

## Plano de ensino

**Curso:** SIN-BAC - Bacharelado em Sistemas de Informação

**Turma:** BSIN231-7 - BSIN231-7

**Disciplina:** 7WEB204 - DESENVOLVIMENTO DE APLICAÇÕES PARA A WEB II

**Período letivo:** 2026/2

**Carga horária:** 36

**Professor:** 608270 - LINCONL NILO PEREIRA

### Ementa

1. Modelo cliente servidor, arquitetura em camadas, tecnologias e frameworks de programação para Web.

### Objetivo geral

1. Apresentar conceitos de ambientes de desenvolvimento para web e possibilitar que os discentes configurem, desenvolvam e publiquem suas aplicações com qualidade.

### Objetivo específico

1. Apresentar e definir o modelo cliente servidor.- Caracterizar e identificar a arquitetura em camadas.- Desenvolver a capacidade de avaliar e escolher diferentes tecnologias e frameworks para uso em programação web conforme necessidade

### Conteúdo programático

1. 1 Introdução
  - 1.1 Apresentação da disciplina
  - 1.2 Metodologia de ensino utilizada
  - 1.3 Avaliação
2. 2 Modelo cliente servidor
  - 2.1 Introdução e conceitos
3. 2.2 Servidores Web
  - 2.3 Servidores de páginas estáticas e dinâmicas
4. 2.4 Protocolos HTTP e WebSockets
  - 2.5 HTTPS e controle de acesso
5. 2.6 Arquitetura e infraestrutura de IoT
6. Avaliação 01
7. Recuperação 01
8. 3 Arquitetura em camadas
  - 3.1 Camada de apresentação
9. 3.2 Camada de Lógica de negócios
  - 3.3 Camada de acesso a dados
10. 3.4 Padrões
  - 3.4.1 MVC
  - 3.4.2 MVVM
11. 3.4.3 dotVVM
  - 3.4.4 Page Controller, Front Controller e Application Controller
  - 3.4.5 Template View, Transform View e Two Step View
12. Avaliação 02
13. Recuperação 02
14. 4 Tecnologias e frameworks de programação para web
  - 4.1 Aplicações orientadas a navegação ou a serviços
15. 4.2 Representação e visualização de dados
  - 4.3 Processamento de dados
  - 4.4 Escalabilidade
16. 4.5 Prototipação e documentação
  - 4.6 Hospedagem e publicação
  - 4.7 Desenvolvimento de aplicações
  - 4.8 Apresentação do Site multidisciplinar como base de conhecimento

### Metodologia

## Plano de ensino

### 1. Metodologia de Ensino

A disciplina será desenvolvida por meio de metodologias ativas de ensino e aprendizagem, buscando integrar teoria e prática, estimular a autonomia do estudante e favorecer o desenvolvimento das competências previstas no Projeto Pedagógico do Curso.

As atividades compreenderão:

- aulas expositivas dialogadas;
- resolução de problemas;
- estudos de caso;
- pesquisas orientadas;
- leitura, interpretação e discussão de artigos científicos e técnicos;
- análise crítica de aplicações Web;
- documentação de sistemas Web;
- desenvolvimento incremental de aplicações Web utilizando tecnologias atuais;
- atividades práticas individuais e em grupo;
- seminários e apresentações técnicas;
- utilização de metodologias colaborativas e aprendizagem baseada em projetos (Project Based Learning - PBL), quando pertinente.

Serão disponibilizados materiais complementares ao longo do semestre, tais como:

- artigos científicos;
- Simuladores;
- vídeos;
- e-books;
- podcasts;
- tutoriais;
- documentação técnica;
- repositórios de código;
- demais recursos digitais relacionados aos conteúdos da disciplina.

Todo o material didático será disponibilizado por meio da plataforma Moodle, que constituirá o ambiente oficial da disciplina para disponibilização de conteúdos, atividades, avisos, materiais de apoio e acompanhamento do desempenho acadêmico.

Os períodos destinados ao atendimento individualizado ocorrerão mediante agendamento, às sextas-feiras, das 16h00 às 19h00, conforme disponibilidade do professor e do estudante.

Em consonância com a Instrução Normativa nº 006/2026 da PROEN/UDESC, a avaliação da aprendizagem possuirá caráter contínuo, processual e formativo, permitindo o acompanhamento permanente do desenvolvimento dos estudantes e a identificação de dificuldades ao longo do semestre, possibilitando intervenções pedagógicas e estratégias de recuperação da aprendizagem durante todo o período letivo.

#### Recuperação da Aprendizagem

Em conformidade com a Instrução Normativa nº 006/2026, a recuperação da aprendizagem ocorrerá durante o próprio semestre letivo, simultaneamente ao desenvolvimento da disciplina, não havendo prova final.

Serão previstas três oportunidades de recuperação:

- Recuperação da Avaliação 1;
- Recuperação da Avaliação 2;
- Recuperação do Trabalho Final.

Cada recuperação abordará os conteúdos, competências e habilidades não alcançados pelo estudante no respectivo instrumento avaliativo.

#### Estratégia adotada

A disciplina adotará a estratégia prevista no Art. 10, inciso I, da Instrução Normativa nº 006/2026:

A nota obtida na recuperação substituirá a nota da avaliação correspondente quando for superior, permanecendo sempre registrada a maior nota obtida pelo estudante. Em nenhuma hipótese a recuperação poderá reduzir a nota anteriormente alcançada.

Dessa forma:

- cada avaliação possuirá sua respectiva recuperação;
- a recuperação ocorrerá antes do encerramento do semestre;
- somente substituirá a nota quando resultar em desempenho superior;
- caso a nota da recuperação seja inferior, será mantida a maior nota obtida anteriormente;
- não haverá exame final, conforme regulamentação institucional vigente.

## Sistema de avaliação

### 1. SISTEMA DE AVALIAÇÃO DA APRENDIZAGEM

A avaliação da aprendizagem será realizada de forma contínua, processual, diagnóstica e formativa, em consonância com a Instrução Normativa PROEN nº 006/2026, considerando tanto a compreensão dos fundamentos teóricos quanto a capacidade de aplicação prática dos conhecimentos na análise, no projeto e na implementação de soluções computacionais.

Para fins de organização pedagógica e acompanhamento da aprendizagem, o semestre letivo será estruturado em três módulos avaliativos.

Os Módulos 1 e 2 serão concluídos com avaliações escritas individuais. O Módulo 3 será desenvolvido por meio de um Trabalho

## Plano de ensino

Prático Final, permitindo ao estudante integrar e aplicar os conhecimentos construídos ao longo da disciplina.

### 1. Organização das avaliações

Módulo Instrumento principal Recuperação da aprendizagem

Módulo 1 Avaliação Escrita 1 Mapa mental + Lista de exercícios + Avaliação oral

Módulo 2 Avaliação Escrita 2 Mapa mental + Lista de exercícios + Avaliação oral

Módulo 3 Trabalho Prático Final Correção/melhoria do trabalho + Avaliação oral

Os três módulos terão o mesmo peso na composição da média semestral, permitindo equilíbrio entre a avaliação dos fundamentos teóricos e a aplicação prática dos conhecimentos.

### 2. Módulo 1 - Avaliação Escrita 1

O primeiro módulo será concluído com uma avaliação escrita, individual e presencial, abrangendo os conteúdos, conceitos, algoritmos e atividades práticas desenvolvidos no respectivo período.

A avaliação buscará verificar a capacidade do estudante de:

compreender os fundamentos relacionados à organização e às estruturas de dados;  
interpretar problemas computacionais;  
identificar estruturas adequadas para diferentes situações;  
interpretar algoritmos e trechos de código;  
elaborar soluções algorítmicas;  
relacionar os conceitos teóricos às aplicações computacionais;  
justificar tecnicamente as soluções apresentadas.  
Recuperação do Módulo 1

O processo de recuperação será constituído por:

Mapa mental - 20% da recuperação: síntese visual e conceitual dos conteúdos essenciais do módulo;  
Lista de exercícios - 40% da recuperação: resolução de problemas teóricos e práticos;  
Avaliação oral - 40% da recuperação: explicação individual dos conceitos e das soluções desenvolvidas.

### 3. Módulo 2 - Avaliação Escrita 2

O segundo módulo também será concluído com uma avaliação escrita, individual e presencial, contemplando os conteúdos desenvolvidos no período e podendo integrar conhecimentos anteriormente trabalhados quando necessários à resolução dos problemas propostos.

Serão avaliadas competências relacionadas à:

manipulação e organização de dados;  
utilização de arquivos e registros;  
aplicação de técnicas de hashing;  
implementação e interpretação de algoritmos;  
utilização de métodos de ordenação;  
utilização de algoritmos de busca;  
análise comparativa de diferentes soluções computacionais;  
avaliação de eficiência e complexidade dos algoritmos.  
Recuperação do Módulo 2

O processo de recuperação será constituído por:

Mapa mental - 20% da recuperação;  
Lista de exercícios - 40% da recuperação;  
Avaliação oral - 40% da recuperação.

A avaliação oral permitirá verificar a compreensão efetiva dos conteúdos e das soluções apresentadas pelo estudante.

### 4. Módulo 3 - Trabalho Prático Final

O terceiro módulo será avaliado por meio de um Trabalho Prático Final, no qual os estudantes deverão desenvolver uma solução computacional capaz de integrar conhecimentos adquiridos ao longo da disciplina.

O trabalho terá caráter integrador e aplicado, aproximando os conteúdos acadêmicos de situações reais da área de Computação e Sistemas de Informação.

O projeto deverá envolver, conforme a proposta definida pelo professor, diferentes conteúdos desenvolvidos durante o semestre, tais como:

manipulação de arquivos;

## Plano de ensino

registros e organização de dados;  
estruturas de dados;  
técnicas de hashing;  
algoritmos de ordenação;  
algoritmos de busca;  
análise de desempenho;  
análise de complexidade computacional;  
modularização e organização do código;  
documentação da solução desenvolvida;  
utilização adequada de ferramentas de desenvolvimento;  
utilização de sistema de controle de versões, quando aplicável.

O estudante deverá ser capaz não apenas de implementar uma solução funcional, mas também de explicar as decisões tomadas durante o desenvolvimento e justificar tecnicamente a escolha das estruturas de dados e dos algoritmos utilizados.

### 5. Critérios de avaliação do Trabalho Prático Final

O Trabalho Prático Final será avaliado considerando os seguintes critérios:

#### Critério Peso

Análise e compreensão do problema 10%  
Aplicação adequada das estruturas de dados 20%  
Implementação dos algoritmos e funcionamento da solução 25%  
Utilização de busca, ordenação, hashing, arquivos ou outros conteúdos pertinentes 15%  
Análise de eficiência e complexidade computacional 10%  
Organização, qualidade e documentação do código 10%  
Apresentação e defesa da solução desenvolvida 10%  
Total 100%

A apresentação do trabalho poderá incluir demonstração prática do sistema, explicação do código-fonte, apresentação dos algoritmos utilizados e questionamentos individuais realizados pelo professor.

### 6. Recuperação do Módulo 3 - Trabalho Prático Final

Caso o estudante não alcance os objetivos de aprendizagem previstos para o terceiro módulo, será oportunizada a recuperação do Trabalho Prático Final.

A recuperação deverá priorizar a correção das dificuldades identificadas durante a avaliação inicial.

O estudante deverá realizar:

- Correção e melhoria do Trabalho Prático Final: revisão da solução inicialmente apresentada, corrigindo problemas conceituais, algorítmicos, estruturais ou de implementação indicados pelo professor.
- Avaliação oral individual: apresentação da versão corrigida e questionamento sobre os conceitos, algoritmos, estruturas de dados e decisões utilizadas na implementação.

Dessa forma, a recuperação não consistirá simplesmente na repetição do trabalho, mas na oportunidade de revisar, corrigir e aperfeiçoar a solução, demonstrando evolução da aprendizagem.

### 7. Composição da média semestral

A avaliação da disciplina será composta por três módulos com pesos equivalentes:

Módulo Instrumento Peso  
Módulo 1 Avaliação Escrita 1 33,33%  
Módulo 2 Avaliação Escrita 2 33,33%  
Módulo 3 Trabalho Prático Final 33,34%

Total 100%

A média semestral será calculada pela seguinte expressão:

Média Semestral = (Módulo 1 + Módulo 2 + Módulo 3) ÷ 3

O resultado das atividades de recuperação será considerado conforme os critérios acadêmicos institucionais estabelecidos pela instituição e pela Instrução Normativa PROEN nº 006/2026.

### 8. Avaliação contínua e processual

## Plano de ensino

Além dos instrumentos formais de avaliação, o professor realizará acompanhamento contínuo da aprendizagem durante todo o semestre.

Serão utilizados como instrumentos diagnósticos e formativos:

resolução de exercícios;  
atividades práticas em laboratório;  
implementação de algoritmos;  
estudos de caso;  
atividades de PBL;  
atividades individuais e em grupo;  
análise de algoritmos;  
discussões orientadas;  
leitura de documentação técnica e artigos;  
exercícios de análise de complexidade;  
desenvolvimento incremental do Trabalho Prático Final.

Essas atividades permitirão identificar dificuldades antes das avaliações formais e orientar intervenções pedagógicas e ações de recuperação.

### 9. Atividades Curriculares de Extensão - ACE

As Atividades Curriculares de Extensão (ACE) serão desenvolvidas ao longo do semestre de forma articulada aos conteúdos da disciplina.

Sempre que pedagogicamente pertinente, o Trabalho Prático Final poderá ser integrado às ACE, possibilitando que os estudantes desenvolvam soluções computacionais destinadas a problemas ou demandas reais da comunidade.

Essa integração buscará aproximar:

Universidade ? Conhecimento ? Tecnologia ? Comunidade

As atividades extensionistas deverão estimular a aplicação dos conhecimentos de estruturas de dados e algoritmos na resolução de problemas concretos, fortalecendo simultaneamente as competências técnicas, profissionais, sociais e cidadãs dos estudantes.

### 10. Critérios gerais de avaliação da aprendizagem

Em todos os módulos serão considerados, de acordo com a natureza da atividade:

domínio dos conceitos fundamentais;  
raciocínio lógico e algorítmico;  
capacidade de resolução de problemas;  
escolha adequada das estruturas de dados;  
correção dos algoritmos;  
qualidade das implementações;  
capacidade de analisar desempenho e complexidade;  
organização e documentação das soluções;  
autonomia no desenvolvimento das atividades;  
capacidade de relacionar teoria e prática;  
argumentação e justificativa técnica;  
evolução da aprendizagem durante o semestre.

O sistema avaliativo buscará, portanto, articular avaliação teórica, prática, recuperação contínua e acompanhamento formativo, culminando no Trabalho Prático Final, no qual o estudante deverá demonstrar a capacidade de integrar os conhecimentos adquiridos durante a disciplina na construção de uma solução computacional fundamentada, eficiente e tecnicamente justificável.

### Bibliografia básica

1. ALVES, William P. Projetos de Sistemas Web Conceitos, Estruturas, Criação de Banco de dados e Ferramentas de Desenvolvimento. Editora Saraiva, 2015. E-book. ISBN 9788536532462. Disponível em: <https://app.minhabiblioteca.com.br/#/books/9788536532462/>. Acesso em: 14 fev. 2024  
OLIVEIRA, Cláudio Luís V.; ZANETTI, Humberto Augusto P. JAVASCRIPT DESCOMPLICADO - PROGRAMAÇÃO PARA WEB, IOT E DISPOSITIVOS MÓVEIS. Editora Saraiva, 2020. E-book. ISBN 9788536533100. Disponível em: <https://app.minhabiblioteca.com.br/#/books/9788536533100/>. Acesso em: 14 fev. 2024  
KALBACH, J. Design de Navegação Web: Grupo A, 2009. 9788577805310  
Disponível em: <https://app.minhabiblioteca.com.br/#/books/9788577805310/>

## Plano de ensino

Acesso em 11/08/2023

### *Bibliografia complementar*

1. ALVES, William P. Java para Web - Desenvolvimento de Aplicações. Editora Saraiva, 2015. E-book. ISBN 9788536519357. Disponível em: <https://app.minhabiblioteca.com.br/#/books/9788536519357/>. Acesso em: 14 fev. 2024.
- MASCHIETTO, Luís G.; VIEIRA, Anderson L N.; TORRES, Fernando E.; et al. Arquitetura e Infraestrutura de IoT. Grupo A, 2021. E-book. ISBN 9786556901947. Disponível em: <https://app.minhabiblioteca.com.br/#/books/9786556901947/>. Acesso em: 14 fev. 2024.
- RODRIGUES, Thiago N.; SILVA, Lídia P C.; NEUMANN, Fabiano B.; et al. Integração de Aplicações. Grupo A, 2020. E-book. ISBN 9786556900216. Disponível em: <https://app.minhabiblioteca.com.br/#/books/9786556900216/>. Acesso em: 14 fev. 2024.
- MILETTO, Evandro Manara; BERTAGNOLLI, Sílvia de Castro. Desenvolvimento de software II: introdução ao desenvolvimento WEB com HTML, CSS, Javascript e PHP. Porto Alegre: Bookman, 2014.- Referência tradicional em cursos de desenvolvimento web no Brasil, abordando linguagens principais e integração com backend, ideal para aprofundar temas de aplicações web completas em uma disciplina II.
- ZELDMAN, Jeffrey; MARCOTTE, Ethan. Designing with Web Standards. 3. ed. New Riders, 2009.- Livro clássico sobre desenvolvimento web com foco em padrões web (HTML/CSS/JavaScript e boas práticas), amplamente usado como referência em cursos avançados de desenvolvimento web em instituições de ensino