



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

PLANO DE DISCIPLINA		
IDENTIFICAÇÃO		
CURSO: CST em Análise e Desenvolvimento de Sistemas		
DISCIPLINA: Programação para a Web II		CÓDIGO DA DISCIPLINA: 62
PRÉ-REQUISITO: 51 (Desenvolvimento de Aplicações Corporativas)		
UNIDADE CURRICULAR: Obrigatória [x] Optativa [] Eletiva []		SEMESTRE: 6º
CARGA HORÁRIA		
TEÓRICA: 43 h/a	PRÁTICA: 40 h/a	EaD: h/a
CARGA HORÁRIA SEMANAL: 6h/a	CARGA HORÁRIA TOTAL: 83 h/a	
DOCENTE RESPONSÁVEL : Diego Ernesto Rosa Pessoa		

EMENTA

Arquitetura, ciclo de vida, conceitos e ferramentas para a construção de Aplicações Web avançadas. Novos padrões arquiteturais e paradigmas de desenvolvimento. Tópicos avançados e tendências.

OBJETIVOS

Geral

- ☐ Exposição prática a conceitos, ferramentas e princípios do desenvolvimento de aplicações Web baseadas na arquitetura de microsserviços, juntamente com as boas práticas e técnicas de implantação utilizando-se dos princípios de DevOps.

Específicos

- ☐ Tornar o aluno capacitado a entender os fundamentos, desenvolver e gerenciar uma aplicação Web avançada baseada em microsserviços.
- ☐ Aplicar e gerenciar os principais frameworks utilizados no desenvolvimento de aplicações Web
- ☐ Apresentar todo o fluxo de desenvolvimento de uma aplicação Web avançada, da concepção, desenvolvimento, configuração, implantação e disponibilização para o usuário final.





MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

CONTEÚDO PROGRAMÁTICO

1. Nivelamento, revisão e aprofundamento de conceitos fundamentais
 - 1.1. Controle de versão e integração contínua
 - 1.2. Ferramentas para construção automática de projetos
2. Migrando de aplicações monolíticas para aplicações baseadas em microserviços
 - 2.1. Definição, propósito e conceitos fundamentais da arquitetura de microserviços
 - 2.2. Modelagem e estratégias para decomposição de microserviços
 - 2.3. Visão geral dos componentes de uma aplicação Web baseada em microserviços
 - 2.4. Frameworks e ferramentas de desenvolvimento (Spring Boot, Netflix OSS)
3. Desenvolvendo lógica de negócio numa arquitetura de microserviços
 - 3.1. Padrões de organização de lógica de negócio (*Transaction Script*, DDD)
 - 3.2. Publicação de eventos de domínio
 - 3.3. Mapeamento do modelo de dados em classes de domínio (entidades e migrações)
 - 3.4. Serviços para gerenciamento de regras de negócio e acesso à base de dados
 - 3.5. Disponibilização de APIs REST com paginação e consultas locais
4. Comunicação entre microserviços
 - 4.1. Padrões de Comunicação entre aplicações Web
 - 4.2. Formatos de Mensagens
 - 4.3. Gerenciamento de transações
 - 4.4. Mecanismos para Distribuição de carga alta-disponibilidade
 - 4.5. Comunicação assíncrona e Message Brokers
5. Consultas envolvendo microserviços
 - 5.1. Consumo de dados utilizando o padrão API *composition*
 - 5.2. Adoção do padrão CQRS (*Command Query Responsible Segregation*)
 - 5.3. Criação de visões para otimização de buscas
6. Padrões para consumo como APIs externas
 - 6.1. Desafios para integração de aplicações externas (Clientes Mobile, Front-end de aplicações Web)
 - 6.2. O padrão API Gateway
 - 6.3. Tecnologias para implementação do API Gateway (Spring Cloud, Eureka, Zuul)
 - 6.4. Integração com tecnologias front-end
7. Desenvolvendo serviços prontos para produção
 - 7.1. Segurança em microserviços (UAA – User account and Authentication, SSL)
 - 7.2. Gerenciamento de Configuração externa (*push-based* e *pull-based*)
 - 7.3. Monitoramento de serviços: *health check*, *logging*, métricas, auditoria e *exception tracking*
 - 7.4. Preparando perfis para diferentes ambientes (dev, teste, prod)
8. Implantação de microserviços
 - 8.1. Componentes do *pipeline* para geração e implantação de serviços
 - 8.2. Construção do pacote executável e cópia para um repositório compartilhado
 - 8.3. Geração de imagem para implantação da aplicação como container
 - 8.4. Avaliação de qualidade do código
 - 8.5. Implantação da aplicação em um serviço na nuvem (Heroku, AWS)

METODOLOGIA DE ENSINO

A metodologia de ensino adotada compreende um misto de aulas tradicionais com atividades e workshops em sala de aula. Durante as aulas, serão abordados os conceitos básicos relacionados aos tópicos tratados na disciplina. Durante os workshops em sala de aula, serão realizados exemplos de exercícios com ferramentas relevantes que reforçam o material de aula.

RECURSOS DIDÁTICOS



INSTITUTO FEDERAL
PARAÍBA
Campus Cajazeiras

Rua José Antônio da Silva, 300, Jardim Oásis,
Cajazeiras, PB, 58900-000
Fone: 3532-4160
campus_cajazeiras@ifpb.edu.br



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

- ☒ Quadro
- ☒ Projetor
- ☐ Vídeos/DVDs
- ☒ Periódicos/Livros/Revistas/Links
- ☐ Equipamento de Som
- ☒ Laboratório
- ☒ Softwares: IntelliJ IDEA, Eclipse, Java 8 (JDK e JRE), Maven, Gradle, PostgreSQL, Docker, NodeJS, NPM, YARN, Apache Kafka, Python 3, PIP 3, MongoDB, Neo4j
- ☐ Outros:

CRITÉRIOS DE AVALIAÇÃO

As avaliações serão baseadas nas aulas, exercícios dirigidos, workshops e o projeto final.

BIBLIOGRAFIA

Bibliografia Básica:

- LUCKOW, Décio; MELO, Alexandre. **Programação Java para a Web – 2ª Edição**. Novatec, 2015.
- DEITEL, Paul; DEITEL, Harvey. **Java: como programar**. 10ª Edição. São Paulo: Pearson, 2016
- KALIN, Martin. **Java Web Services: implementando**. Rio de Janeiro: Alta Books, 2010.

Bibliografia Complementar:

- RICHARDISON, Chris. **Microservices Patterns**. Manning Publication, 2018. Disponível em: <https://www.manning.com/books/microservices-patterns>
- CARNELL, John. **Spring Microservices in Action**. Manning Publication, 2017.
- POSTA, Christian. **Microservices for Java Developers**. O' Reilly, 2016. Disponível em: <https://www.oreilly.com/learning/microservices-for-java-developers>
- WALLS, Craig. **Spring Boot in Action**. Manning Publication, 2015.
- HEMRAJANI, Anil. **Desenvolvimento ágil em Java com Spring, Hibernate e Eclipse**. São Paulo: Pearson, 2006.

