



PROJETO PEDAGÓGICO DE CURSO
Pós-Graduação *Lato Sensu*
Desenvolvimento back end

Sumário

1. Nome do Curso e Área do Conhecimento	3
2. Características Técnicas do Curso	3
3. Público-alvo	3
4. Critérios de Seleção	4
5. Justificativa do Curso	4
6. Objetivos do Curso	4
7. Competências e Habilidades do Egresso	5
8. Metodologia de Ensino e Aprendizagem	6
9. Estágio Não Obrigatório	8
10. Matriz Curricular	9
11. Carga Horária	10
12. Conteúdo programático	10
13. Infraestrutura Física e Pedagógica	24

1. Nome do Curso e Área do Conhecimento

Nome do Curso: Desenvolvimento back end

Área de Avaliação (CAPES): Ciência da Computação

Grande Área (CAPES): Ciências Exatas e da Terra

Área do Conhecimento (CAPES): Ciência da Computação

Classificação OCDE: Computação e Tecnologias da Informação e Comunicação (TIC)

2. Características Técnicas do Curso

Modalidade: Educação a Distância

Número máximo de vagas por Polo/Unidade: 1000 alunos

Período de Oferecimento: O curso possui entrada intermitente, respeitadas as datas de início e de fim cadastradas na oferta, bem como observado o período indicado para a sua integralização.

Limitações legais

Resolução CNE/CES Nº 1, de 06 de abril de 2018, que estabelece normas para o funcionamento de cursos de pós-graduação lato sensu.

O candidato deverá ser graduado com diploma devidamente registrado segundo as normas estabelecidas pelo MEC.

3. Público-alvo

O curso de especialização em desenvolvimento back end está destinado a profissionais da área de Tecnologia da Informação ou demais áreas que

desejam aprofundar os seus conhecimentos na concepção, projeto e implementação do back end das aplicações.

4. Critérios de Seleção

O ingresso na pós-graduação será realizado por meio de inscrição no portal da Instituição e entrega dos documentos solicitados. Em seguida, analisados pela área competente.

5. Justificativa do Curso

Ao observar, como usuários, o funcionamento das aplicações e sistemas de informação, não se imagina todo o aparato em segundo plano a sustentá-lo. Esta percepção também é compartilhada com os profissionais da área de tecnologia da informação que não estão envolvidos com tarefas voltadas para programação.

A verdade é que há todo um aparato formado por bancos de dados, linguagens de programação, arquitetura de sistemas, lógica matemática, interface de programação de aplicação, dentre outros componentes, que permitem a exequibilidade de aplicações de negócios.

Em vista deste contexto, o curso de especialização em desenvolvimento back end objetiva formar profissionais capazes de trabalhar com todo o aparato que está por “trás” dos sistemas e sites da web.

6. Objetivos do Curso

6.1. Objetivo Geral

- Capacitar profissionais da área de TI a trabalharem com a concepção, o projeto e a implementação das estruturas de back end de uma aplicação.

6.2. Objetivos Específicos

- Apresentar frameworks back end para a criação de APIs para o front end e outras possíveis aplicações consumidoras.
- Compreender diferenças entre APIs e web services, bem como a sua importância para o desenvolvimento de sistemas.
- Compreender os ambientes de programação e de banco de dados utilizados pelos sistemas de informação.

7. Competências e Habilidades do Egresso

Competências

Atuar no mercado de tecnologia da informação concebendo, projetando e desenvolvendo o back end de aplicação a partir da utilização de frameworks adequados.

Habilidades

- Modelar e analisar arquiteturas de softwares.
- Utilizar framework back end na criação de APIs adequados.
- Escolher linguagens de programação para o desenvolvimento de softwares e aplicações.

8. Metodologia de Ensino e Aprendizagem

O desenvolvimento das disciplinas do curso se dará no ambiente virtual, onde o aluno cumprirá 40 horas por disciplina.

Estas 40 horas são compostas pelo estudo dos diversos insumos pedagógicos disponibilizados, como:

- ✓ materiais de leitura;
- ✓ videoaulas;
- ✓ slides;
- ✓ podcasts;
- ✓ indicações de leituras extras como artigos e capítulos de livros.

Como parte do modelo acadêmico, também contabilizamos:

- ✓ a interação com os tutores para esclarecimentos de dúvidas pedagógicas (via Fórum);
- ✓ a realização da avaliação disciplinar pelo aluno.

Vale ressaltar que cada aluno é único e tem seu próprio tempo para aprendizagem. Alguns conseguem assimilar o conteúdo com mais agilidade, outros precisam retomar as leituras e videoaulas para tornar sua aprendizagem efetiva.

O tempo gasto pelo aluno para o entendimento de cada disciplina depende também da sua familiaridade com o tema, ou seja, o conhecimento prévio que o aluno tem em relação a temática abordada.

No ambiente virtual, o aluno encontrará o conteúdo das disciplinas, organizados em temas/webaulas.

Para cada um deles, o aluno realizará um conjunto de atividades baseadas em leitura de textos de fundamentação teórica e acesso a recursos audiovisuais.

Um tutor apoiará as atividades realizadas no ambiente virtual, atendendo o aluno nas suas dúvidas por meio de ferramentas de comunicação.

O aluno, ao iniciar os seus estudos, terá um encontro presencial para acolhida/ambientação; esse encontro terá como objetivos:

- ✓ Integrar o aluno ao curso de Pós-Graduação.
- ✓ Dialogar e esclarecer as dúvidas sobre a proposta pedagógica do curso e as regras acadêmicas.
- ✓ Apresentar ao aluno o Ambiente Virtual de Aprendizagem (o primeiro acesso; o envio de documentos; os serviços de secretaria e financeiro; a disciplina Ambientação; a tutoria online; o boletim acadêmico; as disciplinas e seus conteúdos; a biblioteca virtual; entre outros).
- ✓ Proporcionar um momento de Network aos pós-graduandos.

Avaliação do Desempenho do Aluno

O aluno deverá realizar as atividades propostas no ambiente virtual. A realização das atividades irá compor sua frequência no curso, que será considerada para a sua aprovação.

A atividade avaliativa que o aluno realizará para compor a sua média é a Avaliação Virtual (AV); essa atividade é obrigatória e estará disponível no Ambiente Virtual de Aprendizagem – AVA, conforme cronograma de seu curso.

Para a aprovação em cada uma das disciplinas, o aluno deverá obter frequência de, no mínimo, 75% (setenta e cinco por cento) e nota igual ou superior a 7,0 (sete).

As notas devem ser expressas no intervalo de 0 (zero) a 10 (dez).

O aluno que obtiver média inferior a 7,0 (sete) nas disciplinas terá direito ao Programa de Dependência e Recuperação – PDR, mediante a solicitação de requerimento e respeitando o período de jubramento do curso.

O PDR será realizado no ambiente virtual de aprendizagem, sendo que o aluno terá acesso ao conteúdo da disciplina e realizará uma Avaliação Virtual - AV, e a nota obtida substituirá a média do aluno.

Para a obtenção do **Certificado** de Pós-graduação *Lato Sensu* – especialização, o aluno deverá cumprir todas condições seguintes:

- ✓ Frequência mínima de 75% (setenta e cinco por cento) em todas as disciplinas;
- ✓ Nota igual ou superior a 7,0 (sete) em todas as disciplinas.

Certificação

O Certificado de conclusão de curso de Especialização será acompanhado por histórico escolar, em cumprimento às exigências da Resolução CNE/CES nº1, de 06 de abril de 2018, da Câmara de Educação Superior do Conselho Nacional de Educação.

Composição do Corpo Docente

O corpo docente do curso é constituído por profissionais qualificados, com comprovado saber em sua área de atuação, conforme Resolução CNE/CES nº1, de 06 de abril de 2018, sendo integrado, no mínimo, por 30% (trinta por cento) de portadores de título de pós-graduação *stricto sensu*, isto é, portadores de títulos de Mestrado e Doutorado, obtidos em programas de pós-graduação *stricto sensu* devidamente reconhecidos pelo poder público em território nacional, ou revalidados, conforme legislação vigente. Os demais docentes são certificados em nível de especialização, pós-graduação *lato sensu*, de reconhecida capacidade técnico-profissional.

9. Estágio Não Obrigatório

O estágio curricular não obrigatório tem como finalidade estimular o aluno a desenvolver atividades extracurriculares, para que possa inter-relacionar os conhecimentos teóricos e práticos adquiridos durante o curso e aplicá-los na solução de problemas reais da profissão, proporcionando o desenvolvimento da análise crítica e reflexiva para os problemas socioeconômicos do país, de acordo com a Resolução de Estágio curricular não obrigatório vigente na Instituição.

Os principais objetivos da prática do estágio curricular não obrigatório são:

I. proporcionar o exercício do aprendizado comprometido com a realidade socioeconômica-política do país;

II. propiciar a realização de experiências de ensino e aprendizagem visando à educação profissional continuada, alicerçada no desenvolvimento de competências e habilidades e ao exercício do pensamento reflexivo e criativo; e

II. incentivar o trabalho de pesquisa e investigação científica, visando ao desenvolvimento da ciência, da tecnologia e da cultura.

A carga horária é definida pela concedente de estágio, não podendo ultrapassar a carga horária máxima de 6 (seis) horas diárias e 30 (trinta) horas semanais, as quais podem ser realizadas em empresas públicas ou privadas, instituição de pesquisa, órgãos governamentais e não governamentais, e as próprias unidades da Universidade, desde que sigam às condições adequadas para que o estagiário possa aprofundar os seus conhecimentos teóricos e práticos adquiridos no curso.

Para o Curso de Pós-Graduação EAD, a prática do estágio curricular não obrigatório é permitida durante a vigência do curso, não podendo exceder em um mesmo campo de estágio o período de 2 (dois) anos.

Os estágios curriculares não obrigatórios devem estar apoiados em Termo de Compromisso e de comum acordo com a Instituição, devendo explicitar não somente os aspectos legais específicos, como também os aspectos educacionais e de compromisso com a realidade social.

O Planejamento do Estágio Curricular Não Obrigatório é de responsabilidade do coordenador de curso/professor orientador e também do Departamento de Estágios.

10. Matriz Curricular

DISCIPLINAS	CH PRÁTICA	CH TEÓRICA	CHTOTAL
-------------	---------------	---------------	---------

Ambientação	0 h	0 h	0 h
Arquitetura de Software	0 h	40 h	40 h
Programação PHP	0 h	40 h	40 h
Frameworks back end	0 h	40 h	40 h
Codificação do projeto: padrões de programação e projeto do banco de dados	0 h	40 h	40 h
Desenvolvimento avançado em Java	0 h	40 h	40 h
Programação com Node.JS	0 h	40 h	40 h
Linguagens de programação para ciência de dados (Python com Spark)	0 h	40 h	40 h
Interface de programação de aplicações (API) e Web Services	0 h	40 h	40 h
Administração de bancos de dados	0 h	40 h	40 h

11. Carga Horária

A carga horária de 360 h constitui o conteúdo ministrado em 9 (nove) disciplinas.

12. Conteúdo programático

Disciplina: Ambientação

Ementa: Histórico da Educação a Distância. Legislação da Educação a Distância no Brasil. Potencialidades da Educação a Distância. Flexibilidade de Acesso. Tecnologias para apoio à pesquisa. Aprendizagem colaborativa. Características do aluno na EAD. Boa convivência virtual: netiquetas.

Competências e Habilidades:

- Identificar as características do aluno da modalidade de educação a distância;
- Utilizar ferramentas tecnológicas de apoio a pesquisa;
- Comunicar-se com clareza e assertividade oralmente e por escrito;
- Gerenciar tempo e atividades de estudo.

Conteúdo Programático 1: Histórico da Educação a Distância.

Conteúdo Programático 2: Legislação da Educação a Distância no Brasil.

Conteúdo Programático 3: Potencialidades da Educação a Distância.

Conteúdo Programático 4: Flexibilidade de Acesso.

Conteúdo Programático 5: Tecnologias para apoio à pesquisa.

Conteúdo Programático 6: Aprendizagem colaborativa.

Conteúdo Programático 7: Características do aluno na EAD.

Conteúdo Programático 8: Boa convivência virtual: netiquetas.

Bibliografia básica:

GOTTARDI, M. de L. A autonomia na aprendizagem em educação a distância: competência a ser desenvolvida pelo aluno. **Associação Brasileira de Educação a Distância**, São Paulo, v. 14, n. 8, p. 110-123, dez, 2015.

MORAN, J. M. MASETTO, M. T. BEHRENS, M. A. **Novas Tecnologias e Mediação Pedagógica**. 21.ed. Campinas, SP: Papirus, 2013.

PIVA, D. J. PUPO, R. GAMEZ, L. OLIVEIRA, S. **EAD na Prática: Planejamento, métodos e ambientes de educação online**. São Paulo: Elsevier, 2011.

Bibliografia Complementar:

ALVES, L. Educação a distância: conceitos e história no Brasil e no mundo. **Associação Brasileira de Educação A Distância**, São Paulo, v. 10, n. 7, p.85-92, out. 2011.

MAIA, Carmen; MATTAR, João. **ABC da EaD**: a educação a distância hoje. São Paulo: Prentice Hall, 2007.

MOORE, Michael G., **Educação a distância**: sistemas de aprendizagem on-line. 3. ed., São Paulo: Cengage Learning, 2013.

Disciplina: Arquitetura de Software

Ementa: Conceitos fundamentais. Projeto de arquiteturas. Modelagem de arquiteturas. Análise de arquiteturas. Implementação e implantação. Estilos arquiteturais aplicados. Projetos em requisitos de qualidade.

Competências e Habilidades:

- Compreender conceitos fundamentais acerca da arquitetura de software;
- Ser capaz de implementar projetos de arquitetura, desde a concepção até a gestão de diferentes processos;
- Modelar e analisar arquiteturas;
- Conhecer diferentes estilos arquiteturais aplicados.
- Compreender a relação dos requisitos de qualidade na escolha e modelagem de arquiteturas de software.

Conteúdo Programático 1: Conceitos Fundamentais (contextualização, arquiteturas, componentes, configurações, estilos e padrões arquiteturais).

Conteúdo Programático 2: Análise e Projeto de Arquiteturas de Software (concepção, ferramentas, arquiteturas para domínios específicos, metas, escopo).

Conteúdo Programático 3: Modelagem de arquiteturas (conceitos, aspectos em ambiguidade e precisão, técnicas comuns, nível de formalidade e automação, frameworks).

Conteúdo Programático 4: Estilos e Projetos Arquiteturais integrados a Requisitos de Qualidade (Implementação, implantação, e projetos arquiteturais de sistemas distribuídos, descentralizadas, orientados a serviços, orientado a microserviços, requisitos de qualidade).

Bibliografia Básica:

BUSCHAMANN et al. A System of Patterns. 1. ed. West Sussex: Wiley, 1996.

FOROUZAN, Behrouz A. Comunicação de Dados e Redes de Computadores. 4. ed. São Paulo: McGraw-Hill, 2008.

GARLAN e SHAW. Introduction to Software Architecture,” Advances in Software Engineering and Knowledge Engineering, Volume I, edited by V.Ambriola and

G.Tortora, World Scientific Publishing Company, New Jersey, 1996. Disponível em:<https://www.cs.cmu.edu/afs/cs/project/able/ftp/intro_softarch/intro_softarch.pdf>. Acesso em: 19 ago. 2018.

Lobo, Edson J. R. Curso de Engenharia de Software. 1. ed. São Paulo: Digerati Books, 2008.

PERRY e WOLF (1992, P. 44), Foundations for the Study of Software Architecture. Disponível em: <<http://users.ece.utexas.edu/~perry/work/papers/swa-sen.pdf>>. Acesso em: 19 ago. 2018.

PRESSMAN, Roger S.; MAXIM, Bruce R. Engenharia de software-9. McGraw Hill Brasil, 2021.

WAZLAWICK, Raul. Engenharia de software: conceitos e práticas. Elsevier Editora Ltda., 2019.

Disciplina: Programação PHP.

Ementa: Conceitos, técnicas, comandos e instruções em PHP; Revisão de Clientes/servidores/TCP; Revisão/contextualização de programação de computadores: HTML e PHP..

Competências e Habilidades:

- Programar na linguagem de desenvolvimento PHP

Conteúdo Programático 1: Instalando o MySQL, Apache e PHP, será apresentado a instalação das ferramentas que serão utilizadas durante a disciplina, mostrando todos dos parâmetros.

Conteúdo Programático 2: Conhecer o ambiente PHP e utilizar os comandos da linguagem PHP conforme a solução web.

Conteúdo Programático 3: Relacionamentos com ambiente web, desenvolver scripts do lado servidor em PHP.

Conteúdo Programático 4: AJAX + PHP Relacionamentos com ambiente web.

Conteúdo Programático 5: Banco de dados com PHP, utilização do banco com a linguagem PHP.

Conteúdo Programático 6: Desenvolvimento de sites em PHP, desenvolver e manter aplicações dinâmicas para Internet com interface web, criando sites dinâmicos com a linguagem PHP.

Conteúdo Programático 7: Sistemas open-source em PHP, criação e apresentação.

Conteúdo Programático 8: Práticas com frameworks PHP criação e apresentação.

Bibliografia Básica:

CONVERSE, Tim; PARK, Joyce. **PHP a bíblia**. Rio de Janeiro: Campus, 2003.

DEITEL, Paul J.; DEITEL, Harvey M. Ajax, Rich. **Internet Applications e desenvolvimento web para programadores**. São Paulo: Pearson Prentice Hall, 2009.

NIEDERAUER, Juliano. **Web Interativa com Ajax e PHP**. São Paulo: Novatec, 2013.

Bibliografia Complementar:

LEMAY, Laura. **Aprenda a criar páginas web com HTML e XHTML em 21 dias**. São Paulo: Makron Books, 2002.

SHEPERD, George. **Microsoft ASP.NET 3.5 - Passo a Passo**. Porto Alegre: Bookman, 2009.

Disciplina: Frameworks back end

Ementa: Diferenciando o *front end* do *back end* e diferenciando as decisões tecnológicas existentes. Visão geral sobre *frameworks back end*. O *Framework* JavaScript Express. O *Framework* Spring Boot para Java. O *Framework* Laravel para PHP.

Competências e Habilidades:

- Conhecer e selecionar *frameworks back end* apropriados para o desenvolvimento de projetos do lado servidor, principalmente para soluções distribuídas;
- Utilizar, na prática, os *frameworks back end*, na criação de APIs para o *front end* e outras possíveis aplicações consumidoras;
- Gerenciar equipes que podem atuar utilizando *frameworks back end*.

Conteúdo Programático 1: Conhecendo e Selecionando *Frameworks Back End*.

Conteúdo Programático 2: Express: Um *Framework Back End* com JavaScript e Node.js.

Conteúdo Programático 3: Potencializando o PHP com o *Framework Back End* Laravel.

Conteúdo Programático 4: Java com o *Framework Back End* Spring Boot.

Bibliografia Básica:

AMBLER, Tim; CLOUD, Nicholas; HAWKES, Robin Andrew. **JavaScript Frameworks for Modern Web Dev**. Nova Iorque: Apress, 2015. 485p.

BROWN, Ethan. **Web development with node and express: leveraging the JavaScript stack**. Sebastopol: O'Reilly Media, 2019.

GABARDO, Ademir C. **Laravel para ninjas**. São Paulo: Novatec Editora, 2017. 480p.

Bibliografia Complementar:

LEONARD, Anghel. **Spring Boot Persistence Best Practices**. Nova Iorque: Apress, 2020. 1027p.

CORDEIRO, Alexandre Costa. **Dyfocus: Desenvolvimento do Back- End de um Aplicativo Mobile para Smartphone**. Monografia (Graduação em Engenharia de Controle e Automação) - Universidade Federal de Santa Catarina. Florianópolis, 71p., 2016.

MORAES, William Bruno. **Construindo aplicações com NodeJS**. São Paulo: Novatec Editora, 2018. 216p.

Disciplina: Codificação do projeto: padrões de programação e projeto do banco de dados

Ementa: Procedimentos de programação com a utilização das principais linguagens orientadas a objeto: Java, C#, Python e Ruby. Projeto da interface do usuário: análise do usuário e do conteúdo. Modelagem de dados e o projeto do fluxo de dados. Projetos de sistemas fazendo uso de tecnologias de Big Data e de Cloud Computing.

Competências e Habilidades:

- Escolher linguagens de programação para o desenvolvimento de softwares orientados a objetos;
- Analisar e modelar a interface do usuário com as funcionalidades e os fluxos de navegações;
- Saber modelar o banco de dados e produzir o fluxo de dados;
- Conhecer ambientes de sistemas em Big Data e ambientes que fornecem funcionalidade de Cloud Computing.

Conteúdo Programático 1: Procedimentos de programação com a utilização das principais linguagens orientadas a objeto: Java, C#, Python e Ruby.

Conteúdo Programático 2: Projeto da interface do usuário: análise do usuário e do conteúdo das funcionalidades.

Conteúdo Programático 3: Modelagem de dados e o projeto do fluxo de dados.

Conteúdo Programático 4: Projetos de sistemas fazendo uso de tecnologias de Big Data e Cloud Computing.

Bibliografia básica:

SOMMERVILLE, Ian. **Engenharia de Software**; tradução Luiz Cláudio Queiroz; revisão técnica Fábio Levy Siqueira. 10. ed. São Paulo: Pearson Education do Brasil, 2018.

PRESSMAN, Roger S. **Engenharia de Software**: uma abordagem profissional; tradução João Eduardo Nóbrega Tortello; revisão técnica Reginaldo Arakaki, Julio Arakaki, Renato Manzan de Andradel. 8. ed. Porto Alegre: AMGH, 2016.

NOGUEIRA, José H. M. **Engenharia de Software**: Métodos Orientados a Objetos e Ágeis. Fortaleza, CE: Amazon KDP, 2018.

Bibliografia Complementar:

PAULA Filho, Wilson de P. **Engenharia de Software**: projetos e processos. 4. ed. Rio de Janeiro: LTC, 2019.

HIRAMA, Kechi. **Engenharia de Software**: qualidade e produtividade com tecnologia. Rio de Janeiro: Elsevier, 2011.

TEOREY, Tobey J. ...[et al.] **Projeto e Modelagem de Banco de Dados**. 2. ed. Rio de Janeiro: Elsevier, 2014.

RODRIGUES MACHADO, Felipe Nery. **Big Data**: O futuro dos dados e aplicações. São Paulo: Érica, 2018.

CHEE, Brian J. S. e FRANKLIN JR, Curtis. **Computação em Nuvem – Cloud Computing – Tecnologias e Estratégias**. São Paulo: MBooks, 2013.

Disciplina: Desenvolvimento avançado em Java

Ementa: Abordar o desenvolvimento avançado na linguagem Java, sobre paradigmas e interfaces funcionais, incluindo conceitos de lambdas, classes anônimas, funções e operators. Novidades das atualizações do JAVA.

Competências e Habilidades:

- Entender o Paradigma Funcional no Java;
- Aprender como utilizar uma lambda e API lambda do Java 8;
- Entender o funcionamento da API de Threads do Java.
- Entender o funcionamento da função PararelStreams no Java.

Conteúdo Programático 1: Conceitos iniciais sobre estrutura de dados, arrays e registro: Entender o Paradigma Funcional no Java; aprender como utilizar uma lambda e API Lambda do Java; Entender paradigma da recursividade (Tail Call Optimization e Memorization).

Conteúdo Programático 2: Interfaces Funcionais: Entender o Paradigma Funcional no Java; aprender como utilizar uma lambda e API lambda do Java 8.

Conteúdo Programático 3: Processamento Assíncrono e Paralelo: Entender o conceito de síncrono e assíncrono; entender o funcionamento da API de Threads do Java; Entender o funcionamento da função PararelStreams no Java.

Conteúdo Programático 4: Por dentro da Modularização do Java: Entender o que é o Jigsaw. Novidades do Java 10. Novidades do Java 11.

Bibliografia Básica:

BARNES, David J.; KÖLLING, Michael. **Programação orientada a objetos com Java:** uma introdução prática utilizando o Blue J. São Paulo: Pearson Prentice Hall, 2004.

FERREIRA, Haline Kelly; ZUCHI, Jederson Donizete. **Análise Comparativa entre Frameworks Frontend baseados em JavaScript para Aplicações Web.** Taquaritinga: Revista Interface Tecnológica, v. 15, n. 2, p. 111-123, 2018.

SILVA, Maurício Samy. **JavaScript - Guia do Programador: Guia completo das funcionalidades de linguagem JavaScript.** . São Paulo: Novatec Editora, 2020. 608p.

Bibliografia Complementar:

DEITEL, Paul J; DEITEL, Harvey M. **Java: Como Programar.** 10ed. São Paulo: Pearson, 2016. 968p.

LECHETA, Ricardo R. **Web Services RESTful: Aprenda a criar web services RESTful em Java na nuvem do Google.** São Paulo: Novatec Editora, 2015. 432p.

VOHRA, Deepak. **JavaServer Faces 2.0: Essential Guide for Developers**. Toronto: Nelson Education, 2014. 432p.

Disciplina: Programação com Node.JS.

Ementa: Apresentar a programação com Node.JS que pode ser aplicada como parte das soluções web, por meio da linguagem JavaScript usando a tecnologia Node.Js. Fundamentação ao Node.js, apresentando o JavaScript como alternativa de linguagem para desenvolvimento de APIs. Diagnosticar o desenvolvimento como base em eventos do Node.js, entendendo as vantagens relacionadas as performances.

Competências e Habilidades:

- Conhecer os fundamentos do desenvolvimento de uma aplicação Web, scripts e lado do servidor para projetos de front end;
- Utilizar ferramentas para gerenciar sessão e controlar cachê de uma aplicação Web;
- Conhecer os conceitos principais da plataforma Node.JS. e o sistema de módulos do Node;
- Implementar recursos de Call Stack, Event Loop e programação assíncrona com Node.

Conteúdo Programático 1: Introdução ao desenvolvimento de uma aplicação Web, scripts lado do servidor. Estrutura do NodeJS. Instalando e Configurando o NodeJS.

Conteúdo Programático 2: Gerenciamento de sessão e controle de Cache. Blocking Vs Non Blocking. NodeJS Entendendo o Núcleo. NodeJS Single Thread E Multithread. Node e Call Stack, Event Loop.

Conteúdo Programático 3: Controlando o fluxo de uma aplicação NodeJS. Ciclo de vida de uma aplicação Node.JS. Entendendo funções de retorno. Programação assíncrona com Node. Async e o Await. Programação dirigida por evento.

Conteúdo Programático 4: API's nativas do NodeJS. Trabalhando com módulos no NodeJS. Compressão de vídeos com FFMPEG e NodeJS. API's de terceiros. Upload de arquivos. Mongo DB.

Bibliografia Básica:

BROWN, Ethan. **Web development with node and express: leveraging the JavaScript stack.** Sebastopol: O'Reilly Media, 2019.

CORDEIRO, Alexandre Costa. **Dyfocus: Desenvolvimento do Back- End de um Aplicativo Mobile para Smartphone.** Monografia (Graduação em Engenharia de Controle e Automação) - Universidade Federal de Santa Catarina. Florianópolis, 71p., 2016.

MORAES, William Bruno. **Construindo aplicações com NodeJS.** São Paulo: Novatec Editora, 2018. 216p.

Bibliografia Complementar:

FERREIRA, Haline Kelly; ZUCHI, Jederson Donizete. **Análise Comparativa entre Frameworks Frontend baseados em JavaScript para Aplicações Web.** Taquaritinga: Revista Interface Tecnológica, v. 15, n. 2, p. 111-123, 2018.

LOUDON, Kyle. **Desenvolvimento de Grandes Aplicações Web: Produzindo código capaz de crescer e prosperar.** São Paulo: Novatec Editora, 2019. 328p.

SILVA, Maurício Samy. **JavaScript - Guia do Programador: Guia completo das funcionalidades de linguagem JavaScript.** . São Paulo: Novatec Editora, 2020. 608p.

Disciplina: Linguagens de programação para ciência de dados (Python com Spark).

Ementa: Apresentação da linguagem Python e sua utilização na análise de dados de grandes bases de dados. Uso em conjunto com Machine Learning. Integração com ferramentas como Spark e Hadoop.

Competências e Habilidades:

- Utilizar linguagem Python.
- Compreender o ambiente de programação em Python

- Entender conceitos de Machine learning

Conteúdo Programático 1: Introdução: Porque Python, Spark e Hadoop? e preparação do ambiente em Spark e Python.

Conteúdo Programático 2: Manipulação de dados com Python.

Conteúdo Programático 3: Organização e visualização de dados.

Conteúdo Programático 4: Análise estatística dos dados.

Conteúdo Programático 5: Machine Learningem Python.

Conteúdo Programático 6: Processando Big Data com Spark.

Conteúdo Programático 7: Real Time Analytics com Python e Spark.

Bibliografia Básica:

MATTHES, E. Curso Intensivo de Python: Uma Introdução prática e baseada em projetos à programação. São Paulo: Novatec Editora, 2017.

CHAMBERS, B.; ZAHARIA, M. Spark: The Definitive Guide: Big Data Processiong Made Simple. San Francisco: O'Reilly Media, 2018.

MENEZES, N. N. C. Introdução à Programação com Python: Algoritmos e Lógica de Programação Para Iniciantes. São Paulo: Novatec Editora, 2017.

BEAZLEY, D.; JONES, B. K. Python CookBook: Recipes for Mastering Python 3. 3ed. San Francisco: O'Reilly Media, 2013.

SLETKIN, B. Python Eficaz: 59 maneiras de programar melhor em Python. São Paulo: Novatec Editora, 2017.

Disciplina: Interface de programação de aplicações (API) e Web Services.

Ementa: Diferenciando e caracterizando as APIs e *Web Services*. Arquiteturas de APIs e *Web Services*. Tecnologias para o desenvolvimento de APIs e *Web Services*. Relação de projetos *front end* com APIs e *Web Services*. Decisões tecnológicas para o desenvolvimento e testes de APIs e *Web Services*.

Competências e Habilidades:

- Conhecer as diferenças entre APIs e *Web Services*, em relação a diferentes especificidades;
- Analisar, planejar e projetar Arquiteturas de APIs e *Web Services*, bem como selecionar tecnologias para a implementação de tais soluções;

- Implementar e testar APIs e Web Services básicos e entender o processo de consumo e conexão com o *front end*.

Conteúdo Programático 1: Caracterizando as APIs dos Web Services.

Conteúdo Programático 2: Análise de Projetos de Arquiteturas de APIs e Web Services e Seleção de Tecnologias para Implementação.

Conteúdo Programático 3: Consumo das APIs e Web Services por soluções do *Front End*.

Conteúdo Programático 4: Decisões tecnológicas para o desenvolvimento e testes de APIs e Web Services.

Bibliografia Básica:

LECHETA, Ricardo R. **Web Services RESTful: Aprenda a criar web services RESTful em Java na nuvem do Google**. São Paulo: Novatec Editora, 2015. 432p.

MASSE, Mark. **REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces**. Sebastopol: O'Reilly Media, Inc., 2011. 116p.

MITCHELL, Lorna Jane. **PHP Web Services: APIs for the Modern Web**. Sebastopol: O'Reilly Media, Inc., 2016.

Bibliografia Complementar:

AGNOL, Giovani Dall; PAVAN, Willingthon. **Web Services—Como Integrar Empresas Através dos Recursos da Internet**. In IX Congreso Argentino de Ciencias de la Computación. 2003.

Zanelato, Ana Paula Ambrósio. **Web Services: uma Nova Arquitetura da Internet**. ETIC - Encontro De Iniciação Científica. v. 6, n. 6, 2010.

RICHARDSON, Leonard; RUBY, Sam. **RESTful web services**. Sebastopol: O'Reilly Media, Inc., 2008. 288p.

PEREIRA, Caio Ribeiro. **Construindo APIs REST com Node.js**: Caio Ribeiro Pereira. São Paulo: Editora Casa do Código, 2016. 193p.

Disciplina: Administração de bancos de dados.

Ementa: Processamento de transações. Controle de concorrências. Técnicas de backup e recuperação de banco de dados. Ferramentas de gerenciamento para

SQL Server e Oracle. Administração de usuários. Segurança de banco de dados. Logs. Sistemas de banco de dados distribuídos.

Competências e Habilidades:

- Criar projetos de banco de dados
- Operar bancos de dados relacionados como administrador de sistemas
- Praticar conceitos de banco de dados em SGBDs de mercado
- Gerir bancos de dados e seus usuários

Conteúdo Programático 1: Sistemas gerenciadores de bancos de dados. SQL Server. Ferramentas de gerenciamento. SQL Server 2019 Express. Configuration Manager, Management Studio e SQLCMD. Arquitetura do banco de dados Oracle. Oracle 10g Express Edition. Ferramentas sqlplus e SQL Developer.

Conteúdo Programático 2: Processamento de transações. Armazenamento e leitura de logs de transações. Estatísticas. Aplicações em SQL Server. Aplicações em Oracle. Gerenciamento de transações. Controle de concorrência. Falhas e propriedades ACID. Transações, locks, blocks e deadlocks.

Conteúdo Programático 3: Técnicas de backup e recuperação de banco de dados. Gerenciamento do espaço em disco. Aplicações em SQL Server. Aplicações em Oracle.

Conteúdo Programático 4: Administração de usuários. Logins, usuários e permissões. Controle de acesso. Segurança de banco de dados. SQL Injection. Criptografia. Comandos em linguagem SQL. Aplicações em SQL Server. Aplicações em Oracle.

Bibliografia Básica:

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados**. 4ed. São Paulo: Pearson, 2009.

LEBLANC, Patrick. **Microsoft SQL Server 2012: Passo a Passo**. Porto Alegre: Bookman, 2014.

WATSON, John. **OCA Oracle Database 11g: Administração I**. Porto Alegre: Bookman, 2014.

Bibliografia Complementar:

DATE, Christopher J. **Introdução a sistemas de bancos de dados**. 8. ed. Rio de Janeiro: Elsevier, 2004.

DE SOUZA, E. P. Migração de banco de dados Oracle para PostgreSQL. **Embrapa Informática Agropecuária**. Documentos, 2007.

DOS SANTOS, Alex Lima et al. FUNCIONAMENTO DO BANCO DE DADOS RELACIONAL SQL SERVER 2000 ENTERPRISE. **Revista de Informática Aplicada**, v. 3, n. 1, 2007. |

13. Infraestrutura Física e Pedagógica

O aluno encontrará todo o conteúdo do curso e assistirá às aulas gravadas no ambiente virtual. Para assistir às aulas é fundamental que as especificações abaixo sejam obedecidas, possibilitando, assim, uma recepção de maior qualidade dos vídeos.

Hardware:

- Processador Intel Core 2 Duo ou superior.
- 2Gb de Memória RAM.
- Placa de vídeo com resolução 1024x768, qualidade de cor 32 bits e compatível com Microsoft DirectShow.
- Microsoft DirectX 9.0c ou posterior.

Software:

- Navegador: Firefox, Google Chrome, Internet Explorer (sempre atualizado).
- Sistema Operacional: Windows XP ou posterior.
- Adobe Flash Player (atualizado).
- *Plugin* de vídeos SilverLigth (atualizado)

Rede:

- Conexão com a Internet banda larga de no mínimo 2 MB.
- Em caso de acesso em ambientes corporativos além da velocidade, é necessário verificar as condições de segurança de rede de sua empresa e se certificar que o site não estará bloqueado.

Adicionalmente, é prevista a utilização da biblioteca virtual para consultas bibliográficas e pesquisa de assuntos referentes às disciplinas ministradas.

