

## PLANO DE ENSINO

### 1. IDENTIFICAÇÃO

<b>Disciplina</b>	Programação IV	<b>Código</b>		<b>Período</b>	4º
<b>Curso</b>	Bacharelado em Sistemas de Informação			<b>Semestre</b>	2025.1
<b>Créditos</b>					
<b>Professor</b>	Augusto César Ferreira de Miranda Oliveira				
<b>Pré-Requisitos</b>	Programação II				
<b>Carga Horária Teórica</b>	45	<b>Carga Horária Prática</b>	45	<b>Carga Horária Total</b>	90

### 2. DETALHAMENTO

<b>EMENTA</b>
<p><b>Fundamentos da arquitetura Cliente x Servidor:</b> Desenvolvimento back-end e front-end; Linguagens e Frameworks; Instalação e configuração de plataforma e ambiente de desenvolvimento.</p> <p><b>Tecnologias e fundamentos de desenvolvimento back-end:</b> Servidores de Aplicação; Controle de Sessões, Cookies, cache. Fundamentos de Arquitetura e Padrões Web-Based; Utilização e integração com SGBD; Gerenciamento e controle transacional; Mapeamento Objeto Relacional; Autenticação e Autorização de Usuários.</p> <p><b>Tecnologias e fundamentos de desenvolvimento front-end:</b> HTML, CSS e JavaScript; Single Page Applications; UI/UX.</p>
<b>OBJETIVOS</b>
<b>GERAL</b>
<p>Ser capaz de:</p> <ul style="list-style-type: none"> <li>• Compreender o funcionamento de aplicações web-based fundamentadas na arquitetura cliente x servidor;</li> <li>• Produzir soluções de software na arquitetura web-based utilizando tecnologias, linguagens e frameworks modernos e consolidados na indústria;</li> <li>• Desenvolver soluções integradas, utilizando os padrões arquiteturais consolidados e de referência na arquitetura web-based.</li> </ul>
<b>ESPECÍFICOS</b>
<ol style="list-style-type: none"> <li>1. Conhecer e utilizar linguagens e frameworks consolidados na indústria na construção de soluções web-based;</li> <li>2. Saber construir interfaces com o usuário capazes de propiciar experiência rica e integrá-las à aplicação backend;</li> <li>3. Construir e configurar o ambiente de desenvolvimento e de produção para criação e instalação de soluções web-based.</li> </ol>
<b>CONTEÚDO PROGRAMÁTICO</b>
<ol style="list-style-type: none"> <li>1. Arquitetura Cliente x Servidor             <ol style="list-style-type: none"> <li>a. O protocolo HTTP: O lado cliente, o lado servidor e as mensagens.</li> </ol> </li> <li>2. O desenvolvimento back-end, front-end e suas tecnologias</li> <li>3. Fundamentos de Arquitetura e Padrões Web-Based             <ol style="list-style-type: none"> <li>a. Arquitetura de Software e Padrões de Projetos                 <ol style="list-style-type: none"> <li>i. Separação de Responsabilidades;</li> <li>ii. Arquitetura em Camadas;</li> </ol> </li> </ol> </li> </ol>

- iii. Data Access Object (DAO);
  - iv. Modelo Visão Controle (MVC);
  - v. Web Services: Arquitetura Representational State Transfer (REST);
  - vi. Inversão de Controle (IOC);
  - vii. Injeção de Dependências (CDI).
- b. Definição da Arquitetura e Modelos do Sistema
  - i. Visões Arquiteturais;
  - ii. Diagramas UML 4.
- 4. Frameworks e tecnologias do desenvolvimento WEB
  - a. Definição das tecnologias e ferramentas;
  - b. Instalação e configuração dos ambientes de desenvolvimento e produção;
  - c. Definição da Arquitetura de Referência, frameworks e tecnologias do projeto.
- 5. Construção de Aplicação Back-end
  - a. Camada de Acesso à dados
    - i. Configuração, conexão e comunicação com o SGBD;
    - ii. Implementação do DAO;
    - iii. Criação de testes unitários.
  - b. Camada de Serviço
    - i. Configuração de controle transacional;
    - ii. Configuração, implementação e validação de Regras de Negócio.
  - c. Camada de Controle
    - i. Configuração de WebServices no padrão REST;
    - ii. Configuração, implementação e validação das comunicações com camada de serviço.
  - d. Configuração do tratamento de exceções
- 6. Construção de Aplicação Front-End
  - a. HTML, CSS e JavaScript;
  - b. Aplicações Single Page Application (SPA);
  - c. Arquiteturas, Frameworks e Padrões no desenvolvimento de aplicações SPA;
  - d. Noções de UI/UX e prototipação de interfaces com o usuário;
  - e. Construções de Interfaces com Usuário utilizando framework SPA;
  - f. Integração com aplicações Back-End.
- 7. Instalação e Implantação em ambiente de produção da aplicação construída

#### **METODOLOGIA DE ENSINO-APRENDIZAGEM**

Metodologia dinâmica, priorizando a aprendizagem do estudante. Exposição dialogada com utilização de diferentes recursos didáticos. Abordagem PBL (Aprendizagem Baseada em Problemas). Abordagem PDCA (Plan, Do, Check, Act). Trabalhos individuais – leituras, pesquisas bibliográficas, pesquisas de campo, observações, provas. Trabalhos de Grupos – projetos, seminários, painéis, palestras, conferências, e dinâmicas de grupo. Textos, livros, periódicos, vídeos/filmes, filmes/filmadora, slides, materiais de expediente diversos e adequados à disciplina.

#### **RECURSOS DIDÁTICOS**

- 1. Google Slides: Apresentação do conteúdo proposto;
- 2. Computador e Visual Studio Code: Demonstração prática do conteúdo proposto;
- 3. Google Classroom: Disponibilização dos recursos da aula e exercícios;
- 4. GitHub: Disponibilização do código-fonte dos exemplos práticos da aula.

#### **METODOLOGIA DE AVALIAÇÃO**

A avaliação é contínua, sistemática, processual e participativa. Na avaliação da aprendizagem dar-se-á especial atenção ao trabalho individual do discente. São avaliados os conhecimentos adquiridos, as habilidades, e atitudes indispensáveis à formação do Bacharel em Engenharia de Software. A avaliação do ensino com a participação do discente tem como finalidade precípua à melhoria do desempenho docente. No processo avaliativo serão utilizados procedimentos e instrumentos de avaliação tais como: realização de projetos, provas, observação, autoavaliação, avaliação cooperativa, e amostra de trabalho.

#### **REFERÊNCIAS BIBLIOGRÁFICAS**

##### **BÁSICA**

1. Spring Boot in Action, Craig Walls, Manning Publications, 2016
2. Hands-on: Full Stack Development with Spring Boot 2 and React, Packt Publishing
3. Java: Como programar, Pau Deitel e Harvey Deitel, Person Universidades, 2016
4. Padrões de Arquitetura de Aplicações Corporativas, Martin Fowler, Bookman, 2018
5. UML 2: Uma abordagem prática, Gilleanes Guedes, Novatec, 2018
6. HTML & CSS: design and build websites, Jon Duckett, Alta Books, 2016
7. JavaScript: The definitive Guide: Master the World Most-Used Programming Language, David Flanagan, O'Reilly Media, 2020

##### **COMPLEMENTAR**

1. Software Engineering, 8th edition, Ian Sommerville, Pearson Addison-Wesley, 2006.
2. Software Engineering: A Practitioner's Approach, Roger S. Pressman, 6th edition, McGraw-Hill, 2004.
3. Clean Architecture, Robert Martin e Kelvin Henney, Pearson, 2017
4. Clean Code: A Handbook of Agile Software Craftsmanship, Robert C. Martin, Prentice Hall, 2008
5. The pragmatic programmer: Your Journey to Mastery, David Thomas e Andrew Hunt, AddisonWesley Professional, 2019

Planejamento de Aulas - Programação IV - 2025.1

#	 Data	CH	CH acumulada	 Momento	 Tipo	Tr Conteúdo
1	27/02/2025	3	3	Momento 1	Teórica	Apresentações e Ementa
2	27/02/2025	3	6	Momento 2	Teórica	Quiz - Programação para Web
3	06/03/2025					Feriado: Revolução Pernambucana (Data Magna do Estado)
4	06/03/2025					Feriado: Revolução Pernambucana (Data Magna do Estado)
5	13/03/2025	3	9	Momento 1	Teórica	Introdução ao front-end e as suas tecnologias; Estrutura de um documento HTML
6	13/03/2025	3	12	Momento 2	Prática	Tags e elementos HTML
7	20/03/2025			Momento 1	Prática	Formulários
8	20/03/2025			Momento 2	Prática	Tags semânticas e estruturação avançada
9	27/03/2025	3	15	Momento 1	Teórica	Seletores e propriedades CSS
10	27/03/2025	3	18	Momento 2	Prática	Box Model
11	03/04/2025	3	21	Momento 1	Teórica	CSS Flexbox e CSS Grid
12	03/04/2025	3	24	Momento 2	Prática	CSS Flexbox e CSS Grid
13	10/04/2025	3	27	Momento 1	Teórica	Responsividade (media queries)
14	10/04/2025	3	30	Momento 2	Prática	Introdução ao JavaScript
15	17/04/2025					Feriado: Semana Santa
16	17/04/2025					Feriado: Semana Santa
17	24/04/2025	3	33	Momento 1	Teórica	HTML DOM
18	24/04/2025	3	35	Momento 2	Prática	HTML DOM
19	01/05/2025					Feriado: Dia do Trabalho
20	01/05/2025					Feriado: Dia do Trabalho
21	08/05/2025	3	38	Momento 1	Teórica	Introdução ao React
22	08/05/2025	3	41	Momento 2	Prática	Componentização, props e estado
23	15/05/2025	3	44	Momento 1	Teórica	Introdução ao back-end e as suas tecnologias
24	15/05/2025	3	47	Momento 2	Prática	APIs REST
25	22/05/2025	3	50	Momento 1	Teórica	Arquitetura em Camadas (Camada de Acesso à Dados; Camada de Serviços; Camada de Controle)
26	22/05/2025	3	53	Momento 2	Prática	Entidades
27	29/05/2025	3	56	Momento 1	Teórica	Casos de Uso
28	29/05/2025	3	59	Momento 2	Prática	Casos de Uso
29	05/06/2025	3	62	Momento 1	Teórica	Repository Pattern
30	05/06/2025	3	65	Momento 2	Prática	Repository Pattern
31	12/06/2025	3	68	Momento 1	Teórica	Controllers
32	12/06/2025	3	71	Momento 2	Prática	Controllers
33	19/06/2025	3	74	Momento 1	Prática	Consumo de API no front-end
34	19/06/2025	3	77	Momento 2	Teórica	CORS e segurança básica (JWT e middleware)
35	26/06/2025	3	80	Momento 1	Avaliação	Avaliação (projeto)
36	26/06/2025	3	83	Momento 2	Avaliação	Avaliação (projeto)
37	03/07/2025	3	86	Momento 1	Avaliação	Avaliação (projeto)
38	03/07/2025	3	89	Momento 2	Avaliação	Avaliação (projeto)
39	10/07/2025	3	92	Momento 1	Avaliação final	Avaliação final
40	10/07/2025	3	95	Momento 2	Avaliação final	Avaliação final
41	17/07/2025	3	98			
42	17/07/2025	3	101			