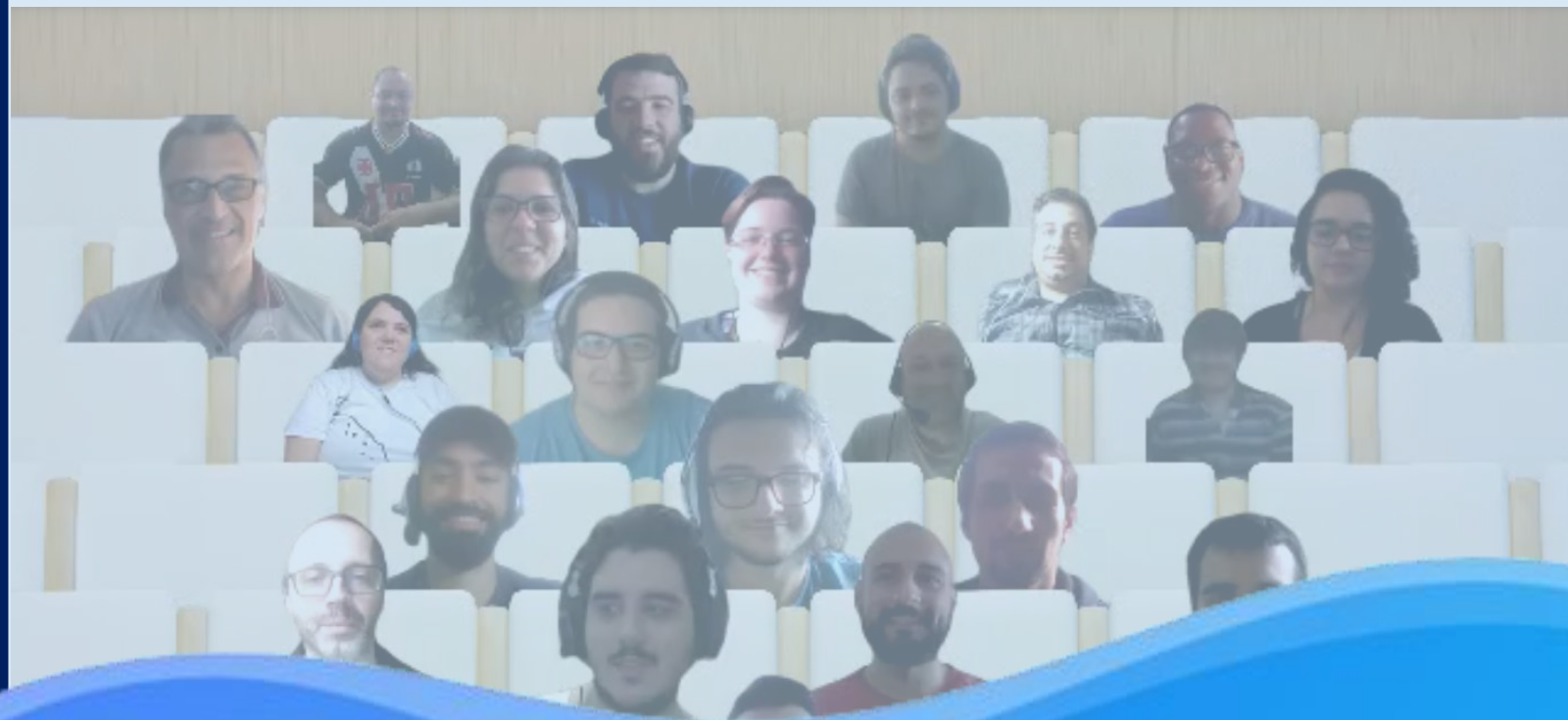


53 45 52 45 49 20 46 49 45 4c 20
 41 4f 53 20 50 52 45 43 45 49 54
 4f 53 20 44 41 20 48 4f 4e 52 41
 20 45 20 44 41 20 43 49 c3 8a 4e
 43 49 41 2c 20 50 52 4f 4d 4f 56
 45 4e 44 4f 20 4f 20 55 53 4f 20
 45 20 4f 20 44 45 53 45 4e 56 4f
 4c 56 49 4d 45 4e 54 4f 20 44 41
 20 49 4e 46 4f 52 4d c3 81 54 49
 43 41 20 45 4d 20 42 45 4e 45 46
 c3 8d 43 49 4f 20 44 4f 20 43 49
 44 41 44 c3 83 4f 20 45 20 44 41
 20 53 4f 43 49 45 44 41 44 45 2e

RESIDÊNCIA DE SOFTWARE

**CAPACITAR
TREINAR
EMPREGAR**

TRANSFORMAR



Conteúdo Programático Residência em TIC/Software
 PETRÓPOLIS/TERESÓPOLIS/NOVA FRIBURGO – 2021.1
 De 22/03 a 01/08

Disciplina	Dias	Horas
Lógica de Programação	9	54
Banco de Dados	7	42
Modelagem de sistemas	3	18
Programação Orientada a Objetos	14	84
FrontEnd Essencial	7	42
Desenvolvimento API Restful	14	84
Desenvolvimento Web	9	54
Desenvolvimento Mobile	11	66
Projeto Aplicado	11	66
Integração de Competência	3 1/2	20
TOTAL	89	530

Conteúdo	Horas
Mentoria de Carreira	17
Inglês Instrumental	16
Maratona de programação	60
Monitoria	96
Mentoria de projetos	20
Palestras e Workshops (2 horas em 8 sábados)	31
TOTAL	240

***134 dias corridos**

***92 dias úteis**

***19 semanas**

***770 horas de curso (aprox.)**

- Aulas da Residência de Software 2021.1 – de 22/03 a 29/07
- Maratona de Programação 30/07, 31/07 e 01/08
- Aulas ao vivo e online
- Aula Inaugural online – 18/03 (das 9h às 11h) – aberto ao público
- Horários das aulas na plataforma do Senai/Firjan: de 8h às 12h e das 13h às 15h (6 horas)
- Frequência de 75% é obrigatória em cada disciplina.
- Monitores atuarão para apoiar alunos nos exercícios e tirando dúvidas, três vezes por semana, na parte da tarde, após as aulas (de 15:15h às 17:15h).
- Às segundas e quartas, de 15:15h às 17:30h, alunos de Petrópolis terão aulas complementares de inglês e mentoria de carreira.
- Às segundas e terças, , de 15:15h às 17:30h, alunos de Teresópolis e Nova Friburgo terão aulas complementares de inglês e mentoria de carreira.
- Aos sábados, mediante agendamento, haverá aulas complementares, na parte da manhã e esporadicamente a tarde

- Compreender a lógica booleana (E, OU, NOT).
- Entender a árvore de decisão.
- Saber utilizar as estruturas de laço.
- Conhecer o conceito de recursividade.
- Entender a estrutura de dados (vetor, matriz, fila, pilha...).
- Compreender o conceito de variável e constante.
- Saber utilizar um sistema de controle de versão (GIT).

- Entender os conceitos de entidade, atributo e relacionamento.
- Saber mapear e projetar um banco de dados.
- Criar um banco de dados (DDL- Data Definition Language).
- Manter a integridade referencial do banco de dados (chave estrangeira).
- Manipular banco de dados (DML - Data Manipulation Language).
- Consultar banco de dados (DQL - Data Query Language).
- Consultar múltiplas tabelas de um banco de dados (junção de tabelas).
- Utilizar funções de agregação (soma, máximo, mínimo, média, etc).
- Fazer agrupamento.
- Criar índices para pesquisa no banco de dados.
- Entender o conceito de normalização de banco de dados.



- Conhecer a UML.
- Entender e elaborar o diagrama de Casos de Uso.
- Entender e elaborar o diagrama de Classes.
- Entender e elaborar o diagrama de Sequência.
- Entender e elaborar o diagrama de Atividades.
- Entender e elaborar o diagrama de Estados.
- Entender e elaborar o diagrama de Componente.



4. Programação Orientada a Objetos

- Compreender compilação, execução e entrada de dados (i/o, teclado, console).
- Dominar os identificadores, palavras chave, tipos e operadores (tipos primitivos, conversão, variáveis, hierarquia de operadores e parênteses).
- Saber utilizar as estruturas de programação (if, loop, switch, break, * Enum , métodos de classe, recursividade) e escopo de variáveis.
- Gerir objetos e classes: acesso, getters, setters, construtores, static.
- Gerir objetos e classes: herança, reescrita, polimorfismo, classes abstratas e Interfaces.
- Manipular data e hora.
- Saber utilizar as estruturas de dados (arrays, coleções).
- Realizar o tratamento de erros.
- Acessar um banco de dados a partir do backend.
- Empacotar aplicações, criar e utilizar bibliotecas.



- Conhecer os pilares de uma aplicação web, identificar a diferença entre HTML, CSS e JavaScript.
- Saber escrever um documento HTML com suas principais tags.
- Fazer um arquivo CSS com suas propriedades básicas: margin, width, height, padding, color, background, border e outros.
- Utilizar seletores do CSS, conhecendo suas especificidades.
- Saber fatiar um layout, criar uma tela entendendo como funcionam os principais displays: block, inline, none, flex e etc.
- Conhecer o gridsystem do bootstrap e as principais classes para utilização em alert, buttons, modais, group e outros.
- Saber utilizar os conceitos básicos de lógica de programação em JavaScript (variáveis, condicionais, loops, escopos, referência, entre outros).
- Conhecer o DOM, capturar elementos, manipular elementos.
- Conhecer Ecma6, arrow function e class.
- Escrever arquivos HTML, CSS e JavaScript seguindo todas as boas práticas e convenções de cada linguagem.
- Fazer requisição AJAX.



- Entender os protocolos HTTP e REST.
- Saber utilizar uma ferramenta de automação, compilação, gerenciamento de dependências e empacotamento de projeto.
- Criar um projeto utilizando framework backend.
- Criar uma API REST (controllers, mappings, rotas, etc).
- Entender e saber aplicar os conceitos de inversão de controle e injeção de dependências.
- Tratar erros do backend e códigos de status HTTP.
- Fazer o mapeamento objeto-relacional.
- Utilizar linguagem de consultas objeto relacional.
- Habilitar a documentação com Swagger.
- Saber acessar outras APIs REST a partir do backend.
- Entender o protocolo HTTPS.
- Gerenciar autenticações.
- Fazer upload de arquivos.
- Criar recursos para envio de e-mails.



- Conhecer a estrutura e os conceitos de um framework frontend web.
- Utilizar gerenciador de pacotes para projetos frontend web e análise de estrutura.
- Compreender estado e imutabilidade.
- Gerenciar propriedades e estados.
- Compreender o conceito de componentização.
- Entender a renderização condicional.
- Gerenciar de rotas.
- Saber estilizar componentes.
- Realizar requisições à API backend (get, post, put e delete).



- Entender os conceitos do desenvolvimento mobile.
- Saber utilizar um framework frontend mobile.
- Saber configurar o ambiente de desenvolvimento mobile.
- Criar a estrutura do projeto.
- Utilizar uma ferramenta de gerenciamento, construção, deploy e emulação para aplicações mobile.
- Utilizar IDE e plugins para desenvolvimento de aplicações mobile.
- Criar componentes de classe.
- Gerenciar propriedades.
- Gerenciar estados.
- Criar componentes funcionais.
- Fazer layout e estilização.
- Definir navegação.
- Desenvolver aplicativo utilizando banco de dados embarcado.
- Realizar requisições à API backend (get, post, put e delete).
- Estilizar a interface do usuário com visual nativo do sistema operacional.



Trabalhar em equipe em projeto proposto por empresas parceiras dentro do conteúdo programático ministrado no curso, com o apoio do professor da disciplina e de mentores das empresas.

Ao longo das disciplinas os alunos trabalharão em grupos na construção de um aplicativo Web e Mobile.

Banco de Dados

- Cliente
- Pedido
- PedidoItem
- Produto
- Categoria
- NotaFiscal
- Usuário
- Dicionário de dados

Frontend Essencial

- *Site Institucional (sistema de pedidos)*
 - Home
 - Quem Somos
 - O que fazemos
 - Contato

Desenvolvimento de API Restful

- Persistência no DB
- API de integração com APP
- API de integração com SITE

Desenvolvimento Web

- *Frontend Sistema Controle NF*
- Login
- CRUD (Cliente, Pedido, Produto, Categoria, NotaFiscal, Usuário)

Desenvolvimento de Aplicativo Mobile

- *APP Controle NF*
- Splash Screen com animação
- Cadastro de dados Pessoais
- Autenticação via Login e Senha
- CRUD (Cliente, Pedido e NotaFiscal)

- Inglês instrumental (16 horas)
 - Mentoria de carreira (17 horas)
 - Monitoria + mentoria dos projetos (116 horas aproximadamente)
 - Maratona de programação (60 horas aproximadamente)
 - Cursos e workshops/palestras complementares (31 horas aprox.)
- 