

# **IMPLEMENTAÇÃO DE UM SISTEMA DE GERENCIAMENTO DE ARBORIZAÇÃO URBANA UTILIZANDO FERRAMENTAS GRATUITAS**

## **IMPLEMENTATION OF AN URBAN FORESTRY MANAGEMENT USING FREE TOOLS**

ORLANDINI, G.<sup>1</sup>, PIRES, L. N.<sup>2</sup>, NETTO, O. E. S.<sup>3</sup>, DELFINO, S. R.<sup>4</sup>

<sup>1a4</sup>Bacharelado em Sistemas de Informação - Faculdades Integradas de Ourinhos –FIO/FEMM

### **RESUMO**

Este artigo descreve sobre o processo de desenvolvimento e implementação de um Sistema para Gerenciamento de Arborização Urbana. Foram utilizadas somente ferramentas de software gratuitas. O sistema, que é acessado via web, é conceitualmente dividido em módulos. Nele o usuário pode cadastrar todas as árvores de uma cidade e todos os pontos onde podem ser plantadas árvores e, a seguir, gerar relatórios com base nestes dados. Podem ser tomadas decisões como: manutenção preventiva, estratégia de combate a pragas e doenças entre outras. O sistema é capaz de importar dados anteriormente cadastrados para serem gerenciados em um módulo próprio de gestão de projetos.

Palavras-chave: Arborização Urbana, Gestão Ambiental, JavaServer Faces

### **ABSTRACT**

This paper describes about the process of developing and implementing a System for Management of Urban Forestry. Have only free software tools. The system, which is accessed via the web, is conceptually divided into modules. There the user can register all the trees of a city and all the places where trees can be planted and then generate reports based on these data. Decisions may be taken as preventive maintenance strategy to combat pests and diseases among others. The system is capable of importing data previously registered to be managed in a module of its own project management.

Keywords: Urban Forestry, Environmental Management

### **INTRODUÇÃO**

A arborização influencia o bem-estar do homem em qualquer ambiente. Isto ocorre através de características próprias das árvores como: formação de sombra, redução da poluição sonora, melhoria da qualidade do ar, conforto térmico entre outros. Para as aves, proporciona abrigo natural.

Em 2007, o governo do Estado de São Paulo lançou o projeto “Município Verde”. A cada ano o projeto ganhou nova edição, sendo somadas diretrizes a serem atendidas pelos municípios do Estado. Em 2010 o projeto recebeu o nome de “VerdeAzul” por acrescentar diretrizes de proteção às águas. Secretarias municipais de meio ambiente, no Estado de São Paulo, buscam

atender as diretrizes do Projeto Ambiental Estratégico Município VerdeAzul (SECRETARIA DE MEIO AMBIENTE DO ESTADO DE SÃO PAULO). Na diretiva 4, o projeto estabelece o valor referencial mínimo de 100 m<sup>2</sup>/hab. de área de sombra projetada por árvores. Este índice é denominado Indicador de Projeção de Copa no Perímetro Urbano (IPCU).

No Estado do Paraná notaram-se de esforços de políticas públicas e de empresas grandes, como a Companhia Paranaense de Energia – COPEL, em implementarem sistemas de informações ambientais. Conforme informado no site da empresa, *“a Copel está desenvolvendo um Sistema de Gestão Ambiental (SGA), no qual existem os Programas Ambientais (PA). Para apoiar alguns PA, foi criado o Sistema de Informações Ambientais (SIA), que são softwares específicos. Alguns deles estão sendo desenvolvidos em plataforma de geoprocessamento.”* (COPEL, 2011).

Em 10 de novembro de 2010 o Governo do Estado do Paraná lançou o “sistema de informações ambientais”<sup>1</sup>, conforme divulgado no site da Agência de Notícias do Estado do Paraná (Agência de Notícias Estado do Paraná, 2010). Trata-se de um site que funciona como portal de informações sobre Meio Ambiente no Estado do Paraná.

Independentemente do Estado, antes de se iniciar projetos de Arborização Urbana, é realizado algum tipo de censo, que visa coletar dados referentes a cada espécime arbóreo plantado no município.

Este levantamento comumente é feito de forma manual, através do preenchimento de formulários de papel que organizam dados como: altura da primeira bifurcação, medida da circunferência do tronco, espécie, rua, número do lote, área de sombra projetada e outras informações relacionadas à árvore.

As informações geradas a partir desse censo dão suporte a tomadas de decisão, como implementação de projetos de manejo. Tais informações são obtidas através da contagem manual dos dados dos formulários obtidos pelo censo. Este processo demanda muito tempo e disponibilidade de mão-de-obra.

---

<sup>1</sup> Segundo a Agência de Notícias do Estado do Paraná, o sistema funciona através do endereço [www.sema.pr.gov.br](http://www.sema.pr.gov.br).

O objetivo deste trabalho foi desenvolver um Sistema de Informações para Gestão de Arborização Urbana. O sistema desenvolvido atenderá empreendedores do Estado do Paraná e do Estado de São Paulo que implementam e executam projetos de arborização urbana.

Os processos a serem informatizados podem ser agrupados em:

- **Inventário:** cadastro de dados sobre árvores e pontos para plantio
- **Relatórios:** geração de informações para embasamento de projetos de arborização urbana e;
- **Gestão de Projetos:** com importação dos dados do inventário, dimensionamento de recursos, geração de cronograma e acompanhamento da execução de cada projeto.

## MATERIAL E MÉTODOS

Existem diversas metodologias para sistematizar o desenvolvimento de software. Elas podem ser divididas em tradicionais e ágeis. As metodologias ágeis “prometem melhorias na produção de software e em sua qualidade” (KOSCIANSKI E SOARES, 2007).

Segundo os mesmos autores as metodologias ágeis “*são adequadas para situações em que a mudança de requisitos é frequente*”. Tal característica encaixa-se ao trabalho proposto dado sua complexidade e possibilidade de diversos tipos de implementações futuras.

O Scrum é uma metodologia ágil e seu ciclo de vida da Scrum é baseado em três fases principais com subfases: pré-planejamento (*pre-game phase*), desenvolvimento (*game phase*) e pós planejamento (*post-game phase*) (KOSCIANSKI E SOARES, 2007).

A Linguagem de Modelagem Unificada (UML) é uma usada para especificar, visualizar, construir e documentar os artefatos de sistemas de software, ela também é usada em projetos que não são de software (LARMAN, 2002).

Neste projeto são utilizados 3 tipos de diagramas do padrão UML: diagrama de casos de uso, diagrama de classes e diagrama de atividades (MELO, 2010).

Para o desenvolvimento dos diagramas, foi utilizada a ferramenta *astah*, que, é um editor de diagramas do tipo UML desenvolvido pela empresa de mesmo nome. O software é distribuído em 3 edições, neste trabalho foi utilizada apenas a edição gratuita *astah community*. Esta atende adequadamente o trabalho proposto por dar suporte a modelagem dos diagramas básicos de UML.

Também foi utilizado o Java, que é uma linguagem de programação orientada a objetos criada para ser usada em dispositivos eletrodomésticos, como decodificadores de TV a cabo e controles remotos (HORSTMANN E CORNEL, 2005).

JavaServer Faces (JSF) é uma tecnologia Java EE desenvolvida para simplificar o desenvolvimento de aplicações web, necessária neste projeto. Torna ágil e simplificado o desenvolvimento de interfaces gráficas (GUI) para web, além de possibilitar o uso de bibliotecas que deixam o design mais amigável ao usuário (GONÇALVES, 2007).

Além disso, JSF é um framework de aplicativo Web que simplifica o design da interface com o usuário de um aplicativo e separa ainda mais a camada de visão de um aplicativo Web da sua regra de negócio (DEITEL E DEITEL, 2010).

Como ambiente de desenvolvimento, adotou-se o Eclipse, que é um software gratuito mantido pela comunidade de fonte aberta Eclipse Foundation. A versão utilizada para este trabalho foi o Eclipse Helios. Existem distribuições do software que são especializadas, como exemplo o Eclipse PHP, para ser utilizada por quem utiliza a linguagem de programação PHP.

Para o banco de dados, adotou-se o programa PostgreSQL é um Sistema de Gerenciamento de Banco de Dados Objeto-Relacional (SGBDOR) baseado no gerenciador de banco de dados POSTGRES versão 4.2 que fora desenvolvido no departamento de Ciência da Computação da Universidade da Califórnia em Berkeley (MANZANO, 2008).

Neste projeto é utilizada a versão 9.0 do SGBD PostgreSQL (última versão disponível quando o sistema começou a ser transformado em linhas de código). Foi escolhido por seus diversos aspectos, sendo os mais atraentes a

sua aceitação pelo Hibernate (ver capítulo correspondente) e por estar disponível gratuitamente um complemento (plugin) para tratamento de coordenadas geográficas (PostGis).

Entre a linguagem Java e o banco existe o Hibernate, que é um framework de código aberto utilizado para o mapeamento objeto-relacional de uma aplicação orientada a objetos para um banco de dados do tipo relacional (BAUER E KING, 2007).

É uma tecnologia cujo objetivo principal é aliviar o desenvolvedor de tarefas comuns relacionadas à persistência de dados, podendo reduzir significativamente o tempo de desenvolvimento gasto com a manipulação de dados no SQL e Java Database Connectivity – JDBC<sup>2</sup> (BAUER E KING, 2007).

Ele funciona em uma camada de interação entre a aplicação e o sistema gerenciador de banco de dados, mapeando objetos em tabelas por meio de arquivos XML. (BAUER E KING, JAVA PERSISTENCE COM HIBERNATE, 2007).

Para geração de relatórios, focou-se no uso do iReport, que é um programa de fonte aberta capaz de criar visualmente relatórios complexos. Além de gerar o relatório propriamente dito é possível exportar a estrutura do relatório no formato da biblioteca JasperReports para ser utilizado por aplicações Java (GONÇALVES, Dominando Relatórios JasperReport com iReport, 2008).

A primeira questão resolvida antes de ser iniciada a implementação do sistema foi “qual o problema do cliente?”. A partir da resposta percebeu-se a que a solução demandaria um sistema de informações computadorizado que pudesse ser acessado via web.

Os requisitos foram levantados a partir de entrevistas com o cliente. Foram realizadas 4 entrevistas que totalizaram 10 horas de conversação. Um dos membros da equipe de desenvolvimento foi o responsável pelas entrevistas. Na metodologia ágil Scrum este é o papel do *product owner*: que é

---

<sup>2</sup> *JDBC* é um conjunto de classes e interfaces (API - Application Programming Interface) escritas em Java que fazem o envio de instruções SQL para qualquer banco de dados.

responsável por compreender as necessidades do cliente sanando as dúvidas da equipe de desenvolvimento com a linguagem técnica necessária.

Após levantados os requisitos, eles foram tabelados, e cada um recebeu um código identificador. A partir da descrição de cada requisito, a equipe de desenvolvimento pôde traçar a prioridade e o esforço necessário em horas para implementar cada requisito.

A organização apresentada na Tabela 1 constitui o que é dominado em Scrum de *product backlog* ou simplesmente de backlog (KOSCIANSKI e SOARES, 2007). O mesmo foi apresentado ao cliente e foi devolvido após 3 dias úteis sem necessidades de mudanças. Foi considerado validado pelo cliente somente após uma explicação, pelo product owner, das descrições de cada requisito detectado.

**Tabela 1.** Product Backlog adaptado a partir da metodologia ágil *scrum* para este projeto.

<i>Id</i>	<i>Nome do requisito</i>	<i>Descrição</i>	<i>Prioridade</i>	<i>Esforço (em horas)</i>	<i>Caso de uso associado</i>
[RF01]	Cadastrar Árvores	Cadastrar dados referentes a árvores. Dados: código, espécie, Imóvel, logradouro, latitude, longitude, altura, área de sombra, diâmetro do tronco, data de plantio, altura da primeira bifurcação, Situação da Cova, situação relativa a rede elétrica, observações.	13	8	Cadastrar Árvore
[RF02]	Cadastrar Ponto Para Plantio	Cadastrar dados referentes ao ponto onde pode ser plantada uma muda: código, logradouro, Imóvel, espécie desejada, situação relativa a rede elétrica, situação da cova e observações	17	12	Cadastrar Ponto Para Plantio
[RF03]	Cadastrar Espécie	Cadastrar dados referentes a espécie: código, família, gênero, nomes populares, altura máxima, melhor época para plantio	1	6	Cadastrar Espécie

Deve-se ressaltar que expõe-se na Tabela 01 apenas 3 dentre n requisitos levantados.

O passo seguinte foi iniciar a construção dos diagramas UML que nortearam as implementações, a começar pelo diagrama de Casos de Uso ( Figura 1). Este diagrama apresenta todos os usos que um usuário (com todas as permissões) pode realizar no sistema.

O diagrama de classes (Figura 2) foi elaborado a partir do agrupamento lógico dos dados e as relações existentes entre estes apresentados nas descrições dos requisitos. Os conceitos de orientação a objetos e normalização de bancos de dados foram considerados na elaboração deste diagrama.

A modelagem do sistema utilizando diagrama de classes UML é peça chave na implementação do sistema por constituir a camada de modelo do sistema, ou seja os objetos de transferências entre telas e banco de dados (também chamados de *transfer objects* – *TO*).

O pacote *backing* reúne as os métodos (ou funções) que podem ser realizados com um objeto no sistema dentro de classes com nomenclatura onde o prefixo está relacionado à classe e sufixo *\_backing* que caracteriza a tal classe no pacote.

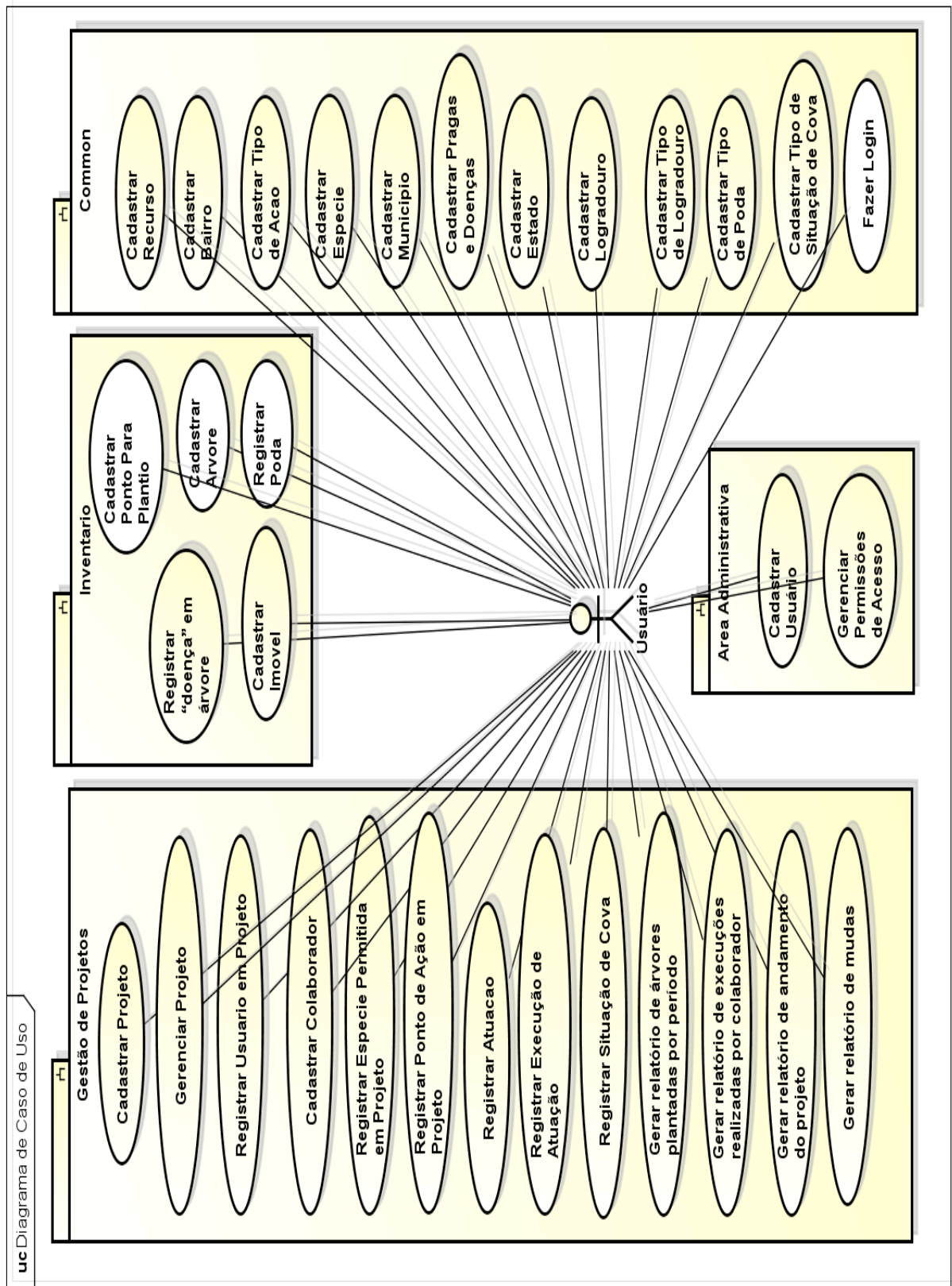


Figura 1. Diagrama de casos de uso do sistema





## RESULTADOS E DISCUSSÃO

Como resultado obteve-se um sistema com várias telas. Uma destas é apresentada na Figura 3, onde tem-se apresentada a interface gráfica do usuário para cadastro de uma árvore no sistema.

Para poder garantir acesso pelo usuário às funcionalidades do sistema na menor quantidade possível de cliques, foi utilizado um leiaute que divide a tela em 3 partes. A superior (top), que ocupa uma faixa horizontal de 200 pontos que além de conter informações extras como nome do sistema, versão, nome do usuário que está utilizando; também foi adicionado uma barra de menu que dá acesso aos principais módulos do sistema, a saber: inventários, cadastros gerais, relatório, gestão de projetos e área administrativa.

The screenshot displays the 'Novo Cadastro de Árvore' (New Tree Registration) page in the SiGAU system. The interface is divided into three main sections: a top header, a left sidebar, and a main content area.

- Top Header:** Features the 'SiGAU' logo with a tree icon, a navigation menu with icons for 'Inventário', 'Novo Cadastro de Árvore', and other modules, and a user profile area.
- Left Sidebar:** Contains a navigation menu with options: 'Árvores', 'Ponto para Plantio', 'Registrar Ocorrência de Poda', and 'Registrar Ocorrência de Praga ou Doença'.
- Main Content Area:** Titled 'Dados da Árvore', it contains a form with the following fields:
  - Código:
  - Espécie:  [Selecione](#)
  - Imóvel a que pertence:  [Selecione](#)
  - Logradouro:  [Selecione](#)
  - Altura:  metros
  - Área de sombra:  metros quadrados
  - Diâmetro do tronco:  centímetros
  - Data provável da próxima poda:
  - Data do plantio:
  - Coordenadas Geográficas:
  - Altura da primeira bifurcação:  metros
  - Área de Sombra Projetada:  metros quadrados
  - Situação relativa a Rede Elétrica:
  - Situação da Cova:

At the bottom of the form are three buttons: 'Gravar', 'Limpar Campos', and 'Cancelar'.

**Figura 3.** Interface gráfica do usuário web para cadastro de Árvore no Sistema de Gestão de Arborização Urbana (SiGAU)

A porção inferior da tela foi dividida verticalmente em 2 partes. A esquerda (left), de largura de 200 pixel contém o menu de acesso as funcionalidades do módulo selecionado. No quadrante da direita são apresentados os formulários que possibilitam o acesso do usuário aos dados.

Com este leiaute foi possível, ao menos na atual versão, garantir o acesso a qualquer cadastro ou registro com 5 cliques ou menos.

Com o desenvolvimento e implantação do sistema, atende-se os requisitos tabulados no product backlog, satisfazendo expectativas do cliente como:

- Inventariação de árvores de um ou mais municípios;
- Geração de relatórios com dados do inventário;
- Implementação de projetos com previsão de prazos e custos;
- Gestão de projetos e geração de relatórios em tempo de execução do projeto;
- Acesso ao sistema via web;

Além disso, o sistema que suporta o desenvolvimento de novos módulos, ampliando o leque de opções para empreendedores da área de gestão ambiental e Secretarias Municipais de Meio Ambiente.

O sistema foi projetado e implementado de forma a possibilitar a inclusão de novas funcionalidades. Isto possibilita que novos tipos relatórios possam ser disponibilizados e mais dados podem ser tratados pelo sistema.

Dado o fato de o SGBD adotado poder ser aperfeiçoado com o uso de uma extensão para tratamento de coordenadas geográficas (PostGis), o que permite que, nas versões posteriores do sistema, o potencial desta ferramenta seja utilizado.

## **CONCLUSÕES**

Após o levantamento de requisitos e implementação do sistema pôde-se concluir que: as Atividades de arborização urbana demandam inventariação e gestão de projetos; os Projetos de arborização urbana requerem previsão de demanda de recursos. Os Dados sobre árvores de uma cidade não são utilizados para novas consultas devido principalmente a desatualização e

demanda de tempo para realização das consultas. Dados de inventários de projetos são pouco aproveitados em gestão de projetos por falta de sistemas baseados em computador que apoiem estas atividades. O sistema desenvolvido neste trabalho resolve as problemáticas das conclusões anteriores e dá abertura a novas possibilidades; conclui-se por final que é possível desenvolver sistema de gestão de arborização urbana utilizando somente ferramentas gratuitas.

### REFERÊNCIAS

AGÊNCIA DE NOTÍCIAS ESTADO DO PARANÁ, 11 nov. 2010. Disponível em: <<http://www.aen.pr.gov.br/modules/noticias/article.php?storyid=60261&tit=Governo-inaugura-biblioteca-e-portal-de-informacoes-ambientais&ordem=250000>>. Acesso em: 01 maio 2011.

BAUER, C.; KING, G. **Java Persistence com Hibernate**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2007.

COPEL. Disponível em: <<http://www.copel.com/hpcopel/root/nivel2.jsp?endereco=%2Fhpcopel%2Fsustentabilidade%2Fpagcopel2.nsf%2Fdocs%2F6FE8DD625C0185D70325740500418044>>. Acesso em: 21 maio 2011.

DEITEL, P.; DEITEL, H. **Java: Como Programar**. 8ª. ed. São Paulo: Pearson Prentice Hall, 2010.

GEARY, D.; HORSTMANN, C. **Core JavaServer Faces**. Rio de Janeiro: Alta Books Ltda., 2005.

GONÇALVES, E. **Desenvolvendo Aplicações Web com JSP Servlets, JavaServer Faces, Hibernate, EJB3 Persistence e Ajax**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2007.

GONÇALVES, E. **Dominando Relatórios JasperReport com iReport**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2008.

HORSTMANN, C. S.; CORNEL, G. **Core Java** <sup>2</sup>. [S.l.]: Alta Books, v. 1, 2005.

JANDL JUNIOR, P. **Desenvolvendo Aplicações WEB com JSP e JSTL**. São Paulo: Novatec, 2009.

KOSCIANSKI, A.; SOARES, M. D. S. **Qualidade de software: aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software**. 2. ed. São Paulo: Novatec Editora, 2007.

LARMAN, C. **Utilizando UML e Padrões**. 2ª. ed. São Paulo: Bookman, 2002.

MANZANO, J. A. N. G. **PostgreSQL 8.3.0**: interativo: guia de orientação e desenvolvimento. São Paulo: Érica, 2008.

MELO, A. C. **Desenvolvendo aplicações com UML2.2**: do conceitual à implementação. 3. ed. Rio de Janeiro: Brasport, 2010. ISBN 978-85-7452-444-3.

SECRETARIA DE MEIO AMBIENTE DO ESTADO DE SÃO PAULO. **Projeto Ambiental Estratégico Município VerdeAzul**. Disponível em: <<http://www.ambiente.sp.gov.br/municipioverdeazul>>. Acesso em: 20 maio 2011.