



PROJETO PEDAGÓGICO DE CURSO
Pós-Graduação *Lato Sensu*
Arquitetura de software

SUMÁRIO

1.	Nome do Curso e Área do Conhecimento	2
2.	Características Técnicas do Curso	2
3.	Público Alvo	2
4.	Critérios de Seleção	3
5.	Justificativa do Curso	3
6.	Objetivos do Curso	3
7.	Competências e Habilidades do Egresso	4
8.	Metodologia de Ensino e Aprendizagem	5
9.	Estágio Não Obrigatório	8
10.	Matriz Curricular	9
11.	Carga Horária	10
12.	Conteúdo programático	10
13.	Infraestrutura Física e Pedagógica	26

1. Nome do Curso e Área do Conhecimento

Nome do Curso: Arquitetura de software

Área de Avaliação (CAPES): CIÊNCIA DA COMPUTAÇÃO

Grande Área (CAPES): 10000003 CIÊNCIAS EXATAS E DA TERRA

Área do Conhecimento (CAPES): 10300007 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

Profissionais da área da tecnologia da informação, experientes no desenvolvimento de softwares, com desejo de ampliar competências na

criação de soluções críticas e de alto nível nas diversas plataformas da tecnologia da informação.

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, serão analisados pela área competente.

5. Justificativa do Curso

A oferta deste curso está baseada na grande demanda por sistemas complexos, em multiplataformas, integração entre aplicações, escalabilidade, volume de transação e de tratamento de dados e banco distribuídos. Uma tendência no alinhamento dos sistemas de software aos processos de negócio requer uma arquitetura de software apropriada para cada segmento de negócio. O propósito é atender às necessidades atuais pela heterogeneidade dos alunos, tais como: expectativas, preparação acadêmica, capacidade intelectual, pretensão profissional, entre outras. Enfim, formar desenvolvedores de software habilitados a arquitetar software eficientes e úteis, atendendo às diretrizes do catálogo nacional.

6. Objetivos do Curso

6.1. Objetivo Geral

- Promover a aproximação de conteúdos da tecnologia e das atividades profissionais de TI pensando em estratégias para fazer com que o aluno domine os conhecimentos em arquitetura de software.

- Formar profissionais competentes em arquitetar software adaptáveis nas mais diversas plataformas e segmentos de negócios.

6.2. Objetivos Específicos

- Analisar e arquitetar software, inclusive a arquitetura Data Warehouse
- Desenhar modelos de arquiteturas em *high level* e em nuvem
- Arquitetar software usando o Design Patterns e a Gerência de Configuração
- Conhecer Web Services e Microserviços na arquitetura de software
- Integrar as aplicações com DevOps
- Conhecer a arquitetura em aplicativos Mobile

7. Competências e Habilidades do Egresso

Competências

Arquitetar componentes e aplicativos de softwares com funcionalidades úteis no segmento de mercado atuais e em *high level*, usando tecnologias contemporâneas e eficientes.

Agregar às experiências anteriores em desenvolvimento de sistemas e em variados papéis da TIC com responsabilidade de inovação e de otimização de recursos, através dos ambientes operacionais eficientes e estratégicos para indivíduos e empresas.

Analisar e desenvolver soluções tecnológicas para empreender a arquitetura adequada de software, garantindo a disseminação dessa implementação através de comunicação eficiente a todos os envolvidos, inclusive possibilitando receber atualizações futuras.

Habilidades

Espera-se do egresso no curso de pós-graduação em Arquitetura de Software estar hábil a:

- Projetar e arquitetar componentes de software adequados às necessidades de negócio
- Construir código-fonte eficientes, duráveis e atualizáveis
- Construir e disseminar os artefatos desenvolvidos usando ferramentas e metodologias atuais
- Desenvolver componentes de software prevendo integrações com Web Services e microsserviços
- Criar arquiteturas para aplicações em ambientes Web, DevOps e Mobile.
- Aplicar na prática a Gestão de Configuração e a técnica de *Design Patterns* para controle de artefatos da arquitetura de software.

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	CH TOTAL
Ambientação	0 h	0 h	0 h
Evolução dos softwares: aplicação da Engenharia de Software em sistemas emergentes	0 h	40 h	40 h
Engenharia de requisitos	0 h	40 h	40 h
Arquitetura de Software	0 h	40 h	40 h
Modelagem e Arquitetura do DW (Data Warehouse)	0 h	40 h	40 h
Arquiteturas e infraestruturas em nuvem	0 h	40 h	40 h
Design Patterns e gerência de configuração: do projeto ao controle de versões	0 h	40 h	40 h
Interface de programação de aplicações (API) e Web Services	0 h	40 h	40 h
Aplicações móveis	0 h	40 h	40 h
Práticas da cultura DevOps no desenvolvimento de Sistemas	0 h	40 h	40h

11. Carga Horária

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

12. Conteúdo programático

Disciplina: Ambientação

Ementa: Ensino a distância: características desta modalidade de estudo. A tecnologia e o ensino à distância. Legislação do Ensino a Distância no Brasil. Aspectos relacionados ao perfil no Ensino a Distância.

Competências e Habilidades:

- Compreender a modalidade de ensino a distância.
- Identificar uma relação entre tecnologia e educação no contexto do EAD;
- Discutir elementos da legislação do ensino a distância no Brasil;
- Refletir sobre o perfil do aluno em ensino a distância.

Conteúdo Programático 1: Introdução ao Ensino a Distância.

Conteúdo Programático 2: Tecnologia e educação.

Conteúdo Programático 3: Legislação do Ensino a Distância no Brasil.

Conteúdo Programático 4: Perfil do aluno no Ensino a Distância.

Bibliografia básica:

BRASIL. Ministério da Educação. **Lei de Diretrizes e Bases da Educação Nacional - Lei n. 9.394/96**. Brasília, 1996. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/l9394.htm. Acesso em: 6 dez. 2022.

BRASIL. Ministério da Educação. **Resolução CNE/CP n. 7, de 18 de dezembro de 2018**. Brasília, 2018. Disponível em: https://normativasconselhos.mec.gov.br/normativa/view/CNE_RES_CNECESN72018.pdf. Acesso em: 6 dez. 2022.

GIL, H. A passagem da Web 1.0 para a Web 2.0 e... Web 3.0: potenciais consequências para uma 'humanização' em contexto educativo. 2014. **Boletim informativo Cybercentro Castelo Branco**. Castelo Branco, 2014. Disponível em: <https://repositorio.ipcb.pt/bitstream/10400.11/2404/1/A%20passagem%20da%20Web%20Henrique.pdf>. Acesso em: 6 dez. 2022.

MEYER, A. I. da S. (2022). Conceituando a Educação a Distância. **Revista Ibero-Americana de Humanidades, Ciências e Educação**, v. 8, n. 1, p. 590-601, [s. l.], 2022. Disponível em: <https://periodicorease.pro.br/rease/article/view/3835>. Acesso em: 6 dez. 2022.

VAZ, M. L. de L.; RIBEIRO, F.; COSTA, L. A. da. Os desafios da educação a distância on-line e a remotividade na nova engenharia educacional. **Brazilian Journal of Science**, v. 1, n. 4, p. 79-86. Goiás, 2022. Disponível em: <https://www.brazilianjournalofscience.com.br/revista/article/view/79>. Acesso em: 6 dez. 2022.

VIEIRA, D. M. L.; COSTA, L. A.; VAZ, M. L. L. Um novo olhar para a educação a distância. In: MACHADO, G. E.; COSTA, S. C.; SILVA, K. R. P. **Debates contemporâneos: perspectivas e reflexões atuais**. Santa Maria: Arco, 2021.

Bibliografia Complementar:

BAPTISTA, M. M. Internet: auxílio à educação. **BIBLOS - Revista do Instituto de Ciências Humanas e da Informação**, v. 16, p. 37- 44, [s. l.], 2004.

Disponível em: <http://hdl.handle.net/20.500.11959/brapci/23717>. Acesso em: 6 dez. 2022.

MOORE, M.; KEARSLEY, G. **Educação a distância**: uma visão integrada. São Paulo: Thomson Learning, 2007.

MORAN, J. M.; MASETTO, M. T.; BEHRENS, M. A. **Novas tecnologias e mediação pedagógica**. 21. ed. Campinas: Papirus, 2013.

PIVA, D. J.; PUPO, R.; GAMEZ, L. et al. **EAD na prática**: planejamento, métodos e ambientes de educação on-line. São Paulo: Elsevier, 2011.

Disciplina: Evolução dos softwares: aplicação da Engenharia de Software em sistemas emergentes.

Ementa: Evolução e melhorias dos processos da Engenharia de Software. Engenharia de Software em aplicativos para dispositivos móveis e aplicações WEB (WebApps). Engenharia de Software no desenvolvimento de software de games. Desenvolvimento e gestão de projetos com DevOps.

Competências e Habilidades:

- Conhecer as novas tendências e melhorias nos processos da Engenharia de Software;
- Saber como aplicar a Engenharia de Software no desenvolvimento de aplicações WEB e aplicativos móveis;
- Saber como aplicar a Engenharia de Software no desenvolvimento de games;
- Conhecer a nova cultura DevOps para o desenvolvimento de softwares.

Conteúdo Programático 1: Evolução e melhorias dos processos da Engenharia de Software.

Conteúdo Programático 2: Engenharia de Software em aplicativos para dispositivos móveis e aplicações WEB (WebApps).

Conteúdo Programático 3: Engenharia de Software no desenvolvimento de softwares de games.

Conteúdo Programático 4: Desenvolvimento e gestão de projetos com DevOps.

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.

WAZLAWICK, Raul S. **Engenharia de Software**: conceitos e práticas. 2. ed. Rio de Janeiro: Elsevier, 2019.

Disciplina: Engenharia de requisitos

Ementa: Definição e características da Engenharia de Requisitos com a visão Ágil e o analista de requisitos. Tipos de requisitos: requisitos funcionais e não funcionais, requisitos de domínio, de usuário e de sistemas. Levantamento de requisitos: técnicas de reunião com o usuário e especificação de requisitos. Análise de requisitos, validação, mudança e gerência de requisitos.

Competências e Habilidades:

- Compreender que a Engenharia de Requisitos é importante como base para gerar um software de qualidade;

- Conhecer os tipos de requisitos necessários para a construção do software;
- Aplicar técnicas de reunião com o usuário para abstrair os requisitos do software;
- Analisar e gerenciar os requisitos levantados.

Conteúdo Programático 1: Definição e características da Engenharia de Requisitos com a visão Ágil e o analista de requisitos.

Conteúdo Programático 2: Tipos de requisitos: requisitos funcionais e não funcionais, requisitos de domínio, de usuário e de sistemas.

Conteúdo Programático 3: Levantamento de requisitos: técnicas de reunião com o usuário e especificação de requisitos.

Conteúdo Programático 4: Análise de requisitos, validação, mudança e gerência de requisitos.

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.

MACHADO, Felipe N. R. **Análise e Gestão de Requisitos de Software**: onde nascem os sistemas. 3. ed. São Paulo: Érica, 2016.

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 microsserviços, 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: Modelagem e arquitetura do DW (Data Warehouse)

Ementa: A disciplina de Modelagem e Arquitetura do DW (*Data Warehouse*) traz conhecimentos de modelos e arquiteturas condizentes com DW. Mostra ferramentas construção e verificar dados dentro das estruturas.

Competências e habilidades

- Reconhecer e modelar a arquitetura DW;
- Reconhecer e utilizar ferramentas em DW;
- Efetuar a obtenção de dados dentro do ambiente.

Conteúdo Programático 1: Banco de Dados Transacionais X Bancos de Dados Analíticos.

Conteúdo Programático 2: Conceitos básicos sobre *Data Warehouse*.

Conteúdo Programático 3: *Data Mart*.

Conteúdo Programático 4: Modelagem de Dados para um *Data Warehouse*.

Conteúdo Programático 5: Esquema Estrela e Esquema Floco de Neve.

Conteúdo Programático 6: Ferramentas de Dados em um *Data Warehouse*.

Conteúdo Programático 7: Mineração de Dados em *Data Warehouse*.

Bibliografia básica:

INMON, W. H. **Como Construir o *Data Warehouse***. Rio de Janeiro: Campus, 1997.

KIMBALL, R. ***The Data Warehouse Toolkit: guia completo para modelagem dimensional***. Rio de Janeiro: Campus, 2002.

Machado, F. N. **Tecnologia e Projeto de *Data Warehouse*: uma visão multidimensional**. São Paulo: Érica, 2004.

KIMBALL, Ralph. ***Data Warehouse – Construindo o Data Warehouse para Web***. Rio de Janeiro: Campus, 2000.

Bibliografia complementar

RASLAN, D. A.; CALAZANS, Angélica Toffano Seidel. ***Data Warehouse: Conceitos e Aplicações***. Brasília, DF, Abril de 2014. Disponível em: <<https://www.publicacoesacademicas.uniceub.br/gti/article/viewFile/2612/2400>>. Acesso em 25 set. 2018.

ZIULKOSKI, L. C. C. **Coleta de Requisitos e Modelagem de Dados para *Data Warehouse*: Um Estudo de Caso utilizando Técnicas de Aquisição de Conhecimento**. Porto Alegre, RS, Março de 2003. Disponível em: <<http://www.inf.ufrgs.br/bdi/wp-content/uploads/Monografia-LuisZiulkoski.pdf>>. Acesso em 25 set. 2018.

NOVAIS, R. C. N. **Modelagem Dimensional**. São Paulo, SP, Agosto de 2012. Disponível em: <<http://www.fatecsp.br/dti/tcc/tcc00071.pdf>>. Acesso em 25 set. 2018.

Disciplina: Arquiteturas e infraestruturas em nuvem

Ementa: Conceito de arquitetura em nuvem. Principais provedores de nuvem e tipos de serviços gerais. Participação no mercado de nuvem. Modelos de nuvens: SaaS, PaaS e IaaS. Estudos de caso de arquiteturas em nuvem aplicadas a Banco de dados, Internet das Coisas e aplicações. Provisionamento de custos de nuvem.

Competências e Habilidades:

- Analisar as arquiteturas disponíveis para implementá-las em projetos de virtualização nas empresas.
- Descrever as principais oportunidades de migração de dados para ambiente de nuvem considerando os custos envolvidos.
- Desenhar modelos de arquiteturas em *high level* para apoiar os executivos na tomada de decisão quanto ao uso de seus ativos em nuvem.
- Gerenciar, especificar e documentar arquiteturas de serviços em nuvens, indicando as tecnologias e as ferramentas mais apropriadas para cada tipo de projeto.

Conteúdo Programático 1: Conceito de arquitetura em nuvem. Principais provedores de nuvem e tipos de serviços gerais. Federação. Participação no mercado de nuvem. Relação da arquitetura com a infraestrutura local, híbrida e em nuvem.

Conteúdo Programático 2: Apresentação do modelo SaaS – Software as a Service, objetivos, arquitetura e aplicações. Apresentação do modelo PaaS – Plataforma as a Service, objetivos, arquitetura e aplicações. Apresentação do modelo IaaS – Infraestrutura as a Service, objetivos, arquitetura e aplicações.

Conteúdo Programático 3: Estudo de caso de arquiteturas para provisionar armazenamento de banco de dados. Estudo de caso de arquiteturas para provisionar infraestrutura e serviços em IoT – Internet das Coisas. Estudo de caso de arquiteturas para provisionar serviços de armazenamento e aplicações.

Conteúdo Programático 4: Projeto de provisionamento com arquiteturas mistas e projeção financeira de custos.

Bibliografia Básica:

LOPER, Adriene Aparecida; SILVA, Nathalia dos Santos; LOPES, Giancarlo Michelino Gaeta. **Projeto de redes e sistemas distribuídos**. Londrina: Editora e Distribuidora, 2019.

VERAS, M. **Computação em Nuvem. Nova arquitetura de TI**. Rio de Janeiro: Brasport, 2015. eBook.

VERAS, M. **Virtualização. Tecnologia Central de Datacenter**. Rio de Janeiro: Brasport, 2016.

Bibliografia Complementar:

FERREIRA, Sergio da Costa. **Sistemas de informação em segurança**.

Londrina: Editora e Distribuidora Educacional, 2017.

DANIEL, R. **Cloud Computing. Planejando a sua jornada**. Auto publicação, 2017.

FOWLER, S. J. **Microserviços prontos para a produção**. São Paulo: Novatec, 2017.

Disciplina: *Design Patterns* e gerência de configuração: do projeto ao controle de versões.

Ementa: Apresentação, diferenciação e aplicação dos principais *Design Patterns* existentes (*Gang of Four*). Conceitos básicos sobre Gerência de Configuração, atividades essenciais: controle de mudança, controle de versão e integração contínua. Ferramentas de controle de versões. Integração entre o *Design Patterns* e Gerência de Configuração com o processo de Desenvolvimento de Software.

Competências e Habilidades:

- Conhecer e diferenciar os principais padrões de projeto, ou *Design Patterns* existentes no Grupo elementar de padrões chamado *Gang of Four* propiciando sua adoção em projetos de software;
- Entender a Gerência de Configuração, no que tange as atividades essenciais de: Controle de Mudança, Controle de Versão e Integração Contínua, e como esta gerência pode ser aplicada no desenvolvimento de software.
- Reconhecer e utilizar ferramentas de controle de versão, favorecendo a atividade de Controle de Versão no contexto da Gerência de Configuração;

- Propor projetos de software que integrem *Design Patterns* e a Gerência de Configuração, visando qualidade.

Conteúdo Programático 1: *Design Patterns*: conhecendo os principais padrões.

Conteúdo Programático 2: Gerência de Configuração: atividades, ferramentas e desafios.

Conteúdo Programático 3: Controle de Versões na Prática com Ferramentas de Versionamento.

Conteúdo Programático 4: Integrando *Design Patterns*: gerência de configuração em projetos de software.

Bibliografia Básica:

SHALLOWAY, Alan e TROTT, James. *Design patterns explained: a new perspective on object-oriented design*. Pearson Education, 2004. 480p.

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

WEILL, Peter; ROSS, Jeanne W. **Governança de TI - Tecnologia da Informação**. M. Books, 2020. 296p.

Bibliografia Complementar:

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.

BORGES, Simone de Sousa. **Apoio de gerência de configuração de software ao ARAGAwCRe**. Universidade Federal de São Carlos. Dissertação de Mestrado. São Carlos: UFSCar, 2009. 102f.

SILVERMAN, Richard E. **Git - Guia Prático**. São Paulo: Novatec Editora, 2019. 208p.

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: Aplicações móveis

Ementa: Desenvolvimento de aplicativos móveis com mentalidade digital para negócios de próxima geração.

Conteúdo Programático 1: Introdução à Mobilidade Digital. Revolução da Mobilidade na vida humana; a transformação mobile; bases de dispositivos móveis; smartphones, tablets, relógios inteligentes e outras tecnologias.

Conteúdo Programático 2: Noções Sobre Processos De Desenvolvimento De Software. O Pensamento Lean; o manifesto agile; os princípios de UX Design; técnicas de Design thinking.

Conteúdo Programático 3: Linguagens de Programação. Programação orientada a objetos; linguagens nativas; Android - Java, iOS - Swift e Objective-c, (Windows Phone - C#); ferramentas de desenvolvimento (IDEs, SDKs Frameworks).

Conteúdo Programático 4: Desenvolvimento de aplicações para dispositivos móveis. Desenvolvimento de Mobile Native App; plataforma operacional (Android, iOS, Windows, etc.); desenvolvimento de plataformas híbridas; ferramentas de desenvolvimento híbrido.

Conteúdo Programático 5: Arquitetura e Ambiente de Desenvolvimento. Plataforma operacional (Android, iOS, Windows, etc.); a era da nuvem para dispositivos móveis (Cloud Computing, Cloud Storage, Cloud Services; e, Open Data, Open API, Open Source).

Conteúdo Programático 6: Desenvolvimento Navite App Android. Paas para Aplicativos e Back-end Android; Ferramentas de Desenvolvimento Android; Android Developer Fundamentals.

Conteúdo Programático 7: Teste De Software. Teste De Aplicativos Sob Demanda Em Um Serviço Na Nuvem.

Conteúdo Programático 8: Boas Práticas De Gestão De Projetos. Critérios para a Escolha do Hosting; diferenças Native App vs. Híbrido; adequação do cloud plataforma ao projeto/negócio; adequação do cloud plataforma ao projeto/negócio.

Bibliografia:

ALVES, Thiago Salhab. **Tecnologias para web e para dispositivos móveis**. Londrina: Editora e Distribuidora Educacional S/A, 2016.

ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. **Fundamentos da programação de computadores**. 2. ed. São Paulo: Pearson Prentice Hall, 2007.

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

BONGIORNO, Renato. **Implementação de negócios digitais**. São Paulo: Editora Senac São Paulo, 2017.

BREITMAN, Karin. **Web semântica: a internet do futuro**. Rio de Janeiro: LTC, 2005.

GOMES, Rafael Caveari; FERNANDES, Jean Alves R.; FERREIRA, Vinícius Corrêa. **Sistema Operacional Android**. 2012. 30f. TCC (graduação) – Curso de

Engenharia de Telecomunicações. Universidade Federal Fluminense. Rio de Janeiro, 2012.

HOWARD, Michael. **Escrevendo Código Seguro: estratégias e técnicas práticas para codificação segura de aplicativos em um mundo em rede**. 2. ed. Porto Alegre: Bookman, 2005.

LECHETA, Ricardo R. **Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 3. ed. São Paulo: Novatec, 2013.

LEE, Valentino; SCHNEIDER, Heather; SCHELL, Robbie. **Aplicações móveis: arquitetura, projeto e desenvolvimento**. São Paulo: Pearson Education do Brasil, 2005.

SUTTER, Herb. **Programação avançada em C++**. São Paulo: Pearson Mkron Books, 2006.

Disciplina: Práticas da cultura DevOps no desenvolvimento de Sistemas

Ementa: Compreendendo os pilares da Cultura DevOps no contexto do desenvolvimento Full Stack; Ferramentas do Ciclo de Vida do DevOps com foco na Integração Contínua. Tecnologias Git com Integração Contínua na Prática. Gerenciamento de equipes no desenvolvimento web com DevOps.

Competências e Habilidades:

- Compreender e Adotar a Cultura DevOps para departamentos de Tecnologia da Informação e Equipes de Desenvolvimento;
- Integrar as práticas da cultura DevOps no contexto de desenvolvedores Full Stack e das soluções implementadas;
- Conhecer e selecionar ferramentas específicas para etapas diferentes do processo de desenvolvimento de software;
- Utilizar ferramentas Git e de integração contínua na prática, com projetos ponto a ponto.

Conteúdo Programático 1: Conceitos Essenciais sobre a Cultura DevOps aplicada ao Desenvolvimento de Sistemas.

Conteúdo Programático 2: Selecionando Conhecendo Ferramentas do DevOps no Desenvolvimento Web.

Conteúdo Programático 3: Utilizando Tecnologia de Versionamento Git e de Integração Contínua na Prática.

Conteúdo Programático 4: Adoção das Práticas DevOps em Equipes de Desenvolvimento.

Bibliografia Básica:

KIM, Gene; HUMBLE, Jez; DEBOIS, Patrick; WILLIS, John. **Manual De DevOps: Como obter agilidade, confiabilidade e segurança em organizações tecnológicas.** Rio de Janeiro: Alta Books, 2020. 464p.

MORAES, Gleicon. **Caixa de Ferramentas DevOps: Um guia para construção, administração e arquitetura de sistemas modernos.** São Paulo: Editora Casa do Código, 2015. 176p.

SATO, Danilo. **DevOps na prática: Entrega de software confiável e automatizada.** São Paulo: Editora Casa do Código, 2014. 248p.

Bibliografia Complementar:

LWAKATARE, Lucy Ellen et al. **DevOps in practice: A multiple case study of five companies.** Information and Software Technology, v. 114, p. 217-230, 2019.

RATO, Francisco de Oliveira. **DevOps em sistemas de informação: implementação em operações de tecnologias de informação.** Dissertação (Mestrado em Gestão de Informação) - Universidade Nova de Lisboa. Lisboa, 50p., 2018.

BRAGA, Filipe Antônio Motta. **Um panorama sobre o uso de práticas DevOps nas indústrias de software.** Dissertação (Mestrado em ciência da computação) - Universidade Federal de Pernambuco. Recife, 124p., 2015.

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.

