplicações sejam manipuladas no browser do usuário, desde validação de formulários, apresentação de novas janelas, manipulação de imagens, criação de camadas, cálculos complexos, e muitas outras ações que podem tornar as páginas extremamente interativas (SOARES, 2000).

A linguagem JavaScript será utilizada para validações de campos e nos componentes feitos em *Asynchronous JavaScript and XML* (Ajax).

5.1.4 Ajax

Ajax é o uso de Javascript e XML (e derivados) para tornar o navegador mais interativo com o usuário, utilizando-se de solicitações assíncronas de informações. Ajax não é uma tecnologia, são realmente várias tecnologias trabalhando em conjunto, cada uma fazendo sua parte, oferecendo novas funcionalidades.

Uma de suas principais funcionalidades é poder processar informações sem a necessidade de recarregar o navegador, se assemelhando às aplicações desktop. Embora muito útil, o Ajax usado em demasia pode tornar uma aplicação mais lenta que o necessário.

No projeto, o Ajax será aplicado uma camada de trabalho (*framework*) denominada sAjax. Disponível em: <<http://www.modernmethod.com/sajax/>>.

5.1.5 UML

A UML é uma linguagem gráfica para visualização, especificação, construção e documentação de artefatos de sistemas complexos de software. A UML proporciona uma forma-padrão para a preparação de planos de arquitetura de projetos de sistemas, incluindo aspectos conceituais tais como processos de negócios e funções do sistema, além de itens concretos como as classes escritas em determinada linguagem de programação, esquemas de bancos de dados e componentes de software reutilizáveis. (BOOCH et al., 2000)

No projeto, a linguagem UML servirá para a documentação dos artefatos descritos no capítulo 5 dessa proposta.

5.1.6 PEAR

PHP Extension and Application Repository (PEAR) é um framework e um sistema de distribuição de componentes reutilizáveis PHP. O PEAR oferece um conjunto amplo de bibliotecas estruturadas de código aberto para usuários, onde cada biblioteca é considerada um projeto separado com seu próprio grupo de desenvolvimento, número de versões, ciclo de *release* e documentação. (NADALETE, 2005)

Dentre os principais componentes deverão ser utilizados:

- **PEAR::DB** : Pacote responsável pela camada de abstração do banco de dados.
- **PEAR::HTML_Template_IT** : Pacote que fornece uma *engine*, possibilitando trabalhar com blocos e substituição de *tags*. Com o uso deste pacote é possível separar a apresentação

do sistema (HTML, CSS e Javascript) da lógica de apresentação (PHP).

Os componentes PEAR, além de serem desenvolvidos de forma genérica, podendo ser aplicados no desenvolvimento de diferentes projetos, possibilitam a modificação e adaptação dos mesmos, conforme necessidade do projeto em desenvolvimento.

Os componentes PEAR podem ser encontrados no *website* <<http://pear.php.net>>

6 ANÁLISE DE RISCOS

Tabela 6.1: Descrição dos riscos do sistema

ID	Risco	Categoria	PO ^a
R1	O sistema poderá tornar um site dinâmico mais lento que um padrão aceitável	Técnico	Média
R2	O sistema poderá isolar certas áreas do site, criando " áreas mortas "	Técnico	Baixa
R3	Os algoritmos genéticos podem levar mais tempo que o previsto para serem desenvolvidos	Técnico	Baixa
R4	O sistema pode não ser aceito, por ser lento ou não se enquadrar nos <i>websites</i> atuais	Negócio	Baixa
R5	Os algoritmos genéticos podem causar um maior custo de processamento que o esperado	Projeto	Baixa
R6	Pode-se deparar com dados de backup corrompidos	Técnico	Baixa
R7	O sistema pode não gerar a melhor sugestão de link para o usuário	Técnico	Média
R8	O usuário prefere procurar seu objetivo que utilizar o sistema de sugestão	Técnico	Baixa
R9	O sistema pode não funcionar caso o admin do <i>website</i> hospedeiro não defina corretamente classes para os usuários	Técnico	Média
R10	Usuários não são formigas, não há como prever seu comportamento no sistema proposto, podendo tornar o sistema completamente ineficaz	Técnico	Média

^aProbabilidade de Ocorrência (PO)

Em um trabalho de pesquisa científica, o resultado não pode ser previsto, diferente de um sistema comercial em que o resultado deve ser traçado como o objetivo da equipe. Este trabalho propõe testar uma idéia e coletar seus resultados, podendo assim divulgar sua eficácia ou suas deficiências.

Os riscos citados têm como parâmetro suposições com o conhecimento técnico obtido ao longo do curso, entretanto não podem ser tomadas como realidade, tão pouco sua PO como real, e sim apenas como uma estimativa.

7 LIMITES DO SISTEMA

É objetivo dessa proposta traçar o escopo do projeto, deixando claro quais são os limites do sistema.

Tabela 7.1: Descrição dos limites do sistema

ID	Limite do sistema
L1	O sistema não se irá direcionar o usuário a seu objetivo, apenas orientá-lo
L2	O sistema não é um <i>website</i> e sim uma ferramenta de orientação aos usuários
L3	O sistema não funcionará isolado, deverá ser aplicado a um <i>website</i> hospedeiro
L4	O sistema não fará uma busca no web site hospedeiro
L5	O sistema não cria as classificações de usuários
L6	O sistema não cadastrará os usuários

8 RECURSOS COMPUTACIONAIS

Os recursos necessários são divididos entre:

8.1 RECURSOS DISPONÍVEIS PARA DESENVOLVIMENTO

8.1.1 Recursos de Hardware

1. Microcomputador Desktop:

- (a) **Processador:** AMD Athlon(tm) XP 2700+ (2.16 GHz);
- (b) **Memória:** 512 MB;
- (c) **Armazenamento:** 120GB + 40GB;

2. Microcomputador Notebook:

- (a) **Modelo:** HP Pavilion dv1040;
- (b) **Processador:** Pentium M Centrino (1.7 GHz);
- (c) **Memória:** 512 MB;
- (d) **Armazenamento:** 60GB;

8.1.2 Recursos de Software

1. Sistemas Operacionais:

- (a) **Windows XP Professional Edition SP2 PT-BR** → Sistema operacional utilizado com o microcomputador *desktop*;
- (b) **Windows XP Home Edition SP2 EN-US** → Sistema operacional utilizado com o microcomputador *notebook*;

2. Outros Softwares:

- (a) **Adobe Photoshop CS2** → Necessário para a construção dos *layouts (design)* do sistema.

- (b) **Adobe Acrobat Reader Professional 7.0** → Necessário para visualização e alteração do *Portable Document Format* (PDF) referente à monografia.
- (c) **Apache HTTP Server 1.3.33** → Servidor HTTP com suporte a linguagem PHP.
- (d) **Macromedia Dreamweaver 8.0** → Utilizado para a codificação e design do sistema.
- (e) **Visual Paradigm versão 5.2** → Utilizado para construção dos diagramas UML.
- (f) **Mozilla Firefox versão 1.5.0.5** → Navegador compatível com as especificações HTML 4.01, CSS 2, e possui suporte a JavaScript. Utilizado para testes no sistema.
- (g) **Internet Explorer 6.0** → Navegador compatível com as especificações HTML 4.01, CSS 2, e possui suporte a JavaScript. Utilizado para testes no sistema.
- (h) **winEdit 5.4** → Programa utilizado para codificação da monografia em \LaTeX .
- (i) **PostgreSQL 8.0.1** → Banco de dados utilizados no projeto.
- (j) **pgAdmin 1.4.2** → Ferramenta para gerenciamento do banco de dados PostgreSQL.
- (k) **ERwin 4.0** → Ferramenta para criação dos diagramas de entidades e relacionamentos.
- (l) **Cerberus FTP Server** → Servidor FTP.
- (m) **MiKTeX 2.5** → Compilador \LaTeX para Windows.
- (n) **PHP 5.1.4** → Linguagem de programação utilizada no projeto.
- (o) **JabRef 2.1** → Ferramenta para criação de referências bibliográficas.
- (p) **LaTable 0.7.2** → Ferramenta para criação de tabelas.

8.2 RECURSOS NECESSÁRIOS PARA UTILIZAÇÃO

8.2.1 Usuário

Para utilizar o sistema o usuário deverá possuir um microcomputador conectado a Internet com um navegador compatível com as especificações HTML 4.01, CSS 2, e deve possuir suporte a JavaScript.

8.2.2 Website Hospedeiro

O sistema proposto, poderá ser implantado em *websites* comuns que contemplem determinadas restrições, tais como:

- O *website* deve estar rodando em um navegador compatível com as especificações HTML 4.01, CSS 2, e deve possuir suporte a JavaScript.

- O *website* deve suportar a linguagem de programação PHP.
- O *website* deve possuir suporte ao banco de dados PostgreSQL 8.1, com disponibilidade de um banco de dados para o sistema a ser implantado;

9 CRONOGRAMAS

9.1 ATIVIDADES

- A1. Reunião com orientador;
- A2. Pesquisa bibliográfica;
- A3. Redação da monografia;
- A4. Defesa do trabalho de diplomação;

9.2 DISCIPLINAS

- D1. Requisitos;
- D2. Análise;
- D3. Projeto;
- D4. Implementação;
- D5. Testes;

9.3 ITERAÇÕES

- I1. Iteração #1 → Módulo: Computação de Feromônio; (Marcado com: ●)
- I2. Iteração #2 → Módulo: Adaptação; (Marcado com: ●)
- I3. Iteração #3 → Módulo: Administração; (Marcado com: ●)

As tarefas estão divididas dentro do prazo previsto para a execução deste trabalho conforme as tabelas 9.2 e 9.1.

Tabela 9.1: Cronograma das atividades desenvolvidas

Atividades	Nov/06				Dez/06				Jan/07				Fev/07				Mar/07				Abr/07				Maio/07				Jun/07				Jul/07			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4				
A1				●		●		●		●		●		●		●		●		●		●		●		●		●		●		●				
A2					●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●													
A3																						●		●		●		●	●	●	●	●				
A4																																●				

Tabela 9.2: Cronograma da metodologia de desenvolvimento

Atividades	Nov/06				Dez/06				Jan/07				Fev/07				Mar/07				Abr/07				Maio/07				Jun/07				Jul/07			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4				
D1				●	●	●	●	●						●	●	●	●	●					●	●	●	●	●									
D2						●	●	●	●	●						●	●	●	●	●					●	●	●	●	●	●						
D3								●	●	●	●	●					●	●	●	●	●	●				●	●	●	●	●						
D4												●	●									●	●								●	●				
D5													●										●									●				

REFERÊNCIAS

- AMARAL, L. G. *Cascading Style Sheets (CSS) - Guia de Consulta Rápida*. [S.l.]: Novatec, 2001.
- ARLOW, J. *UML and the unified process: practical object-oriented analysis and design*. [S.l.]: Addison Wesley, 2002.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. *UML guia do usuário*. [S.l.]: Editora Campus, 2000.
- COUTO, R. *Curso PHP*. 6 2004.
- DORIGO, M.; MANIEZZO, V.; COLORNI, A. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, v. 26, n. 1, p. 29–41, 1996. Disponível em: <citeseer.ist.psu.edu/dorigo96ant.html>.
- FILHO, W. de P. P. *Engenharia de Software: Fundamentos, métodos e padrões*. [S.l.]: LTC, 2003.
- LOPES, H. S. Fundamentos da computação evolucionária e aplicações. *XIII Escola Regional de Informática da Sociedade Brasileira de Computação (SBC) - Paraná*, 2006.
- NADALETE, L. G. Sistema de controle de ocorrências de docentes e discentes. 2005.
- NIEDERAUER, J. *PHP para quem conhece PHP : Recursos avançados para a criação de Websites dinâmicos*. 2. ed. [S.l.]: Novatec, 2004.
- PALAZZO, L. A. M. Sistemas de hipermídia adaptativa. 2002.
- PANSANATO, L. T. E. Aspectos cognitivos de aplicações hipermídia. 2003.
- PARPINELLI, R. S.; LOPES, H. S.; FREITAS, A. A. Data mining with an ant colony optimization algorithm. *IEEE Trans. Evolutionary Computation*, 2002.
- SCOTT, K. *O Processo Unificado Explicado*. [S.l.]: Bookman, 2003.
- SILVA, L. K. da et al. Um assistente digital para responder automaticamente perguntas de usuários humanos em portais corporativos. 2003.
- SOARES, W. *Programando em PHP : conceitos e aplicações*. 2. ed. [S.l.]: Érica, 2000.
- TELES, W. M.; WEIGANG, L.; RALHA, C. G. Uma heurística para guiar os usuários da internet baseada no comportamento da formiga. 2003.
- VIEIRA, P. H. G.; CAIXETA, R. G. Implementação da heurística da formiga para auxiliar à navegação do portal interlegis. 2004.
- WONG, C. *HTTP Pocket Reference: Hypertext Transfer Protocol (Paperback)*. 1. ed. [S.l.]: O'Reilly Media, 2000.
- ZARNANI, A.; RAHGOZAR, M. Mining spatial trends by a colony of cooperative ant agents. 2006.