



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
Centro de Ciências, Tecnologias e Saúde  
Departamento de Computação  
PLANO DE ENSINO

SEMESTRE 2018.1

I. IDENTIFICAÇÃO DA DISCIPLINA:

CÓDIGO	NOME DA DISCIPLINA	Nº DE HORAS-AULA SEMANAIS		TOTAL DE HORAS-AULA SEMESTRAIS
		TEÓRICAS	PRÁTICAS	
DEC7551	Tópicos Especiais I - Desenvolvimento aplicações móveis	0	4	72

TURMAS TEÓRICAS	HORÁRIO	MÓDULO
	TURMAS PRÁTICAS	Presencial
	4	

II. PROFESSOR(ES) MINISTRANTE(S)

Prof. Fábio Rodrigues de la Rocha, Dr.  
Fabio.rocha.ufsc@gmail.com

III. PRÉ-REQUISITO(S)

CÓDIGO	NOME DA DISCIPLINA
	Esta disciplina não possui pré-requisitos

IV. CURSO(S) PARA O(S) QUAL(IS) A DISCIPLINA É OFERECIDA

Engenharia de Computação

V. JUSTIFICATIVA

Esta disciplina explora arquitetura de clientes-servidores onde os clientes são aplicações móveis rodando em smartphones e os servidores são servidores de aplicação rodando em alguma máquina na Internet.

VI. EMENTA

Desenvolvimento de aplicações móveis híbridas usando HTML, JavaScript e CSS (front-end). Ferramentas de desenvolvimento. Implementação do lado servidor usando NodeJS. (back-end), Mensagens AJAX, websockets, acesso à banco de dados, Desenvolvimento de uma aplicação completa

VII. OBJETIVOS

Objetivo Geral:

- Capacitar o aluno a projetar um sistema composto por aplicações clientes (smartphones) e aplicações servidoras de aplicações.
- Capacitar a desenvolver o código interno de cada um destes elementos e instalar nos servidores
- Desenvolver código em NodeJs, compreender o conceito de eventos, callbacks
- Compreender como trocar informações entre as entidades e as questões envolvidas
- Capacitar o aluno a desenvolver uma aplicação completa para testar o funcionamento e medir desempenho

VIII. CONTEÚDO PROGRAMÁTICO

**UNIDADE 1:** Arquitetura típica de um servidor WEB, exemplos de servidores: Apache, nginx, lighttpd.  
- funcionamento de um servidor WEB, instalação e configuração de um servidor no Linux  
- requisições GET, POST, linguagens server side  
- HTML, CSS, Javascript

## UNIDADE 2: NodeJS

- O que é NodeJS, vantagens e desvantagens
- Instalação do NodeJs
- Programação síncrona e assíncrona, event loop, event emitter uso de callbacks, orientação a objetos, eventos
- Javascript: variáveis, comentários, funções, objetos, vetores
- NPM, pacotes, express
- exemplos de código
- criando um servidor
- websockets
- Criando um servidor usando websockets
- Criando um servidor usando AJAX
- Banco de dados MongoDB, BD orientado a documento, operações mais comuns
- Gerenciando aplicações com o PM2

## UNIDADE 3: Aplicações móveis

- Aplicações híbridas
- Framework Cordova, plugins, criação de projeto
- Exemplos de aplicações cordova
- Desenvolvimento de uma aplicação cordova para acessar um servidor usando AJAX e websockets

## X. METODOLOGIA DE ENSINO / DESENVOLVIMENTO DO PROGRAMA

- Aulas teóricas: desenvolvidas em sala e com emprego de meios audiovisuais tais como transparências e apresentações sobre PC portátil de produção própria expostas com projetor. Todo o material didático estará disponível "a priori" para os alunos na página do professor: [fabiodelarocha.paginas.ufsc.br](http://fabiodelarocha.paginas.ufsc.br)

## X. METODOLOGIA E INSTRUMENTOS DE AVALIAÇÃO

A verificação do rendimento escolar compreenderá frequência e aproveitamento nos estudos, os quais deverão ser atingidos conjuntamente. Será obrigatória a frequência às atividades correspondentes a cada disciplina, no mínimo a 75% das mesmas (Frequência Suficiente - FS), ficando nela reprovado o aluno que não comparecer a mais de 25% das atividades (Frequência Insuficiente - FI).

Media final = (Trabalho T1 + Trabalho T2) /2

O aluno com Frequência Suficiente (FS) e média das notas de avaliações do semestre MF entre 3,0 e 5,5 NÃO terá direito a uma nova avaliação no final do semestre (REC) de acordo com o art.70, § 2º. A Nota Final (NF) será calculada por meio da média aritmética entre a média das notas das avaliações parciais. (Art. 70 e 71 da Res. nº 17/CUn/1997).

Ao aluno que não comparecer às atividades práticas ou não apresentar trabalhos no prazo estabelecido será atribuída nota 0 (zero). (Art. 70, § 4º da Res. nº 17/CUn/1997)

Observações:

Avaliação de recuperação

Não há avaliação de recuperação nas disciplinas de caráter prático que envolve atividades de laboratório. (Res.17/CUn/97).

**Nova oportunidade de realizar atividade avaliativa**

- *O aluno, que por motivo de força maior e plenamente justificado, deixar de realizar atividades avaliativas previstas no plano de ensino, deverá formalizar pedido à Chefia do Departamento de Ensino ao qual a disciplina pertence, dentro do prazo de 3 (três) dias úteis, apresentando documentação comprobatória. (Ver formulário)*

## XI. CRONOGRAMA TEÓRICO

AULA (semana)	DATA	ASSUNTO	
1ª	26/02/18	03/03/18	Introdução à disciplina, site da disciplina, plano de ensino,

			forma de avaliação.
2ª	05/03/18	10/03/18	Instalação e preparação do ambiente de trabalho, Linux, servidores WEB (nginx), node, npm. Unidade 1
3ª	12/03/18	17/03/18	Unidade 1
4ª	19/03/18	24/03/18	Unidade 2
5ª	26/03/18	31/03/18	Unidade 2
6ª	02/04/18	07/04/18	Unidade 2
7ª	09/04/18	14/04/18	Unidade 2
8ª	16/04/18	21/04/18	Unidade 2
9ª	23/04/18	28/04/18	Unidade 2
10ª	30/04/18	05/05/18	Unidade 2
11ª	07/05/18	12/05/18	Unidade 2
12ª	14/05/18	19/05/18	Unidade 2
13ª	21/05/18	26/05/18	Apresentação do Trabalho T1
14ª	28/05/18	02/06/18	Unidade 3
15ª	04/06/18	09/06/18	Unidade 3
16ª	11/06/18	16/06/18	Unidade 3
17ª	18/06/18	23/06/18	Unidade 3
18ª	25/06/18	30/06/18	Apresentação do Trabalho T2 fechamento da disciplina e publicação das notas

Obs.: Eventos:

- 12 a 15 de março de 2018 Semana acadêmica de Eng. de Computação

Obs: O calendário está sujeito a pequenos ajustes de acordo com as necessidades das atividades desenvolvidas.

## XII. Feriados previstos para o semestre 2018.1:

DATA	
30/03/18	Sexta-feira santa
31/03/18	Dia não letivo
03/04/18	Aniversário de Araranguá
21/04/18	Tiradentes
30/04/18	Dia não letivo
01/05/18	Dia do trabalhador
04/05/18	Dia da Padroeira de Araranguá
31/05/18	Corpus Christi
01/06/18	Dia não letivo
02/06/18	Dia não letivo

## XIII. BIBLIOGRAFIA BÁSICA

- [1] Construindo aplicações com NodeJS, Willian Bruno Moraes, 2015, novatec
- [2] Aprendendo a Desenvolver Aplicações Web. Semmy Purewal, 2014, novatec
- [3] The Node Craftsman Book. Manuel Kiessling, 2017, Packt

## XIV. BIBLIOGRAFIA COMPLEMENTAR:

- [4] Node.js Web Development, David Herron, 2016, Packt
- [5] <https://www.tutorialspoint.com/nodejs/>
- [6] Learning Node: Moving to the Server-Side, Shelley Powers, 2016, O'Reilly
- [7] Mobile App Development with Ionic, Revised Edition: Cross-Platform Apps with Ionic, Angular, and Cordova, Chris

Griffith, 2017, O'Reilly

[8] Apache Cordova 4 Programming (Mobile Programming), John M. Wargo, 2015, Addison-Wesley

#### XV. INFRAESTRUTURA E MATERIAS NECESSÁRIOS:

1. Espaço físico com mesas, cadeiras e tomadas em quantidades adequadas
2. Acesso à internet
3. Datashow que possa ser operado de forma segura, sem risco de acidentes
4. Computadores rodando Linux ou emulado

**Obs.:** A indisponibilidade de infraestrutura/materiais listados pode causar prejuízos ao processo pedagógico, inviabilizando tanto as atividades dos docentes como as dos alunos, podendo, ainda, acarretar em cancelamento de aulas em último caso.

\*\*\*Assinado digitalmente por \*\*\*  
Fábio Rodrigues de la Rocha  
2018.02.09 10:08:14 BRST SIAPE:  
1781774  
96300752020 fabio.rocha@gmail.com

Professor da Disciplina

/ / 2018

Aprovado pelo  
departamento em

/ / 2018

Eliane

Pozzebon:7792  
1313949

Assinado de forma digital  
por Eliane  
Pozzebon:77921313949  
Dados: 2018.05.14  
15:20:19 -03'00'

Aprovado pelo colegiado do curso  
de graduação em

/ / 2018