

## PLANO DE DISCIPLINA

<b>CURSO: ENGENHARIA DE COMPUTAÇÃO</b>		
<b>DIPLINA: QUALIDADE DE SOFTWARE</b>	<b>CÓDIGO: EC P - 845</b>	<b>PERÍODO: 8º</b>
<b>CARGA HORÁRIA: 40 ha = 33,3 h</b>		
<b>REVISÃO: 01/2019</b>		
<b>I – COMPETÊNCIAS</b>		
<p>Conhecer os conceitos de arquitetura de software; conhecer normas de qualidade, confiabilidade e segurança de software; conhecer o modelo CMM e CMMi; conhecer os princípios das normas de qualidade de software como produto, processo e pacote; conhecer conceitos da gestão de configuração de software; conhecer as técnicas e especificação de testes de software; conhecer conceitos básicos de manutenção de software.</p>		
<b>II – HABILIDADES</b>		
<p>Ser capaz de caracterizar as diferentes visões da arquitetura de software; saber especificar e aplicar normas de qualidade, confiabilidade e segurança de software; saber aplicar o modelo CMM e CMMi de qualidade; saber aplicar procedimentos básicos da norma ISO/IEC 25010 e 20246 ; ser capaz de aplicar os conceitos da gestão de configuração de software; ser capaz de aplicar técnicas reengenharia em sistemas legados; ser capaz de planejar, especificar e aplicar técnicas de testes de software; ser capaz de planejar e especificar técnicas de manutenção de software.</p>		
<b>III – CONTEÚDOS PROGRAMÁTICOS</b>		
<p>Arquitetura de software; visão funcional/lógica; visão de código; visão de desenvolvimento/estrutural; visão de concorrência/processo/thread; visão física/evolutiva; visão de ação do usuário/feedback; conceitos de qualidade, garantia de Qualidade de Software, revisão de software; abordagens formais para SQA (Software Quality Assurance); confiabilidade de software, software a prova de erro; normas de Qualidade de Software (padrões de qualidade, ISO/IEC 25010 e 20246); CMM e CMMi; gestão de configuração de software; processo de gestão de configuração de software; identificação de objetos de configuração; controle de versão, controle de modificação, controle de configuração; ferramentas e padrões de SCM; técnicas de teste de software; fundamentos de teste de software: objetivos, fluxo de informações, projeto de casos de teste; teste de caixa branca; teste de caixa preta; teste de caminho básico; teste de estrutura de controle; ferramentas de testes automatizadas; manutenção de software; definição e características; manutenibilidade; tarefas de manutenção; engenharia reversa e reengenharia.</p>		
<b>IV – METODOLOGIA</b>		
<p>A metodologia aplicada utiliza aulas teóricas expositivas mescladas com parte prática (aplicação dos conceitos apresentados) através de exercícios e elaboração de projeto (equipes de alunos).</p>		
<b>V – AVALIAÇÃO</b>		
<p>Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser <math>\geq</math> a 5,0 (cinco inteiros).</p>		
<b>VI – BIBLIOGRAFIA</b>		
<p><b>Básica</b>          PRESSMAN, R. S. <b>Engenharia de Software</b>. São Paulo. McGraw-Hill, 2006.          MOLINARI, L. <b>Testes de Software: Produzindo Sistemas Melhores e Mais Confiáveis</b>. 9ª ed. São Paulo: Érica, 2006.          BARTIE, A. <b>Garantia da qualidade de software</b>. Rio de Janeiro: Elsevier, 2002.</p> <p><b>Complementar</b>          FOWLER, M. <b>UML Distilled: A Brief Guide to the Standard Object Modeling Language</b>. 3ª ed. Boston: Addison-Wesley, 2003.          PADUA, W. <b>Engenharia de Software: Fundamentos, métodos e padrões</b>. 3ª ed. Rio de Janeiro: LTC, 2009.          SOMMERVILLE, I. <b>Engenharia de Software</b>. 8ª ed. São Paulo: Pearson Prentice Hall, 2011. (Acesso Virtual e Físico)          MCMAHON, P. E. <b>Integrating CMMI and Agile Development: Case Studies and Proven Techniques for Faster Performance Improvement</b>. Fairfield: SEI Series in Software Engineering, 2010.          PFLEEGER, S. <b>Engenharia de Software: Teoria e Prática</b>. São Paulo: Pearson Prentice Hall, 2004. (Acesso Virtual)</p>		

## PLANO DE DISCIPLINA

<b>CURSO: ENGENHARIA DE COMPUTAÇÃO</b>		
<b>DISCIPLINA: GESTÃO DE PROCESSOS ORGANIZACIONAIS</b>	<b>CÓDIGO: EC P - 846</b>	<b>PERÍODO: 8º</b>
<b>CARGA HORÁRIA: 40 ha = 33,3 h</b>		
<b>REVISÃO: 01/2019</b>		
<b>I – COMPETÊNCIAS</b>		
Conhecer como funciona a estrutura organizacional das empresas e suas relações com os sistemas e a tecnologia; saber como é feita a análise e o diagnóstico organizacional; conhecer os processos das organizações e suas análises; conhecer as formas de gestão de processos organizacionais; saber como podem ser feitas melhorias nos processos e na organização; conhecer como é feito um plano diretor de tecnologia.		
<b>II – HABILIDADES</b>		
Fazer a análise e o diagnóstico de uma organização utilizando-se de métodos e técnicas consagrados; realizar a análise de processos de uma organização e procurar formas de melhorá-los e otimizá-los de forma contínua; saber como aplicar as formas de gestão por diretrizes e gestão estrutural numa organização; conhecer as melhores práticas internacionais de gestão com qualidade; conceber um plano diretor de tecnologia para uma empresa.		
<b>III – CONTEÚDOS PROGRAMÁTICOS</b>		
Estrutura organizacional; o ambiente das organizações; as relações internas e a estrutura das organizações; tecnologia e sistemas de informação empregados; análise e diagnóstico organizacional; missão, visão, objetivos, metas, valores da organização; hierarquia, departamentalização: conceito e histórico; funcionograma, lotacionograma, organograma; análise e diagnóstico de layout; tipos de layout e aplicação; ergonomia; diagnóstico de problemas em layout; análise de fluxo de trabalho; métodos de operação; conceituação de macro-processo, processo, atividade (tarefa), rotinas e procedimentos; responsabilização sobre os processos; os macro-processos mais comuns de uma organização; análise de processos, fluxogramas, modelagem de processos; revisão de processos e regra para o sucesso na revisão de processos; gestão por diretrizes; conceituação de índice, indicador, métrica e diretriz; técnicas de identificação de indicadores-chave em processos e negócios; introdução à administração estatística com base em diretrizes; gestão de mudança estrutural na organização; melhoria de processos e da organização; conceituação de melhoria contínua; organização da companhia para suportar a melhoria contínua; conceituação de rupturas; conduções de reuniões de negócio e apresentações, técnicas de comunicação; plano diretor de tecnologia; passos de um planejamento de tecnologia aplicada para uma organização; o papel do analista no suporte à elaboração do PDT.		
<b>IV – METODOLOGIA</b>		
A metodologia se baseia em aulas teóricas expositivas mescladas com uma grande quantidade de exercícios em cada unidade do conteúdo programático, além de jogos, dinâmicas de grupo, exibição de vídeo e seminários.		
<b>V – AVALIAÇÃO</b>		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser $\geq$ a 5,0 (cinco inteiros).		
<b>VI – BIBLIOGRAFIA</b>		
<b>Básica</b>		
CHIAVENATO, I. <b>Iniciação a sistemas, organização e métodos</b> . Barueri: Manole, 2010. (Acesso Virtual e Físico)		
BALLESTERO-ALVAREZ, M. E. <b>Manual de organização, sistemas e métodos: abordagem teórica e prática da engenharia da informação</b> . 2ª ed. São Paulo: Atlas, 2000.		
FERREIRA, A. A.; REIS, A. C. F.; PEREIRA, M. I. <b>Gestão empresarial de Taylor aos nossos dias</b> . São Paulo: Pioneira, 2002.		
<b>Complementar</b>		
OLIVEIRA, D. de P. R. <b>Sistemas, organização e métodos: uma abordagem gerencial</b> . 13ª ed. São Paulo: Atlas, 2002.		
ARAUJO, L. C. G. de. <b>Organização, sistemas e métodos: e as modernas ferramentas de gestão organizacional</b> . São Paulo: Atlas, 2001.		
LAUDON, K. C.; LAUDON, J. P. <b>Sistemas de Informação Gerenciais</b> . 11ª ed. São Paulo: Prentice-Hall, 2014.		

## PLANO DE DISCIPLINA

(Acesso Virtual)

DAFT, R. L. **Organizações: teorias e projetos**. São Paulo: Pioneira Thomson Learning, 2002.

JONES, G. R. **Teoria das organizações**. 6ª ed. São Paulo: Pearson Education do Brasil, 2010. (Acesso Virtual)

## PLANO DE DISCIPLINA

<b>CURSO: ENGENHARIA DE COMPUTAÇÃO</b>		
<b>DISCIPLINA:</b> REDE DE COMPUTADORES	<b>CÓDIGO:</b> EC E- 847	<b>PERÍODO:</b> 8º
<b>CARGA HORÁRIA:</b> 80 ha = 66,7 h		
<b>REVISÃO:</b> 01/2019		
<b>I – COMPETÊNCIAS</b>		
Identificar as camadas do Modelo de Referência OSI e Pilha TCP/IP, Identificar os principais tipos de tecnologias de redes de computadores; reconhecer os aspectos físicos e lógicos de redes de computadores; compreender o funcionamento das tecnologias Ethernet e Wi-Fi, compreender o funcionamento dos protocolos IPv4, IPv6, RIP, OSPF e BGP.		
<b>II – HABILIDADES</b>		
Contextualizar as redes de computadores como ferramenta de produtividade e integração de dados, aplicações e pessoas nas organizações; diferenciar e classificar as redes de computadores baseando-se em sua abrangência geográfica e pela distribuição no ambiente; identificar as tecnologias básicas de redes PAN, LAN, MAN e WAN; explorar detalhes físicos de projeto e implantação de redes de computadores, como: topologias, cabeamento e equipamentos básicos de conectividade; explorar detalhes técnicos da camada de rede e seus protocolos; ser capaz de identificar características ligadas ao funcionamento dos protocolos de roteamento RIP, OSPF e BGP.		
<b>III – CONTEÚDOS PROGRAMÁTICOS</b>		
Modelo OSI e arquitetura TCP/IP; classificação das Redes de Computadores; PAN – Personal Area Network; LAN – Local Area Network; MAN – Metropolitan Area Network; WAN – Wide Area Network; arquitetura IEEE802; cabeamento estruturado; padrão EIA/TIA 568; equipamentos de rede; IEEE 802.3; IEEE 802.11, VLANs IEEE802.1q, STP (Spanning Tree Protocol); funções e protocolos da camada de rede; endereçamento IPv4/IPv6 e protocolos de roteamento.		
A metodologia se baseia em aulas teóricas expositivas e aulas práticas com o desenvolvimento de trabalhos, projetos e casos de estudo.		
<b>V – AVALIAÇÃO</b>		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser $\geq$ a 5,0 (cinco inteiros).		
<b>VI – BIBLIOGRAFIA</b>		
<b>Básica</b>		
FOROUZAN, B. A. <b>Comunicação de Dados e Redes de Computadores</b> . São Paulo: McGrawHill Brasil, 2008.		
TANENBAUM, A. S. <b>Redes de Computadores</b> . 5ª ed. Rio de Janeiro: Pearson Prentice Hall, 2011.		
KUROSE, J. F.; ROSS, K. W. <b>Redes de Computadores e a internet: Uma abordagem top-down</b> . 6ª ed. São Paulo: Pearson Addison Wesley, 2013. (Acesso Virtual e Físico)		
<b>Complementar</b>		
RAPPAPORT, T. S. <b>Comunicações sem fio: princípios e práticas</b> . 2ª ed. São Paulo: Pearson, 2009. (Acesso Virtual)		
COMER, D. E. <b>Redes de Computadores e Internet</b> . 6ª ed. Porto Alegre: Bookman, 2016.		
MOREIRAS, A. M. <b>Laboratório de Ipv6: aprenda na prática usando em emulador de redes</b> . São Paulo: Novatec, 2015.		
BRAGA, J. et al. <b>O livro do IETF</b> . São Paulo: Comitê Gestor da Internet no Brasil, 2014.		
PETERSON, L.; DAVIE, B. <b>Redes de computadores: uma abordagem de sistemas</b> . Rio de Janeiro: Elsevier, 2013.		

## PLANO DE DISCIPLINA

<b>CURSO: ENGENHARIA DE COMPUTAÇÃO</b>		
<b>DISCIPLINA: SISTEMAS OPERACIONAIS</b>	<b>CÓDIGO: EC P - 848</b>	<b>PERÍODO: 8º</b>
<b>CARGA HORÁRIA: 80 ha = 66,7 h</b>		
<b>REVISÃO: 01/2019</b>		
<b>I – COMPETÊNCIAS</b>		
<p>Conhecer os conceitos básicos de um sistema operacional; conhecer a evolução e classificação dos sistemas operacionais; conhecer os conceitos de gerência de processos, conceituação de processos, escalonamento e comunicação entre processos; conhecer os conceitos de gerência de memória, sistemas de arquivos e entrada e saída; avaliar as arquiteturas e mecanismos dos sistemas operacionais modernos e dos principais problemas enfrentados na sua construção e as soluções empregadas; conhecer as possibilidades e limitações dos sistemas operacionais modernos, bem como a avaliação dos diferentes sistemas no uso de aplicações específicas.</p>		
<b>II – HABILIDADES</b>		
<p>Ser capaz de identificar as funcionalidades dos sistemas operacionais bem como seus principais serviços; analisar os tipos de sistemas operacionais quanto à sua capacidade de executar tarefas e atender usuários; ser capaz de identificar as diferenças entre processos e threads, bem como avaliar o custo de processamento entre eles; ser capaz de julgar os diferentes algoritmos de escalonamento de CPU e memória; ser capaz de analisar deadlocks de sistema; ser capaz de identificar os mecanismos de gerenciamento de dispositivos e sistema de arquivos.</p>		
<b>III – CONTEÚDOS PROGRAMÁTICOS</b>		
<p>Conceituação de sistema operacional; conceitos básicos sobre Sistemas Operacionais; evolução dos Sistemas Operacionais; classificação dos Sistemas Operacionais; processos; conceituação de processo; escalonamento de processos; escalonamento de processos em sistemas com múltiplos processadores e em sistemas de tempo real; <i>threads</i>; comunicação entre processos; condições de corrida; regiões críticas; exclusão mútua com espera ocupada; bloqueio e desbloqueio de processos; semáforos; <i>deadlocks</i>; recursos; tratamento de <i>deadlocks</i>; recuperação de <i>deadlocks</i>; tentativas de se evitar <i>deadlocks</i>; prevenção de <i>deadlocks</i>; gerência de memória; organização hierárquica da memória; alocação contígua simples; alocação particionada estática e dinâmica; estratégias de alocação; <i>swapping</i>; memória virtual: paginação e segmentação; sistemas de arquivos; conceituação de sistemas de arquivos; arquivos; diretórios; serviços do sistema operacional; implementação lógica; implementação física; Entrada/Saída (E/S); dispositivos de Entrada/Saída; controladoras de dispositivos de Entrada/Saída; módulos de Entrada/Saída; operação de módulos de Entrada/Saída: <i>pooling</i>, interrupção e DMA; conexão com dispositivos de Entrada/Saída; <i>software</i> de Entrada/Saída; unidades de disco; introdução ao Linux.</p>		
<b>IV – METODOLOGIA</b>		
<p>Usar aulas teóricas expositivas intercaladas com exercícios em sala de aula para cada unidade do conteúdo programático, além de aulas práticas no laboratório.</p>		
<b>V – AVALIAÇÃO</b>		
<p>Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser <math>\geq</math> a 5,0 (cinco inteiros).</p>		
<b>VI – BIBLIOGRAFIA</b>		
<p><b>Básica</b>          SILBERSCHATZ, A.; GALVIN, P.; GAGNE, G. <b>Sistemas operacionais: conceitos e aplicações</b>. São Paulo: Prentice Hall, 2000.          TANENBAUM, A. S. <b>Sistemas Operacionais Modernos</b>. 3ª ed. São Paulo: Prentice Hall, 2016. (Acesso Virtual e Físico)          DEITEL, H. M.; DEITEL, P. J.; CHOFFNES, D. R. <b>Sistemas Operacionais</b>. 3ª ed. São Paulo: Pearson Prentice Hall, 2005. (Acesso Virtual e Físico)</p> <p><b>Complementar</b>          CÔRTEZ, P. L. <b>Sistemas Operacionais – Fundamentos</b>. São Paulo: Érica, 2003.          TOBLER, M. J. <b>Desvendando Linux</b>. Rio de Janeiro: Campus, 2005.          MACHADO, F. B.; MAIA, L. P. <b>Arquitetura de Sistemas Operacionais</b>. 4ª ed. Rio de Janeiro: LTC, 2007.          FLYNN, I. M.; MCHOES, A. M. <b>Introdução aos sistemas operacionais</b>. São Paulo: Pioneira, 2002.          NEMETH, E.; SNYDER, G.; HEIN, T. R. <b>Manual Completo de Linux: guia do administrador</b>. 2ª ed. São Paulo: Pearson Prentice Hall, 2007. (Acesso Virtual)</p>		

## PLANO DE DISCIPLINA

<b>CURSO: ENGENHARIA DE COMPUTAÇÃO</b>		
<b>DISCIPLINA:</b> ARQUITETURA DE COMPUTADORES II	<b>CÓDIGO:</b> EC E - 849	<b>PERÍODO:</b> 8º
<b>CARGA HORÁRIA:</b> 80 há = 66.7 h		
<b>REVISÃO:</b> 01/2019		
<b>I – COMPETÊNCIAS</b>		
O aluno será capaz de reconhecer, identificar e trabalhar com as diversas estrutura de microprocessadores x86, memórias física, virtual e compartilhada, dispositivos de comunicação, formatos de instruções e endereçamento, estrutura e função de processamento, linguagem de montagem, computadores RISC, paralelismo, multiprocessamento simétrico, virtualização, computadores multicore, criptoprocessadores.		
<b>II – HABILIDADES</b>		
A aluno será capaz de compreender os sistemas de memórias físicas, virtual e compartilhada. Entender, analisar e adotar os uso diversas instruções e funções de processamento de baixo nível sendo capaz de gerenciar o processamento em seus diversos níveis através do sistema operacional.		
<b>III – CONTEÚDOS PROGRAMÁTICOS</b>		
Memórias física, virtual e compartilhada, dispositivos de comunicação, formatos de instruções e endereçamento, estrutura e funcionamento da CPU, instruções e função binárias de processamento, linguagem de montagem, macros, computadores RISC, paralelismo, multiprocessamento simétrico, virtualização, computadores multicore, criptoprocessadores.		
<b>IV – METODOLOGIA</b>		
Aula expositiva; com recursos audiovisuais; Projeto de pesquisa teórica e prática.		
<b>V – AVALIAÇÃO</b>		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser $\geq$ a 5,0 (cinco inteiros).		
<b>VI – BIBLIOGRAFIA</b>		
<p><b>Básica</b>          TANENBAUM, A. S. et al. <b>Organização Estruturada de Computadores</b>. 6ª ed. São Paulo: Prentice Hall, 2013. (Acesso Virtual e Físico)          STALLINGS, W. <b>Arquitetura e Organização de Computadores</b>. 8ª ed. São Paulo: Pearson, 2017. (Acesso Virtual e Físico)          DELGADO, J.; RIBERIO, C. <b>Arquitetura de Computadores</b>. 2ª ed. Rio de Janeiro: LTC, 2009.</p> <p><b>Complementar</b>          VASCONCELOS, L. <b>Hardware Total</b>. São Paulo: Makron Books, 2002.          RAPPAPORT, T. S. <b>Comunicações sem fio: princípios e práticas</b>. 2ª ed. São Paulo: Pearson, 2009. (Acesso Virtual)          ZELENOVSKY, R. <b>PC - um guia prático de hardware e interfaceamento</b>. 3ª ed. Rio de Janeiro: MZ, 2002.          PATTERSON, D.; HENNESSY, J. L. <b>Organização e Projetos de Computadores: A Interface Hardware/Software</b>. 2ª ed. Rio de Janeiro: Campus, 2003.          HENNESSY, J.; PATTERSON, D. <b>Arquitetura de Computadores: uma abordagem quantitativa</b>. 3ª ed. Rio de Janeiro: Campus, 2003.</p>		

## PLANO DE DISCIPLINA

<b>CURSO: ENGENHARIA DE COMPUTAÇÃO</b>		
<b>DISCIPLINA:</b> LINGUAGEM DE PROGRAMAÇÃO IV	<b>CÓDIGO:</b> EC E - 850	<b>PERÍODO:</b> 8º
<b>CARGA HORÁRIA:</b> 80 ha = 66,7 h		
<b>REVISÃO:</b> 01/2019		
<b>I – COMPETÊNCIAS</b>		
Tecnologia WEB, aprofundamento nas linguagens HTML, CSS e JavaScript (versões mais recentes) para camada cliente (Front-End) com componentes gráficos tal como Material Design. Utilização de frameworks (bibliotecas) JavaScript Angular e Express, arquitetura MVC (Model, View, Controller). Desenvolvimento de serviços (WEB Services) REST, JSON, XML com NodeJS para camada servidor (Back-End). Single Page App e sites reponsivos com Progressive WEB App (PWA). Utilização de banco de dados de documentos tal como MongoDB para dados não estruturados. Utilização de TypeScript (variação do JavaScript) com Angular.		
<b>II – HABILIDADES</b>		
A disciplina tem o objetivo de demonstrar e capacitar os alunos a utilizar os principais recursos da plataforma WEB, implementando os componentes na camada cliente (Front-End) interface com usuário, bem como servidor (Back-End) serviços e criar APIs (Application Program Interface). Aprender programação funcional com JavaScript e TypeScript, programação orientada a aspecto, bem como aprender a utilizar banco de dados de documentos (NoSQL) e identificar o melhor tipo de banco de dados conforme a necessidade (requisito) do sistema.		
<b>III – CONTEÚDO PROGRAMÁTICO</b>		
Conceitos de WEB, HTTP, JSON e NodeJS; programação funcional; definição de arquitetura; desenvolvimento em camadas; NodeJS; Express; Angular; Material Design; MEAN Stack; controle de acesso e sessão; criação de interface amigável para o usuário; tecnologia WEB avançada; REST e RESTful; Mashups; segurança de aplicações WEB; Segurança com banco de dados MongoDB; Construir e consumir APIs em nuvem; Microserviços.		
<b>IV – METODOLOGIA</b>		
A metodologia se baseia em aulas teóricas expositivas, aulas práticas, exemplos de natureza profissional e projetos de natureza corporativa em cada unidade do conteúdo programático, além de um projeto desenvolvido no decorrer da disciplina.		
<b>V – AVALIAÇÃO</b>		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser $\geq$ a 5,0 (cinco inteiros).		
<b>VI – BIBLIOGRAFIA</b>		
<p><b>Básica</b></p> <p>JORGE, M. <b>Java: passo a passo Lite</b>. São Paulo: Pearson Makron Books, 2004. (Acesso Virtual)</p> <p>DEITEL, P. J. <b>Ajax, Rich Internet Applications e Desenvolvimento Web para Programadores</b>. São Paulo: Pearson Makron Books, 2009. (Acesso Virtual)</p> <p>SHARMA, V e SHARMA, R. <b>Desenvolvendo Sites de E-Commerce: como criar um eficaz e lucrativo site de e-commerce, passo a passo</b>. São Paulo: Pearson Makron Books, 2001. (Acesso Virtual)</p> <p><b>Complementar</b></p> <p>SILVA, Maurício. <b>Jquery mobile: desenvolva aplicações WEB para dispositivos móveis com HTML5, CSS3, Ajax, jquery e jquery UI</b>. São Paulo: Novatec, 2013.</p> <p>LEMAY, L.; COLBURN, R.; TYLER, D. <b>Aprenda a Criar Páginas Web com HTML e XHTML em 21 Dias</b>. São Paulo: Pearson Education do Brasil, 2002. (Acesso Virtual)</p> <p>GONÇALVES, E. <b>Ajax na prática</b>. Rio de Janeiro: Moderna, 2007.</p> <p>BONATTI, D. <b>Desenvolvimento de Jogos em HTML 5</b>. Rio de Janeiro: Brasport, 2014. (Acesso Virtual)</p> <p>SOUSA, R. F. M. <b>CANVAS HTML 5 – Composição gráfica e interatividade na web</b>. Rio de Janeiro: Brasport, 2013. (Acesso Virtual)</p>		