

PLANO DE ENSINO

CURSO SISTEMA WEB	MÓDULO	Componente Curricular Sigla	
Sistema WEB Cadastro de Patrimônio			
COMPONENTE CURRICULAR	AULAS PREVISTAS	DOCENTE	TURMA(S)
Sistema Web – Cadastro de Patrimônio	20 a 30 minutos	LUCAS PAIVA	
UNIDADE DE COMPETÊNCIA	OBJETIVO		
	O curso de Sistema Web – Cadastro de Patrimônio tem por objetivo o desenvolvimento de competências relativas à análise, modelagem, desenvolvimento e manutenção de sistemas web, utilizando arquitetura API REST, com separação entre back-end e front-end, aplicando conceitos de CRUD, relacionamento entre entidades, responsividade, organização de código e boas práticas de mercado.		

CAPACIDADES TÉCNICAS E CIENTÍFICAS

1. Compreender os conceitos de sistemas web e arquitetura cliente-servidor
2. Identificar os princípios da arquitetura API REST
3. Desenvolver aplicações web utilizando CRUD (Create, Read, Update, Delete)
4. Implementar relacionamento entre entidades (Patrimônio e Setor)
5. Utilizar banco de dados relacional em aplicações web
6. Aplicar conceitos de responsividade e experiência do usuário
7. Separar responsabilidades entre front-end e back-end

CAPACIDADES SOCIAIS, ORGANIZATIVAS E METODOLÓGICAS.

1. Demonstrar autogestão, organização e responsabilidade no desenvolvimento do projeto
2. Demonstrar raciocínio lógico na resolução de problemas computacionais
3. Trabalhar de forma estruturada seguindo padrões de arquitetura de software
4. Comunicar de forma técnica as decisões de desenvolvimento

CONHECIMENTOS

1. Fundamentos de Sistemas Web

- 1.1 Conceitos de aplicações web
- 1.2 Arquitetura cliente-servidor
- 1.3 HTTP e HTTPS
- 1.4 Métodos HTTP (GET, POST, PUT, PATCH, DELETE)

2. Arquitetura de Software

- 2.1 Conceito de API REST
- 2.2 Padrão de arquitetura MVC
- 2.3 Separação entre front-end e back-end

3. Banco de Dados

Página 2 de 10

- 3.1 Conceitos de banco de dados relacional
- 3.2 Modelagem de entidades
- 3.3 Relacionamento entre tabelas
- 3.4 Integridade referencial

4. Desenvolvimento Back-end

- 4.1 Criação de rotas REST
- 4.2 Validações de dados
- 4.3 Tratamento de erros
- 4.4 Status HTTP

5. Desenvolvimento Front-end

- 5.1 HTML e CSS
- 5.2 Responsividade com framework CSS
- 5.3 Consumo de API via JavaScript
- 5.4 Formulários e listagens

ATIVIDADE	
SITUAÇÃO DE APRENDIZAGEM - Formativa	
	TEXTO
	<p>Contextualização: Você foi contratado como desenvolvedor júnior por uma empresa que necessita organizar e controlar seus bens patrimoniais. A empresa precisa de um sistema web para cadastrar setores e patrimônios, permitindo o controle, edição e exclusão dessas informações.</p>
	<p>Desafio: Desenvolver um sistema web de Cadastro de Patrimônio, utilizando arquitetura API REST, onde seja possível:</p> <ul style="list-style-type: none">• Cadastrar setores• Cadastrar patrimônios vinculados a setores• Listar, editar e excluir registros
	<p>Entregas:</p> <ul style="list-style-type: none">• Sistema funcional executando localmente• Código organizado com separação entre front-end e back-end• Demonstração prática do funcionamento do sistema

INSTRUMENTO DE REGISTRO DE AVALIAÇÃO FORMATIVA

Natureza dos Critérios	Fundamentos Técnicos e Científicos ou Capacidades Técnicas	Critérios de avaliação			Alunos						
			Crítico	Desejável							
		0	NÃO Atingiu	1	Atingiu						
		F	Formativa	S	Somativa						
Competências Técnicas	1. Compreender os conceitos de sistemas web e arquitetura cliente-servidor	<p>Identificou corretamente os métodos HTTP (GET, POST, PUT e DELETE) e sua finalidade dentro do sistema.</p> <p>Explicou corretamente o fluxo cliente-servidor, descrevendo a comunicação entre front-end e back-end.</p>			F						
	2. Identificar os princípios da arquitetura API REST	<p>Reconheceu a API REST como responsável pela comunicação via HTTP e troca de dados em formato JSON.</p> <p>Justificou a escolha da arquitetura API REST em relação a outros padrões, como MVC.</p>			F						
	3. Desenvolver aplicações web utilizando CRUD (Create, Read, Update, Delete)	<p>Implementou corretamente as operações de cadastro, listagem, edição e exclusão de dados.</p> <p>Demonstrou domínio dos métodos HTTP associados a cada operação do CRUD.</p>			F						
	4. Implementar relacionamento entre entidades (Patrimônio e Setor)	<p>Estabeleceu corretamente o relacionamento entre as entidades Patrimônio e Setor no banco de dados.</p> <p>Explicou o conceito de chave estrangeira e sua importância para a integridade dos dados.</p>			F						
	5. Utilizar banco de dados relacional em aplicações web	<p>Utilizou banco de dados relacional para persistência dos dados da aplicação.</p> <p>Demonstrou compreensão sobre migrations, integridade referencial e uso de ORM.</p>			F						
	6. Aplicar conceitos de responsividade e experiência do usuário	<p>Desenvolveu interface funcional e responsiva, adaptável a diferentes tamanhos de tela.</p> <p>Aplicou boas práticas de usabilidade, organização visual e clareza na interface.</p>			F						
	7. Separar responsabilidades entre front-end e back-end	<p>Separou corretamente as responsabilidades entre front-end (interface) e back-end (regras de negócio)..</p> <p>Explicou claramente o papel de cada camada da aplicação.</p>			F						
	1. Demonstrar autogestão	<p>Executou as tarefas propostas com autonomia.</p>			F						

Página 4 de 10

PLANO DE AULA		
CONHECIMENTOS	ESTRATÉGIAS DE EN-SINO	INTERVENÇÕES MEDIADORAS
<p>Alterar conforme os conhecimentos do plano de Curso</p> <ul style="list-style-type: none">• Sistemas Web• API REST• CRUD• Banco de dados• Front-end e Back-end	<p>Aula expositiva dialogada Demonstração prática em laboratório Desenvolvimento orientado de sistema web Uso de exemplos reais de mercado</p>	<p>1 - O que é uma API REST e por que utilizá-la? 2 - Qual a diferença entre back-end e front-end? 3 - Por que relacionar patrimônio a um setor? 4 - Como garantir a integridade dos dados?</p>

SITUAÇÃO DE APRENDIZAGEM - Somativa	
ATIVIDADE	SITUAÇÃO DE APRENDIZAGEM - Somativa
TEXTO	
<p>Contextualização: A empresa aprovou a primeira versão do sistema e solicitou a entrega final da aplicação.</p>	
<p>Desafio: Finalizar o sistema de Cadastro de Patrimônio, garantindo:</p> <ul style="list-style-type: none">• CRUD completo de setores e patrimônios• Relacionamento funcional entre as entidades• Interface responsiva	
<p>Entregas:</p> <ul style="list-style-type: none">• Sistema funcional• Apresentação do projeto explicando o funcionamento• Código documentado	

INSTRUMENTO DE REGISTRO DE AVALIAÇÃO SOMATIVA

Natureza dos Critérios	Fundamentos Técnicos e Científicos ou Capacidades Técnicas	Critérios de avaliação			Alunos						
			Crítico	Desejável							
		0	NÃO Atingiu	1	Atingiu						
		F	Formativa	S	Somativa						
Competências Técnicas	1. Compreender os conceitos de sistemas web e arquitetura cliente-servidor	<p>Identificou corretamente os métodos HTTP (GET, POST, PUT e DELETE) e sua finalidade dentro do sistema.</p> <p>Explicou corretamente o fluxo cliente-servidor, descrevendo a comunicação entre front-end e back-end.</p>			S						
	2. Identificar os princípios da arquitetura API REST	<p>Reconheceu a API REST como responsável pela comunicação via HTTP e troca de dados em formato JSON.</p> <p>Justificou a escolha da arquitetura API REST em relação a outros padrões, como MVC.</p>			S						
	3. Desenvolver aplicações web utilizando CRUD (Create, Read, Update, Delete)	<p>Implementou corretamente as operações de cadastro, listagem, edição e exclusão de dados.</p> <p>Demonstrou domínio dos métodos HTTP associados a cada operação do CRUD.</p>			S						
	4. Implementar relacionamento entre entidades (Patrimônio e Setor)	<p>Estabeleceu corretamente o relacionamento entre as entidades Patrimônio e Setor no banco de dados.</p> <p>Explicou o conceito de chave estrangeira e sua importância para a integridade dos dados.</p>			S						
	5. Utilizar banco de dados relacional em aplicações web	<p>Utilizou banco de dados relacional para persistência dos dados da aplicação.</p> <p>Demonstrou compreensão sobre migrations, integridade referencial e uso de ORM.</p>			S						
	6. Aplicar conceitos de responsividade e experiência do usuário	<p>Desenvolveu interface funcional e responsiva, adaptável a diferentes tamanhos de tela.</p> <p>Aplicou boas práticas de usabilidade, organização visual e clareza na interface.</p>			S						
	7. Separar responsabilidades entre front-end e back-end	<p>Separou corretamente as responsabilidades entre front-end (interface) e back-end (regras de negócio)..</p> <p>Explicou claramente o papel de cada camada da aplicação.</p>			S						
	1. Demonstrar autogestão	Executou as tarefas propostas com autonomia.			S						

Página 7 de 10

Página 8 de 10

Instrumento de avaliação / exercícios / Atividades.

Componente curricular: Sistema Web – Cadastro de Patrimônio	Atividade: Desenvolvimento de sistema web com API REST	Nota:
Unidade: SENAI	Data: / /	
Aluno:	Nº:	Turma:

Listas de exercício
Atividades de fixação
Questionários
Atividades de recuperação

Página 9 de 10

NÍVEIS DE DESEMPENHO	NÍVEIS	NOTA
Atingiu todos os critérios críticos e desejáveis	1	100
Atingiu todos os critérios críticos e 2 desejáveis	2	85
Atingiu todos os critérios críticos e 1 desejáveis	3	70
Atingiu todos os critérios críticos	4	50
Atingiu 2 critérios críticos	5	30
Atingiu 1 critério crítico	6	15
Não atingiu nenhum critério crítico	7	0

NÍVEL MÍNIMO DE DESEMPENHO ESPERADO	4
-------------------------------------	----------

ELABORAÇÃO	DATA	APROVAÇÃO	DATA
Júlio	/ /		/ /

Página 10 de 10

ANEXOS:

Exercícios ou qualquer material utilizado no dia a dia:

CRONOGRAMA (deixar por ultimo) O cronograma deve ser atualizado a cada turma nova.

Cronograma e Acompanhamento de Distribuição de Aulas

Curso: Sistema WEB

Componente Curricular: Sistema WEB cadastro de Patrimônio

Turma:

Professor: Lucas Paiva

Programa Analítico	Dia/Mês/Ano (Previsto)	Dia/Mês/Ano (Realizado)
1. Contextualização do sistema e objetivos da aula	04/02/2026	04/02/2026
2. Conceitos de Sistemas Web e API REST 2.1 Métodos HTTP (GET, POST, PUT, DELETE)	04/02/2026	04/02/2026
3. Arquitetura do sistema (Back-end e Front-end)	04/02/2026	04/02/2026
4. Modelagem do banco de dados (Patrimônio x Setor)	04/02/2026	04/02/2026
5. Demonstração prática do sistema (CRUD)	04/02/2026	04/02/2026
6. Responsividade e usabilidade	04/02/2026	04/02/2026
Avaliações Formativa.	04/02/2026	04/02/2026
Avaliação Somativa.	04/02/2026	04/02/2026
Elaborado por: Lucas Paiva Data: 04/02/2026		