



## Plano de Ensino

---

### 1) Identificação

**Disciplina:** INE5646 - Programação para Web  
**Turma(s):** 03238A  
**Carga horária:** 72 horas-aula      Teóricas: 36      Práticas: 36  
**Período:** 1º semestre de 2025

### 2) Cursos

- Sistemas de Informação (238)

### 3) Requisitos

- Sistemas de Informação (238) (currículo: 20001)
  - INE5609 - Estruturas de Dados
- Sistemas de Informação (238) (currículo: 20111)
  - INE5605 - Desenvolvimento de Sistemas Orientados a Objetos I

### 4) Professores

- Wyllian Bezerra da Silva (wyllian.bs@ufsc.br)

### 5) Ementa

Arquiteturas. Serviços. Protocolos de comunicação entre cliente e servidor e entre servidores. Representação de dados no cliente. Acesso a bases de dados. Escalabilidade. Linguagens de modelagem da interface com o usuário. Linguagens de programação no cliente e no servidor. Segurança. Instalação e configuração de servidores. Prática de programação.

### 6) Objetivos

**Geral:** Capacitar os estudantes para que compreendam os fundamentos conceituais e tecnológicos bem como as linguagens pertinentes ao desenvolvimento de aplicações para web.

**Específicos:**

- Apresentar as principais tecnologias utilizadas no desenvolvimento de aplicações exemplo.
- Possibilitar que os estudantes pratiquem o desenvolvimento de aplicações para web.
- Possibilitar que os estudantes aprendam a instalar, configurar e administrar servidores web.

### 7) Conteúdo Programático

#### 7.1) Parte 1 - Linguagens para Desenvolvimento Web [14 horas-aula]

- Visualização de dados
  - HTML
  - CSS
  - Frameworks
  - Pré-processadores
- Representação de dados
  - XML
  - JSON
- Programação
  - Visão Geral sobre PHP, Java, Python e Ruby
  - JavaScript
    - Elementos de Programação Orientada a Objetos
    - Elementos de Programação Funcional
    - Técnicas de Processamento Assíncrono: Callback, Promise, Observable, Async/Await
    - Modularização
    - Bibliotecas
    - Linguagens Alternativas: TypeScript e Elm

- 7.2) Parte II - Modelagem de Aplicações para Web [42 horas-aula]
- Modelo Cliente-Servidor
    - Protocolos de Comunicação
      - HTTP
      - Websocket
    - Tecnologias de Comunicação
      - Ajax
      - SSE
  - Finalidade
    - Aplicações orientadas a serviço
      - Protocolo SOAP
      - Estilo Arquitetural RESTful: REST, GraphQL e Protobuf
    - Aplicações orientadas a página
      - Múltiplas Páginas
      - Página Única
      - Progressivas
      - Browser APIs
  - Arquiteturas
    - Monolíticas
      - Em camadas
    - Distribuídas
      - CQRS
      - Microserviços
  - Aspectos de Segurança
    - Transmissão da Informação
    - Autenticação do Usuário
    - Autorização para Acesso aos Recursos
    - Protocolo OAuth 2
      - OpenID Connect
      - JSON Web Token
    - Protocolo HTTP
      - Política de Mesma Origem
      - Cabeçalhos de Segurança
      - Filtros
      - Cookies
  - Operações de E/S Escaláveis
    - Acesso a webserviços
    - Acesso a bases de dados
  - Técnicas de Desenvolvimento
    - No lado servidor
      - Domain Driven Design
      - Event Driven Design
      - Event Sourcing
      - Programação Reativa
    - No lado cliente
      - Componentes Web
      - Design Responsivo
      - Material Design
      - Gerenciamento de Estado
      - Processamento de Fluxos de Dados
      - MVC
      - Fluxo Unidirecional: Redux, MVI
- 7.3) Parte III – Estudo de Um Servidor de Aplicações [16 horas-aula]
- Servidor Node.js
    - Características:
      - Ferramenta npm
      - Biblioteca libuv
      - Event Loop
      - Streams
      - Bibliotecas
    - Instalação, configuração e gerenciamento

## 8) Metodologia

Os tópicos serão abordados por meio de aulas expositivas e demonstrações práticas do conteúdo. Para isso, serão usados slides, notas de aula, tutoriais, documentações e exemplos que serão disponibilizados aos alunos para estudo posterior, assim como outros materiais complementares.

Durante as aulas serão propostos exercícios, problemas e discussões com o intuito de reforçar o conteúdo ministrado e sanar eventuais dúvidas.

Na disciplina também é previsto o desenvolvimento de um projeto web que contempla os principais temas da ementa e que deverá ser desenvolvido ao longo do semestre.

Em caso de dúvidas sobre o conteúdo da disciplina, o(a) discente poderá recorrer ao atendimento disponibilizado pelo docente, cujo horário está agendado no Moodle. Comunicação e informações gerais, ensalamento ou troca de sala/laboratório, exercícios, tarefas, material de apoio, slides, documentos etc. serão disponibilizados no Moodle da disciplina.

## 9) Avaliação

A verificação do rendimento escolar compreenderá frequência e aproveitamento de estudos, os quais deverão ser atingidos conjuntamente. Será obrigatória a frequência às atividades maior ou igual a 75% (Frequência Suficiente - FS), cuja frequência será registrada até a última semana do semestre, estará reprovado o(a) discente com frequência menor do que 75% (Frequência Insuficiente - FI).

O processo de avaliação será realizado de forma progressiva ao decorrer da disciplina. Serão propostas atividades avaliadas (AA), com frequência aproximadamente semanal ao longo do semestre, que podem incluir pequenos questionários, participação em fóruns, apresentação de código, entre outras. Além disso, haverá duas provas (P1 e P2) e um projeto web (PW) entregue ao final do semestre. O item de nota PW será composto pela parte escrita (EP) e apresentação (AP). As provas serão individuais, sem consulta e divididas em questões teóricas e práticas. As questões práticas envolverão temas que pressupõem que o estudante domine linguagens, frameworks e bibliotecas utilizados nos exercícios ao longo do semestre.

A média final (MF) e nota final (NF) de cada estudante é calculada pela expressão a seguir, com base nas atividades avaliadas (AA), provas (P1 e P2) e na nota do projeto web (EP e AP):

$$NF = MF = 0,2*(AA + P1 + P2 + EP + AP).$$

O arredondamento da UFSC será aplicado sobre os itens de nota **MF, NF, AA, P1, P2, EP, AP**.

Não é prevista atividade de recuperação para esta turma, nos termos previstos no art. 70, parágrafo 2o, da Resolução 17/CUn/97, uma vez que cumpre pelo menos um dos seguintes requisitos:

- ter pelo menos 50% de carga prática;
- ter pelo menos 50% do peso da média final originado de trabalho prático;
- ter a inadequação da aplicação de avaliação de recuperação reconhecida pelo colegiado do curso, a partir da avaliação de solicitação fundamentada de dispensa de avaliação de recuperação, encaminhada pelo(s) professor(es) autor(es) do respectivo plano de ensino, para disciplinas com carga prática prevista no programa da disciplina, com nota de trabalho prático considerada no cálculo da média final e que não tenham cumprido um dos requisitos anteriores.

## 10) Cronograma

As avaliações estão previstas nos períodos a seguir, os quais estão sujeitos a alterações.

- Prova 1 (**P1**): entre a 8ª e 9ª semana.
- Prova 2 (**P2**): entre a 15ª e 16ª semana.
- Projeto Web (**EP e AP**): entrega na 17ª semana, com apresentações entre a 17ª e 18ª semana.

## 11) Bibliografia Básica

- LINDLEY, Cody. Front-end Developer Handbook 2019. Frontend Masters, 2019. Disponível em <https://raw.githubusercontent.com/FrontendMasters/front-end-handbook-2019/master/exports/Front-end%20Developer%20Handbook%202019.pdf>
- MARDAN, A. Practical Node.js, 2nd ed, Apress, 2018. Disponível em [https://itbook.download/topic/Practical\\_Node\\_js\\_2nd\\_Edition](https://itbook.download/topic/Practical_Node_js_2nd_Edition)
- KALBACH, James. Design de navegação web: otimizando a experiência do usuário. Porto Alegre: Bookman, 2009. xiii, 427p. ISBN 9788577804917.
- SEBESTA, Robert W. Programming the world wide web 2009. 5th ed. Boston: Addison Wesley, c2010. xviii, 734p. ISBN 9780136076636.
- SILVA, Maurício Samy. HTML 5: a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011. 320 p. ISBN 9788575222614.

- EVANS, Eric. Domain-driven design: atacando as complexidades no coração do software. Rio de Janeiro: Alta Books, 2009. xxviii, 499 p. ISBN 9788576083603.

## **12) Bibliografia Complementar**

- AMUNDSEN, Michael. Building hypermedia APIs with HTML5 and Node. Beijing: O'Reilly, 2011. xvii, 219 p. ISBN 9781449306571.
- BONÉR, Jonas. Reactive Microservices Architecture. Sebastopol: O'Reilly Media, Inc, 2017. 84 p. ISBN 9781491994368.
- BURNS, Brendam. Designin Distributed Systems. Sebastopol: O'Reilly Media, Inc, 2017. 160 p. ISBN 9781491983638.
- FREDERICK, Gail Rahn; LAL, Rajesh. Dominando o desenvolvimento web para smartphone: construindo aplicativos baseados em JavaScript, CSS, HTML e Ajax para iPhone, Android, Palm Pre, BlackBerry, Windows Mobile e Nokia S60. Rio de Janeiro: Alta Books, 2011. xiii, 344 p. ISBN 788576085140.
- GHOSH, Debasish. Functional and Reactive Domain Modeling. New York: Manning Publications Co, 2017. 322p, ISBN 9781617292248.
- HERRON, David. Node Web Development. Birmingham: Packt Publishing Ltd, 2013. 230p. ISBN 9781782163305.