

Plano de Curso

Técnico em Informática para Internet

Habilitação Profissional Técnica de Nível
Médio

Eixo Tecnológico: Informação e
Comunicação

Segmento: Tecnologia da Informação

Ano: 2020



Autorizado pelo Conselho Regional do Senac _____ em ____/____/____, pela Portaria _____.

1.**Identificação do Curso**

Título do Curso: Técnico em Informática para Internet

Eixo Tecnológico: Informação e Comunicação

Segmento: Tecnologia da Informação

Carga Horária: 1.000 horas

Código DN: 2691

Código CBO: CBO associada – 3171-05 – Programador de Internet

2.**Requisitos e Formas de Acesso¹****Requisitos de acesso:**

- Idade mínima: 16 anos.
- Escolaridade: Cursando o 2º ano do ensino médio.

Documentos exigidos para matrícula:

- Documento de identidade.
- CPF.
- Comprovante de escolaridade.
- Comprovante de residência.

Quando a oferta deste curso ocorrer por meio de parceria, convênio ou acordo de cooperação com outras instituições, deverão ser incluídas neste item as especificações, caso existirem. Menores de 18 anos devem estar acompanhados de um responsável legal, este portando documento de identidade e CPF.

3.**Justificativa e Objetivos**

A internet consolida-se como a responsável por uma revolução na maneira pela qual nos comunicamos com o mundo. Com esse movimento, uma área que certamente ganhou importância, não só pelo número de usuários envolvidos, mas também pelo montante

¹ Os requisitos de acesso indicados neste plano de curso consideram as especificidades técnicas da ocupação e legislações vigentes que versam sobre idade mínima, escolaridade e experiências requeridas para a formação profissional e exercício de atividade laboral. Cabe a cada Conselho Regional a aprovação de alterações realizadas neste item do plano de curso, desde que embasados em parecer da Diretoria de Educação Profissional.

financeiro movimentado, é a de comércio eletrônico (*e-commerce*). A 40ª edição do *Webshoppers*², o mais amplo relatório sobre *e-commerce* do país elaborado pela Ebit|Nielsen, mostrou que o comércio eletrônico cresceu no primeiro semestre de 2019 impulsionado pela expansão no volume de compra de produtos das categorias de Bens Não Duráveis. O exemplo desse salto é a expansão nos pedidos nos segmentos de “Alimentos e Bebidas” (82%) e “Petshop” (144%) – ambos na comparação com o mesmo período do ano passado.

O aumento do comércio eletrônico passa a ser também um fator de relevância para a necessidade de criar soluções informatizadas. Informação, entretenimento, cultura, lazer, educação, publicidade, negócios, entre outros, são serviços que proliferam na rede e, com o fortalecimento do mercado de dispositivos móveis, o *mobile marketing* se tornou o principal diferencial competitivo das marcas no país. Todo esse cenário traz uma demanda por *softwares* que podem ser acessados em multiplataformas, exigindo profissionais completos, atualmente desenvolvedores conhecidos como *full stack*.

Tudo isso cria um contexto favorável à inserção do Técnico em Informática para Internet no mercado de trabalho, dominando linguagens de programação tanto do lado servidor como do lado cliente, de banco de dados e de desenvolvimento relativos a interfaces, publicação e manutenção de aplicações *web* e *mobile*.

Objetivo geral:

Formar profissionais com competências para atuar e intervir em seu campo de trabalho, com foco em resultados.

Objetivos específicos:

- Promover o desenvolvimento do aluno por meio de ações que articulem e mobilizem conhecimentos, habilidades, valores e atitudes de forma potencialmente criativa e que estimule o aprimoramento contínuo.
- Estimular nos alunos, por meio de situações de aprendizagens, atitudes empreendedoras, sustentáveis e colaborativas.
- Articular as competências do perfil profissional com projetos integradores e outras atividades laborais que estimulem a visão crítica e a tomada de decisão para resolução de problemas.

² NIELSEN. **Brasileiro amplia experiência no e-commerce impulsionada por não-duráveis**. Insights. Disponível em: <https://www.nielsen.com/br/pt/insights/article/2019/brasileiro-amplia-experiencia-no-e-commerce-impulsionada-por-nao-duraveis/>. 22 out. 2019. Acesso em: 14 fev. 2020.

- Promover uma avaliação processual e formativa com base em indicadores das competências, que possibilitem a todos os envolvidos no processo educativo a verificação da aprendizagem.
- Incentivar a pesquisa como princípio pedagógico e para consolidação do domínio técnico-científico, utilizando recursos didáticos e bibliográficos.

4.

Perfil Profissional de Conclusão

O profissional técnico em informática para internet é responsável pelo desenvolvimento *web* e *mobile*. Planeja, desenvolve, testa e publica projetos de aplicações *web* para diversos dispositivos, inclusive móveis. Realiza, ainda, a manutenção, contribui nas decisões técnicas e fornece suporte para a equipe de desenvolvimento, proporcionando uma melhor experiência de acessibilidade, usabilidade e performance.

Esse profissional atua em empresas dos mais diferentes setores e segmentos, prestando serviços às organizações que façam uso de aplicações *web* e *mobile*. Também pode atuar de maneira autônoma e empreendedora, na construção de aplicações *web* e *mobile* a partir da identificação de uma necessidade e/ou demanda. Relaciona-se com equipes de desenvolvimento e outros profissionais de TI, bem como com clientes e usuários.

O profissional habilitado pelo Senac tem como Marcas Formativas: domínio técnico-científico, visão crítica, atitude empreendedora, sustentável e colaborativa, com foco em resultados. Essas marcas reforçam o compromisso da instituição com a formação integral do ser humano, considerando aspectos relacionados ao mundo do trabalho e ao exercício da cidadania. Tal perspectiva propicia o comprometimento do aluno com a qualidade do trabalho e com o desenvolvimento de uma visão ampla e consciente sobre sua atuação profissional e sobre sua capacidade de transformação da sociedade.

A ocupação está situada no eixo tecnológico Informação e Comunicação, cuja natureza é “comunicar”, e pertence ao segmento de Informática.

A seguir estão as competências que compõem o perfil do técnico em informática para internet:

- Elaborar projetos de aplicações para *web*.
- Desenvolver aplicações para *websites*.
- Codificar *front-end* de aplicações *web*.
- Publicar aplicações *web*.
- Desenvolver algoritmos.
- Codificar *back-end* de aplicações *web*.

- Implementar banco de dados para *web*.
- Desenvolver serviços *web*.
- Organizar o processo de trabalho no desenvolvimento de aplicações.
- Desenvolver interface gráfica para dispositivos móveis.
- Codificar aplicações para dispositivos móveis.
- Codificar acesso a *web services* e recursos de sistemas móveis.
- Publicar aplicações para dispositivos móveis.

O curso de habilitação profissional técnica de nível médio em Informática para Internet do Senac possibilita ao aluno as seguintes certificações intermediárias:

Qualificação Profissional Técnica de Nível Médio Desenvolvedor *Front-End*

O profissional **desenvolvedor *front-end*** é responsável pela codificação de toda a interface visual de um *site* dinâmico. Suas atribuições envolvem desde o projeto, construção e manutenção do *layout* do produto digital para internet até a disposição de seus elementos multimídias e a interação de dados, proporcionando melhor interatividade entre os elementos da aplicação, por meio de tecnologias e dos princípios de acessibilidade, usabilidade e responsividade.

Esse profissional atua em empresas dos mais diferentes setores e segmentos e, ainda, de forma empreendedora, como autônomo, prestando serviços às organizações que utilizam sua comunicação por meio de *si*, promovendo a experiência do usuário a partir da interface gráfica.

Relaciona-se com equipes de desenvolvimento e outros profissionais de TI, bem como com clientes e usuários da aplicação.

O profissional qualificado pelo Senac tem como marcas formativas: domínio técnico-científico, visão crítica, atitude empreendedora, sustentável e colaborativa, com foco em resultados. Essas marcas reforçam o compromisso da instituição com a formação integral do ser humano, considerando aspectos relacionados ao mundo do trabalho e ao exercício da cidadania. Essa perspectiva propicia o comprometimento do aluno com a qualidade do trabalho e o desenvolvimento de uma visão ampla e consciente sobre sua atuação profissional e sobre sua capacidade de transformação da sociedade.

A ocupação está situada no eixo tecnológico Informação e Comunicação, cuja natureza é “comunicar”, e pertence ao segmento de Informática.

A seguir estão as competências que compõem o perfil do desenvolvedor *front-end*:

- Elaborar projetos de aplicações para *web*.
- Desenvolver aplicações para *websites*.
- Codificar *front-end* de aplicações *web*.
- Publicar aplicações *web*.

Qualificação Profissional Técnica de Nível Médio Desenvolvedor *Back-End*

O profissional **desenvolvedor *back-end*** é responsável pelo desenvolvimento de *sites* utilizando linguagens de programação. Dentre suas principais atribuições estão a elaboração de projeto, a estabilidade e a integração com banco de dados, *webservices*, aplicações *front-end* e com outros sistemas, além do desenvolvimento e manutenção das regras de negócios de aplicações. Realiza, ainda, a documentação e testes durante todo o processo e o suporte técnico aos clientes internos e externos.

Esse profissional atua em empresas dos mais diferentes setores e segmentos e, ainda, de forma empreendedora, como autônomo, prestando serviços as organizações que façam uso de aplicações *web*.

Relaciona-se com equipes de desenvolvimento e outros profissionais de TI, bem como com clientes e usuários de aplicativos.

O profissional qualificado pelo Senac tem como marcas formativas: domínio técnico-científico, visão crítica, atitude empreendedora, sustentável e colaborativa, com foco em resultados. Essas marcas reforçam o compromisso da instituição com a formação integral do ser humano, considerando aspectos relacionados ao mundo do trabalho e ao exercício da cidadania. Essa perspectiva propicia o comprometimento do aluno com a qualidade do trabalho e o desenvolvimento de uma visão ampla e consciente sobre sua atuação profissional e sobre sua capacidade de transformação da sociedade.

A ocupação está situada no eixo tecnológico Informação e Comunicação, cuja natureza é “comunicar”, e pertence ao segmento de Informática.

A seguir estão as competências que compõem o perfil do desenvolvedor *back-end*:

- Elaborar projetos de aplicações para *web*.
- Desenvolver algoritmos.
- Codificar *back-end* de aplicações *web*.
- Implementar banco de dados para *web*.
- Desenvolver serviços *web*.
- Publicar aplicações *web*.
- Organizar o processo de trabalho no desenvolvimento de aplicações.

Qualificação Profissional Técnica de Nível Médio Desenvolvedor *Mobile*

O profissional **desenvolvedor *mobile* ou *web mobile*** é responsável por codificar aplicativos *mobile*. Realiza, ainda, um ciclo de desenvolvimento de uma solução para dispositivos móveis, para plataformas como IOS ou Android, por meio de *frameworks* ou padrões *web*, garantindo a acessibilidade e usabilidade e possibilitando a integração dos dados.

Esse profissional atua em empresas dos mais diferentes setores e segmentos e, ainda, de forma empreendedora, como autônomo, prestando serviços às organizações que façam uso de aplicações para dispositivos móveis. Relaciona-se com equipes de desenvolvimento e outros profissionais de TI, bem como com clientes e usuários da aplicação.

O profissional qualificado pelo Senac tem como marcas formativas: domínio técnico-científico, visão crítica, atitude empreendedora, sustentável e colaborativa, com foco em resultados. Essas marcas reforçam o compromisso da instituição com a formação integral do ser humano, considerando aspectos relacionados ao mundo do trabalho e ao exercício da cidadania. Essa perspectiva propicia o comprometimento do aluno com a qualidade do trabalho e o desenvolvimento de uma visão ampla e consciente sobre sua atuação profissional e sobre sua capacidade de transformação da sociedade.

A ocupação está situada no eixo tecnológico Informação e Comunicação, cuja natureza é “comunicar”, e pertence ao segmento de Informática.

A seguir estão as competências que compõem o perfil do desenvolvedor *mobile*:

- Elaborar projetos de aplicações para *web*.
- Organizar o processo de trabalho no desenvolvimento de aplicações.
- Desenvolver interface gráfica para dispositivos móveis.
- Codificar aplicações para dispositivos móveis.
- Codificar acesso a *web services* e recursos de sistemas móveis.
- Publicar aplicações para dispositivos móveis.

O Modelo Pedagógico Senac estrutura o currículo do curso Técnico em Informática para Internet com base nos fazeres profissionais – as competências –, organizadas a partir das seguintes Unidades Curriculares:

Unidades Curriculares		Carga horária
UC 5: Projeto Integrador Desenvolvedor <i>Front-End</i> (24 horas)	UC 1: Elaborar projetos de aplicações para <i>web</i> *	36 horas
	UC 2: Desenvolver aplicações para <i>websites</i>	72 horas
	UC 3: Codificar <i>front-end</i> de aplicações <i>web</i>	96 horas
	UC 4: Publicar aplicações <i>web</i> **	36 horas
Qualificação Profissional Técnica em Desenvolvedor <i>Front-End</i> (UC 1 a UC 5)		264 horas
UC 11: Projeto Integrador Desenvolvedor <i>Back-End</i> (32 horas)	UC 6: Desenvolver algoritmos	96 horas
	UC 7: Codificar <i>back-end</i> de aplicações <i>web</i>	120 horas
	UC 8: Implementar banco de dados para <i>web</i>	84 horas
	UC 9: Desenvolver serviços <i>web</i>	48 horas
	UC 10: Organizar o processo de trabalho no desenvolvimento de aplicações***	48 horas
Qualificação Profissional Técnica em Desenvolvedor <i>Back-End</i> (UC 1* e UC 4** com UC 6, UC 7, UC 8, UC 9, UC 10 e UC 11)		500 horas
UC 16: Projeto Integrador Desenvolvedor <i>Mobile</i> (32 horas)	UC 12: Desenvolver interface gráfica para dispositivos móveis	60 horas
	UC 13: Codificar aplicações para dispositivos móveis	120 horas
	UC 14: Codificar acesso a <i>web services</i> e recursos de sistemas móveis	60 horas

² De acordo com a Resolução CNE/CEB nº 6, de 20 de setembro de 2012, os planos de cursos técnicos de nível médio presenciais podem prever até 20% (vinte por cento) do total da carga horária do curso em atividades não presenciais, desde que haja suporte tecnológico e que seja garantido o atendimento por docentes e tutores qualificados.

Unidades Curriculares		Carga horária
	UC 15: Publicar aplicações para dispositivos móveis	36 horas
Qualificação Profissional Técnica em Desenvolvedor <i>Mobile</i> (UC 1* e UC 10***, com UC 12, UC 13, UC 14, UC 15 e UC 16)		392 horas
Técnico em Informática para Internet (UC 1 a UC 16)		1000 horas

Equivalência entre Unidades Curriculares

As Unidades Curriculares 1, 2 e 4 desse curso técnico são equivalentes às competências do perfil profissional da Qualificação Profissional em Web Designer.

As Unidades Curriculares 1 e 4 desse curso técnico são equivalentes a algumas competências do perfil profissional da Qualificação Profissional em Programador Web.

5.1. Detalhamento das Unidades Curriculares:

Unidade Curricular 1: Elaborar projetos de aplicações para *web*

Carga horária: 36 horas

Indicadores
1. Define os objetivos do projeto, de acordo com as necessidades do cliente e público-alvo.
2. Elabora proposta de trabalho de acordo com arquitetura da informação da aplicação <i>web</i> e das estratégias tecnológicas.
3. Elabora protótipo para <i>websites</i> de acordo com o <i>briefing</i> .
4. Redige o projeto da aplicação <i>web</i> de acordo com a proposta e protótipo.

Elementos da Competência
Conhecimentos <ul style="list-style-type: none"> • <i>Briefing</i>: conceito, especificidades para projetos <i>web</i> e modelos. • Técnicas de criatividade: <i>brainstorming</i>, mapas mentais e painéis semânticos. • Análise de mercado: identificação do perfil do cliente e público-alvo, concorrentes diretos e indiretos. • Domínios de internet: conceito, registro e disponibilidade de serviços. • Projetos <i>web</i>: tendências, tecnologias, gestão de projetos (custos, calendários de

Elementos da Competência
<p>tarefas e relatórios de acompanhamento).</p> <ul style="list-style-type: none"> Arquitetura da informação: conceito e aplicações, mapa do <i>site</i> e estruturas de navegação, organização de conteúdo. Proposta comercial: características, requisitos, elaboração e modelos de contrato de serviço. Requisitos: conceitos e técnicas para análise e gerenciamento. <p>Habilidades</p> <ul style="list-style-type: none"> Interpretar <i>briefing</i> para projetos de <i>websites</i>. Identificar requisitos técnicos para projetos de <i>websites</i>. Pesquisar domínio disponível e serviços de hospedagem para <i>websites</i>. Organizar arquivos e atividades por etapas do projeto. Estruturar arquitetura dos elementos de conteúdo de <i>websites</i>. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> Responsabilidade e comprometimento com os acordos estabelecidos. Proatividade no desenvolvimento das atividades profissionais. Colaboração no desenvolvimento do trabalho em equipe. Respeito aos direitos de propriedade intelectual. Comprometimento com padrões de usabilidade e acessibilidade <i>web</i>.

Unidade Curricular 2: Desenvolver aplicações para *websites*

Carga horária: 72 horas

Indicadores
<ol style="list-style-type: none"> Cria imagens, <i>layouts</i> e animações otimizadas para <i>website</i> de acordo com os princípios de comunicação visual, normas e tendências de mercado. Utiliza linguagem de marcação de conteúdo e estilo, de acordo com as normas e padrões tecnológicos. Testa padrões de acessibilidade e usabilidade do <i>website</i> de acordo com as normas. Analisa inconsistências no funcionamento do <i>website</i>, de acordo com os padrões W3C.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> Imagem digital: conceitos de vetor e <i>bitmap</i>, formatos e aplicações, unidades de medida, densidade de <i>pixels</i>, taxa de <i>bits</i>, animações <i>web</i>.

Elementos da Competência

- Comunicação visual: tipografia, teoria e modos de cor, *grid* e alinhamento, *Gestalt* (conceito e aplicabilidade).
- *Wireframes* e protótipos: conceito, estrutura e desenvolvimento.
- *Design* responsivo: conceitos e aplicações.
- Projeto de *website*: características funcionais, usabilidade, acessibilidade e ergonomia.
- Estrutura semântica: conceitos, linguagem de marcação de conteúdo.
- Estilização de páginas: definições de estilos e integração com estrutura de conteúdo.
- *Web standards*: boas práticas e padrões recomendados pelo W3C.
- Desempenho e compatibilidade: conceito e ferramentas.
- Código do *website*: testes, validação e correção.
- *Content Management System* (CMS) – interface, requisitos básicos, servidor local e banco de dados, configuração de usuários administrativos, customização, CSS e temas, *plug-ins*, implantação e publicação.
- Lei Geral de Proteção de Dados Pessoais: fundamentos, aplicabilidade, princípios legais, direitos, transferência internacional de dados, agentes de tratamento, encarregado pelo tratamento de dados pessoais (DPO), segurança, boas práticas, fiscalização e penalidades e a definição e papel da ANPD – Agência Nacional de Proteção de Dados.

Habilidades

- Aplicar o sistema de cor pertinente ao *layout* da aplicação.
- Organizar conteúdo visual e textual para web.
- Integrar linguagens de estilo e marcação de conteúdo.
- Utilizar ferramentas para análise de desempenho.
- Testar compatibilidade nos diversos navegadores.

Atitudes/Valores

- Cordialidade no trato com as pessoas.
- Sigilo no tratamento de dados e informações.
- Iniciativa na proposição de soluções de projetos.
- Proatividade no desenvolvimento das atividades profissionais.
- Colaboração no desenvolvimento do trabalho em equipe.
- Transparência no tratamento e proteção de dados pessoais.

Elementos da Competência
<ul style="list-style-type: none"> • Zelo pela organização do ambiente de trabalho. • Respeito aos direitos de propriedade intelectual. • Comprometimento com padrões de usabilidade e acessibilidade na <i>web</i>.

Unidade Curricular 3: Codificar *front-end* de aplicações *web*

Carga horária: 96 horas

Indicadores
<ol style="list-style-type: none"> 1. Cria blocos de <i>back-end</i>, utilizando linguagens de programação de <i>script</i>, de acordo com os requisitos do projeto de <i>software</i>. 2. Manipula os elementos estruturais, de acordo com os requisitos do projeto. 3. Programa comportamentos dinâmicos, definindo estilos e animações, de acordo com requisitos do projeto e padrões de acessibilidade e usabilidade. 4. Implementa <i>frameworks</i> de acordo com as necessidades do projeto. 5. Comunica requisições conforme recursos disponibilizados pelo <i>back-end</i>. 6. Implementa correções e melhorias de acordo com a depuração de código de <i>script</i>. 7. Implementa usabilidade e acessibilidade a páginas <i>web</i> de acordo com os padrões do W3C.

Elementos da Competência
Conhecimentos <ul style="list-style-type: none"> • <i>Sites</i> estáticos e dinâmicos: conceitos, diferenças, linguagens de <i>script</i>, tecnologias e aplicações. • Fundamentos de Lógica de Programação: conceitos de algoritmos, de entradas e saídas, manipulação e processamento de dados. • Programação: conceito, análise de requisitos do projeto e a relação com a codificação em <i>front-end</i>. • Linguagem de <i>scripts</i>: sintaxe – operadores, palavras reservadas, identificadores, delimitadores e comentários; variáveis e tipos de dados, estruturas de controle condicional e laços de repetição. • Eventos e funções: parâmetros, retornos e tipos de eventos. • <i>Document Object Model</i> (DOM): objetos, propriedades e eventos; manipulação de elementos, atribuição de eventos e estilos dinâmicos. • <i>Framework</i>: conceitos, aplicabilidade e tendências.

Elementos da Competência
<ul style="list-style-type: none"> • Usabilidade e acessibilidade: princípios aplicados ao comportamento dinâmico da página. • Requisições assíncronas: conceitos e aplicabilidade. • Depuração de linguagem de <i>scripts</i>: <i>de bugger</i> e testes de código. <p>Habilidades</p> <ul style="list-style-type: none"> • Interpretar requisitos de projetos. • Identificar estruturas e funcionalidades da linguagem de <i>script</i>. • Utilizar termos técnicos nas rotinas de trabalho. • Utilizar padrões de boas práticas no desenvolvimento do <i>site</i>. • Aplicar responsividade no comportamento do <i>website</i>. • Interpretar manuais e documentações técnicas. • Indentar códigos de <i>script</i>. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> • Cordialidade no trato com as pessoas. • Sigilo no tratamento de dados e informações. • Comprometimento com os padrões e as normas de desenvolvimento <i>web</i>. • Iniciativa na proposição de soluções de projetos. • Proatividade no desenvolvimento das atividades profissionais. • Colaboração no desenvolvimento do trabalho em equipe. • Zelo pela organização do ambiente de trabalho. • Respeito aos direitos de propriedade intelectual.

Unidade Curricular 4: Publicar aplicações *web*

Carga horária: 36 horas

Indicadores
<ol style="list-style-type: none"> 1. Define serviço de hospedagem de acordo com os objetivos do projeto. 2. Formata e exporta arquivos do projeto local para servidor <i>web</i>, de acordo com as tecnologias disponíveis. 3. Verifica compatibilidade e performance do <i>website</i>, de acordo com as normas e exigências do mercado.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> • Tecnologias de servidor <i>web</i>: conceitos e princípios de funcionamento. • Serviços de hospedagem: gratuitos, pagos, compartilhados e dedicados. • Gerenciamento do <i>site</i>: atualização de informações e <i>backups</i>. • Transferência de arquivos: hospedagem via FTP e <i>upload</i> no servidor. • Testes de desempenho: comportamento e integridade do <i>website</i>. <p>Habilidades</p> <ul style="list-style-type: none"> • Hospedar <i>websites</i>. • Identificar e corrigir erros no <i>website</i>. • Realizar <i>backups</i> de <i>websites</i>. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> • Cordialidade no trato com as pessoas. • Sigilo no tratamento de dados e informações. • Iniciativa na proposição de soluções de projetos. • Proatividade no desenvolvimento das atividades profissionais. • Colaboração no desenvolvimento do trabalho em equipe. • Zelo pela organização do ambiente de trabalho. • Respeito aos direitos de propriedade intelectual.

Unidade Curricular 6: Desenvolver algoritmos

Carga horária: 96 horas

Indicadores
<ol style="list-style-type: none"> 1. Estrutura uma sequência lógica de ações, de acordo com as etapas de resolução de um problema. 2. Estrutura algoritmos, com base na lógica computacional e nos requisitos funcionais descritos no projeto. 3. Seleciona e manipula estruturas de dados conforme as funcionalidades descritas no projeto de <i>software</i>. 4. Cria sub-rotinas genéricas e reutilizáveis, conforme as funcionalidades descritas no projeto de <i>software</i>. 5. Valida algoritmo com base em valores de entradas e saídas de acordo com funcionalidades descritas do projeto de <i>software</i>. 6. Seleciona a metodologia ágil, de acordo com os requisitos do projeto de <i>software</i>.

Elementos da Competência

Conhecimentos

- Lógica de programação: conceito de algoritmos, pensamento computacional.
- Algoritmos: por descrição narrativa, por fluxograma e por pseudocódigo.
- Manipulação de dados: variáveis, constantes, tipos de dados, operadores e expressões.
- Estrutura lógica: lógica booleana; condicional simples, composta e repetição.
- Sub-rotinas: declaração – funções e procedimentos; parâmetros, retorno e chamada.
- Estruturas de dados: vetores e matrizes, declaração, manipulação e aplicabilidade.
- Validação de algoritmos: testes de mesa e depuração.
- Regras de negócio: conceitos, características, requisitos funcionais e não funcionais.
- Metodologias ágeis: conceitos e tipos aplicados ao desenvolvimento de *software*.
- Orientação a objetos: definição, fundamentos e requisitos do sistema.
- UML: conceitos, tipos (diagrama de caso de uso e diagrama de classe) e aplicabilidade.

Habilidades

- Identificar requisitos de *software*.
- Interpretar requisitos de projetos.
- Resolver problemas lógicos e aritméticos.
- Elaborar documentos técnicos.
- Interpretar códigos de programação.
- Aplicar metodologias ágeis.
- Testar algoritmo.

Atitudes/Valores

- Cordialidade no trato com as pessoas.
- Sigilo no tratamento de dados e informações.
- Assertividade na tomada de decisões.
- Proatividade no desenvolvimento das atividades profissionais.
- Colaboração no desenvolvimento do trabalho em equipe.
- Zelo pela organização do ambiente de trabalho.
- Respeito aos direitos de propriedade intelectual.

Elementos da Competência
<ul style="list-style-type: none"> Comportamento investigativo na automação de recursos.

Unidade Curricular 7: Codificar *back-end* de aplicações *web*

Carga horária: 120 horas

Indicadores
<ol style="list-style-type: none"> 1. Configura o ambiente de desenvolvimento conforme as funcionalidades e características do projeto. 2. Cria estruturas de código utilizando linguagem de programação para <i>back-end</i>, de acordo com os requisitos do projeto de <i>software</i>. 3. Documenta as funcionalidades do projeto <i>web</i> conforme requisitos do projeto. 4. Desenvolve tratamento de requisições POST e GET do <i>front-end</i>, de acordo com a linguagem de programação <i>back-end</i>. 5. Manipula elementos de <i>front-end</i>, de acordo com as funcionalidades descritas no projeto de <i>software</i>. 6. Implementa <i>session</i> e <i>cookies</i> da aplicação <i>web</i>, de acordo com a linguagem de programação <i>back-end</i>. 7. Integra soluções de segurança de dados ao projeto, de acordo com normas e boas práticas de segurança da informação. 8. Desenvolve recursos de chamadas assíncronas, de acordo com requisitos do projeto. 9. Aplica correções e melhorias a partir da validação e depuração do código de <i>back-end</i>, conforme funcionalidade do projeto.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> Arquitetura de aplicações <i>web</i>: conceito de <i>back-end</i> e sua interação com <i>front-end</i>. Ambiente de desenvolvimento: conceito, interface, configuração e compilação. Linguagem de programação para <i>back-end</i>: sintaxe, palavras reservadas, variáveis e estruturas de dados; desvios condicionais e laços de repetição; declaração e uso de sub-rotinas, passagem de parâmetro e retorno. Biblioteca da linguagem: funções próprias da linguagem e suas aplicações. Memória da aplicação <i>web</i>: armazenamento de dados entre requisições, <i>cookies</i> e <i>sessions</i>; aplicabilidade. Orientação a objetos na linguagem de programação <i>back-end</i>: classes, objetos,

Elementos da Competência

interfaces, atributos, tipos de dados, modificadores de acesso, métodos e propriedades, encapsulamento e agregação; conceitos de herança, polimorfismo, tratamento de erros e exceções.

- Política de recuperação de dados em programação para aplicações *web* – conceito, análise periódica, procedimentos de *backup* e *restore*.
- Segurança da informação: conceito, mecanismos, ameaças e políticas de segurança.
- Integração com *front-end*: manipulação de elementos da página, tratamento de requisição GET e POST, processamento de dados de formulários e resposta ao *front-end*.
- Chamadas assíncronas: integração com requisições assíncronas de *front-end*, padrões XML e Json, resposta de requisição de cliente via servidor, aplicação em página *web*.
- Testes: conceitos de qualidade de *software*, técnicas de teste, plano de teste, caixa branca e caixa preta.
- Versionamento de código: segurança da informação, instalação e configuração.
- Documentação de programas de computador: diagramas (classes, casos de uso, sequência) e fluxo.

Habilidades

- Interpretar requisitos de projetos.
- Executar comandos de interatividade em interfaces *web*.
- Registrar e organizar as informações e códigos da aplicação *web*.
- Testar serviços *web*.
- Utilizar termos técnicos nas rotinas de trabalho.
- Aplicar versionamento ao código do projeto.

Atitudes/Valores

- Cordialidade no trato com as pessoas.
- Sigilo no tratamento de dados e informações.
- Iniciativa na proposição de soluções de projetos.
- Proatividade no desenvolvimento das atividades profissionais.
- Colaboração no desenvolvimento do trabalho em equipe.
- Respeito aos direitos de propriedade intelectual.
- Atualização contínua no desenvolvimento das atividades profissionais.

Unidade Curricular 8: Implementar banco de dados para *web*

Carga horária: 84 horas

Indicadores
<ol style="list-style-type: none">1. Modela a estrutura do banco de dados de acordo com os requisitos de uma aplicação <i>web</i>.2. Prepara o ambiente de desenvolvimento, conforme requisitos do Sistema de Gerenciamento de Banco de Dados (SGBD).3. Estrutura banco de dados físicos utilizando linguagem de manipulação de dados de acordo com a necessidade do projeto.4. Integra banco de dados à aplicação <i>web</i> de acordo com os requisitos do projeto.5. Manipula dados a partir da aplicação, utilizando linguagem de programação <i>back-end</i> e integração ao banco de dados, de acordo com a necessidade do projeto.

Elementos da Competência
Conhecimentos <ul style="list-style-type: none">• Banco de dados: conceito, tipos e volume de dados; principais aspectos da evolução do banco de dados.• Modelo conceitual e lógico: conceitos, entidade, relacionamentos, atributos, índices, cardinalidade, chave candidata, chave primária e estrangeira, integridade referencial, tabelas, campos e normalização.• Sistema Gerenciador de Banco de Dados (SGBD): conceito, arquitetura, implantação, instalação, funcionalidades e manutenção.• <i>Structured Query Language</i> (SQL) – Linguagem de Consulta Estruturada: conceitos, sintaxe, ordenação de resultados, filtragem de linhas, utilização de cálculos em SQL, manipulação de datas, união, intersecção e junção de dados, funções de agrupamentos e subconsultas, criação de entidade, visão, atributos, índices, chave candidata, chave primária e estrangeira, relacionamentos e integridade referencial, importação e exportação de dados e <i>SQL procedural</i>.• <i>Data Definition Language</i> (DDL) – Linguagem de Definição de Dados: criação de tabelas, colunas, chaves primária, chaves estrangeiras e índices.• <i>Data Manipulation Language</i> (DML) – Linguagem de Manipulação de Dados: inclusão, atualização e exclusão de dados.• <i>Data Query Language</i> (DQL) – Linguagem de Consulta de Dados: seleção, filtragem, ordenação, agrupamento, funções de consulta e junção de dados (<i>join</i>).

Elementos da Competência

- Manipulação de dados com linguagem de programação: bibliotecas de conexão, comandos de conexão e desconexão, comandos de manipulação de dados – *Create-Read-Update-Delete* (CRUD), manipulação de respostas de consultas e desenvolvimento de CRUD.
- Programação via banco de dados: *procedures, functions, triggers* e *views*.
- Manutenção de banco de dados: *backup* e *tunning*.
- *Frameworks* de manipulação de banco de dados: conceitos e aplicabilidade.
- Segurança do banco de dados para aplicações *web*: normas de segurança da informação, redundância, concorrência, integridade e consistência, criptografia, autenticação, *SQL Injection*, ameaças, controle de acesso, cópia de segurança e *restore*.
- Orientação a objetos na linguagem de programação *back-end*: herança, polimorfismo, tratamento de exceções; aplicação com o banco de dados.
- Testes: registros, *bugtrackers*, depuração de código.
- Versionamento de código: criação de repositório, comandos de envio e recuperação de código.
- Bancos de dados NoSQL: conceitos e aplicações.
- Lei Geral de Proteção de Dados Pessoais: fundamentos, aplicabilidade, princípios legais, direitos, transferência internacional de dados, agentes de tratamento, encarregado pelo tratamento de dados pessoais (DPO), segurança, boas práticas, fiscalização e penalidades e a definição e papel da Agência Nacional de Proteção de Dados (ANPD).

Habilidades

- Interpretar requisitos de projetos.
- Documentar etapas de desenvolvimento do *software*.
- Criar diagramas de banco de dados.
- Aplicar os comandos da linguagem SQL.
- Aplicar políticas de segurança no banco de dados.
- Executar comandos de interatividade entre interfaces *web* e banco de dados.
- Aplicar versionamento ao código do projeto.
- Utilizar termos técnicos nas rotinas de trabalho.
- Transparência no tratamento e proteção de dados pessoais.

Elementos da Competência
Atitudes/Valores <ul style="list-style-type: none"> • Sigilo no tratamento de dados e informações. • Iniciativa na proposição de soluções de projetos. • Proatividade no desenvolvimento das atividades profissionais. • Colaboração no desenvolvimento do trabalho em equipe. • Respeito aos direitos de propriedade intelectual.

Unidade Curricular 9: Desenvolver serviços web

Carga horária: 48 horas

Indicadores
<ol style="list-style-type: none"> 1. Cria estruturas de código de serviço <i>web</i> utilizando linguagem de <i>back-end</i> de acordo com os requisitos do projeto. 2. Realiza integração do código <i>back-end</i> com serviços de terceiros conforme os requisitos do projeto e as regras de consumo de dados via <i>web</i>. 3. Consome serviços <i>web</i> utilizando linguagem de <i>back-end</i> e <i>front-end</i> de acordo com os requisitos do projeto e as funcionalidades das linguagens. 4. Manipula os registros de <i>web services</i> conforme os requisitos do projeto.

Elementos da Competência
Conhecimentos <ul style="list-style-type: none"> • <i>Web services</i>: histórico e conceito. • Programação orientada a serviços: objetivos, benefícios, contratos, acoplamento, abstração, capacidade de reuso, autonomia, independência de estado, visibilidade e composição de serviços. • Protocolo HTTP: conceitos, requisições e respostas, operações (GET, POST, PUT, DELETE), informações de cabeçalhos, códigos de <i>status</i> da resposta. • Padrão <i>Representational State Transfer</i> – REST: conceitos, tipos de requisições, interpretação das operações, parâmetros, processamento e resposta; ferramentas de linguagem de programação <i>back-end</i> para programação de serviços REST. • Representações de dados: texto plano, <i>Java Script Object Notation</i> (JSON); <i>Extensible Markup Language</i> (XML). • Programação de serviços: implementação e instalação. • Consumo de serviços <i>web</i>: tipos, ferramentas de <i>back-end</i> e <i>front-end</i>, chamadas

Elementos da Competência
<p>síncronas e assíncronas, envio de parâmetros e tratamento de respostas.</p> <ul style="list-style-type: none"> • Testes unitários: conceitos e ferramentas. • Versionamento de código: recuperação de histórico e resolução de conflitos. • Segurança em serviços <i>web</i>: HTTPS, autenticação e melhores práticas. <p>Habilidades</p> <ul style="list-style-type: none"> • Interpretar requisitos de projetos. • Estruturar dados com XML e JSON. • Expandir aplicação com serviços de terceiros. • Publicar serviços <i>web</i> reutilizáveis. • Testar serviços <i>web</i>. • Aplicar versionamento ao código do projeto. • Utilizar termos técnicos nas rotinas de trabalho. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> • Sigilo no tratamento de dados e informações. • Proatividade no desenvolvimento das atividades profissionais. • Colaboração no desenvolvimento do trabalho em equipe. • Zelo pela organização do ambiente de trabalho. • Respeito aos direitos de propriedade intelectual.

Unidade Curricular 10: Organizar o processo de trabalho no desenvolvimento de aplicações

Carga horária: 48 horas

Indicadores
<ol style="list-style-type: none"> 1. Executa procedimentos de controle e registro de ocorrência em <i>softwares</i> de gestão, de acordo com o fluxo de trabalho. 2. Identifica a necessidade de adaptação das etapas e cronograma de trabalho, de acordo com os procedimentos da organização. 3. Elabora relatórios dos processos de trabalho conforme métricas de uma organização.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> • Etapas de trabalho: tipos de equipe, cultura organizacional, cronograma, acompanhamento do desenvolvimento e custos do trabalho.

Elementos da Competência
<ul style="list-style-type: none"> • Empreendedorismo: características, intraempreendedorismo e mercado de trabalho. • Serviços de terceiros: levantamento de fornecedores; prioridades, custos e características. • Suporte técnico: comunicação, controle e registro de ocorrências e tipos de ações corretivas; documentações do <i>software</i>, conforme necessidades do cliente e normas da organização. • Negociação de conflitos: mediação, processo de comunicação, ruídos e <i>feedbacks</i> (formas de fornecer e receber <i>feedbacks</i> na resolução de conflitos). • Carreira profissional: itinerário, limites de atuação e planejamento; etiqueta pessoal e profissional. • Relatórios de desenvolvimento: tipos, coerência e coesão textual; estrutura de documentos. <p>Habilidades</p> <ul style="list-style-type: none"> • Comunicar-se de forma assertiva. • Operar editor de texto, planilhas eletrônica e apresentação de <i>slides</i>. • Mediar conflitos nas situações de trabalho. • Pesquisar dados e informações do desenvolvimento do trabalho. • Analisar as etapas de trabalho. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> • Cordialidade e empatia no trato com as pessoas. • Colaboração no desenvolvimento do trabalho em equipe. • Responsabilidade e comprometimento com os acordos estabelecidos. • Sigilo no tratamento de dados e informações. • Zelo na apresentação pessoal e postura profissional. • Proatividade no atendimento e na resolução de problemas. • Respeito à diversidade humana.

Unidade Curricular 12: Desenvolver interface gráfica para dispositivos móveis

Carga horária: 60 horas

Indicadores
1. Estrutura leiaute para aplicativos móveis conforme os princípios de comunicação visual, normas e tendências de mercado.

Indicadores
<ol style="list-style-type: none"> 2. Implementa interface gráfica conforme leiaute e arquitetura <i>mobile</i>. 3. Testa a usabilidade e acessibilidade de aplicativos, conforme requisitos do projeto e padrões da arquitetura móvel. 4. Programa interatividade em dispositivos móveis, conforme leiaute e sistemas <i>mobile</i>. 5. Define conteúdo visual e textual para dispositivos móveis de acordo com os requisitos e elementos disponibilizados pelo projeto.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> • Sistemas <i>mobile</i>: conceitos e características. • Desenvolvimento híbrido de aplicativos: conceito, linguagem de marcação de conteúdo e estilo; <i>webstandards</i> – práticas e padrões recomendados pelo W3C; modelo <i>Progressive Web Apps</i> (PWA). • Ambiente de desenvolvimento para dispositivos móveis: instalação e configuração; emuladores de dispositivos móveis (versões de SDK e parâmetros de <i>hardware</i>). • Programação para <i>mobile</i>: linguagens de programação e sintaxe. • Arquitetura em camadas: conceitos <i>Model View Controller</i> (MVC). • Leiaute para aplicativo móvel: responsividade, usabilidade e acessibilidade, métodos de entradas de dados, uso de cores, posicionamento de elementos. • Telas para aplicativos: tecnologias de programação de interface do sistema <i>mobile</i>, organização e configuração dos elementos visuais, componentes de interface do sistema <i>mobile</i>. • Padrões de <i>design</i> de interação para dispositivos móveis. • Interação com interface gráfica: programação de resposta ao comando do usuário, uso de toque e teclado virtual, alternância de telas, <i>feedback</i> ao usuário, limitações e possibilidades de navegação. • Testes em <i>mobile</i>: verificação de usabilidade e acessibilidade. • Versionamento de código: ramificações e <i>tags</i>. <p>Habilidades</p> <ul style="list-style-type: none"> • Interpretar requisitos de projetos. • Organizar elementos gráficos para dispositivos móveis. • Otimizar imagens para dispositivos móveis.

Elementos da Competência
<ul style="list-style-type: none"> • Aplicar linguagem de marcação de conteúdo e estilo e aplicativos híbridos. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> • Sigilo no tratamento de dados e informações. • Iniciativa na proposição de soluções de projetos. • Proatividade no desenvolvimento das atividades profissionais. • Colaboração no desenvolvimento do trabalho em equipe. • Respeito aos direitos de propriedade intelectual. • Comprometimento com padrões de usabilidade e acessibilidade na <i>web</i>. • Atualização contínua no desenvolvimento das atividades profissionais. • Cordialidade no trato com as pessoas. • Atitude propositiva no desenvolvimento do trabalho.

Unidade Curricular 13: Codificar aplicações para dispositivos móveis

Carga horária: 120 horas

Indicadores
<ol style="list-style-type: none"> 1. Configura ambientes de programação (IDE) para dispositivos móveis de acordo com os requisitos do projeto. 2. Cria estruturas de código conforme requisitos do projeto e a linguagem de programação para dispositivos móveis. 3. Desenvolve interatividade entre aplicação e interface conforme elementos da tecnologia <i>mobile</i> e as necessidades do projeto. 4. Aplica correções e melhorias a partir da validação e depuração do código do aplicativo, conforme necessidades do projeto. 5. Aplica recursos da biblioteca do sistema <i>mobile</i> de acordo com necessidades do aplicativo. 6. Programa persistência local de dados utilizando arquivos e banco de dados portáteis de acordo com as necessidades do sistema. 7. Gera arquivos executáveis, compilando o código desenvolvido de acordo com as especificações da arquitetura móvel.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> • Linguagem de programação orientada a objetos para dispositivos móveis: visão

Elementos da Competência

geral da linguagem de programação, palavras reservadas, classes usadas e sintaxe.

- Ambiente de desenvolvimento para dispositivos móveis: plataforma, *Integrated Development Environment* (IDE), estrutura de projeto e arquivos específicos.
- *Frameworks*: bibliotecas de terceiros para desenvolvimento, recursos e aplicabilidade; conceitos de desenvolvimento *cross-platform*.
- Versionamento de código: união de ramificações (*merge*) e tratamento de conflitos.
- Testes e depuração de aplicações: testes automatizados; qualidade de *software*.
- *Application Programming Interface* – API da plataforma *mobile*: arquitetura, recursos e aplicabilidade.
- Persistência de dados em dispositivos móveis: conceito, parâmetros, manipulação de arquivos e gerenciamento de dados em bancos de dados portáteis.
- Fluxo da aplicação: construção e manipulação de telas, comunicação entre componentes do aplicativo; serviços de *background* e comunicação com o sistema.
- Controles visuais: manipulação via código, controles de UI, eventos, alertas, caixas de diálogo, *Webview*.
- Fragmentos de aplicação: implementação de bloco reutilizável, *Fragment* e *Controller*.

Habilidades

- Interpretar requisitos de projetos.
- Utilizar orientação a objeto na codificação.
- Efetuar controle de versão da aplicação *web*.
- Utilizar termos técnicos nas rotinas de trabalho.
- Utilizar *frameworks* de terceiros para desenvolvimento de aplicações para dispositivos móveis.
- Modelar e integrar bancos de dados para dispositivos móveis.
- Testar com máquinas virtuais e emuladores.

Atitudes/Valores

- Sigilo no tratamento de dados e informações.
- Iniciativa na proposição de soluções de projetos.
- Proatividade no desenvolvimento das atividades profissionais.
- Colaboração no desenvolvimento do trabalho em equipe.
- Zelo pela organização do ambiente virtual de trabalho.
- Respeito aos direitos de propriedade intelectual.

Elementos da Competência
<ul style="list-style-type: none"> • Comprometimento com padrões de usabilidade e acessibilidade na <i>web</i>. • Atualização contínua no desenvolvimento das atividades profissionais.

Unidade Curricular 14: Codificar acesso a *web services* e recursos de sistemas móveis

Carga horária: 60 horas

Indicadores
<ol style="list-style-type: none"> 1. Consome serviços <i>web</i> por meio do aplicativo de acordo com funcionalidades da plataforma <i>mobile</i>. 2. Persiste dados na nuvem usando integração do aplicativo com serviços <i>web</i> de acordo com funcionalidades da plataforma <i>mobile</i>. 3. Realiza autenticação a partir de serviços <i>web</i> públicos e redes sociais de acordo com funcionalidades da plataforma <i>mobile</i>. 4. Integra recursos de notificações de acordo com as necessidades do aplicativo e as características do sistema <i>mobile</i>. 5. Integra recursos nativos do dispositivo, de acordo com as necessidades do aplicativo e as características do sistema <i>mobile</i>. 6. Aplica correções e melhorias a partir da validação e depuração do código de integração do <i>web services</i>, conforme necessidades do projeto.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> • Aplicativos RESTful: características e conexão <i>mobile-web</i>. • Programação RESTful: recursos da API <i>mobile</i>, consumo de serviços REST, preparação consulta em rede e tratamento de retorno. • Serviços SOAP: conceito, recursos da API <i>mobile</i> e consumo de serviços. • Segurança em <i>web services</i>: técnicas de autenticação no serviço e recomendações de boas práticas da <i>Open Web Application Security Project</i> (OWASP). • Integração com serviços públicos: serviços de autenticação; integração com rede social; APIs de terceiros. • Armazenamento em nuvem: conceitos, tipos e técnicas de integração. • Notificações: conceitos, tipos e programação com API do sistema <i>mobile</i>. • <i>Push Notification</i>: conceitos, implementação e serviços de envio. • Recursos de conectividade: sensores, NFC, <i>bluetooth</i> e outros.

Elementos da Competência
<ul style="list-style-type: none"> Recursos de navegação: bússola, geolocalização e integração com serviços de mapa. Recursos de mídia: manipulando sons, vídeo e imagem, acesso à câmera e a arquivos. Recursos de sistema: permissões de acesso, sistema de arquivos e integração com serviços nativos. Testes de API: técnicas e ferramentas; <i>mocks</i> e <i>stubs</i> em testes unitários. Versionamento de código: comandos avançados e conceito de integração contínua. <p>Habilidades</p> <ul style="list-style-type: none"> Interpretar requisitos de projetos. Identificar serviços <i>web</i> públicos. Interpretar documentos técnicos. Utilizar termos técnicos na rotina de trabalho. Desenvolver acessibilidade e usabilidade em aplicativos <i>mobile</i>. Efetuar controle de versão da aplicação <i>mobile</i>. Testar aplicação <i>mobile</i>. Aplicar protocolos de segurança no uso dos serviços <i>web</i>. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> Sigilo no tratamento de dados e informações. Colaboração no desenvolvimento do trabalho em equipe. Comprometimento com padrões de usabilidade e acessibilidade. Atualização contínua no desenvolvimento das atividades profissionais. Atitude propositiva no desenvolvimento do trabalho. Respeito aos direitos de propriedade intelectual.

Unidade Curricular 15: Publicar aplicações para dispositivos móveis

Carga horária: 36 horas

Indicadores
<ol style="list-style-type: none"> 1. Compila o <i>app</i> no formato nativo, de acordo com regras e características da loja de aplicativos e requisitos do sistema <i>mobile</i>. 2. Prepara ambiente de venda de acordo com regras e características da loja de aplicativos do sistema <i>mobile</i>. 3. Publica o aplicativo, de acordo com regras e características da loja de aplicativos do sistema <i>mobile</i>.

Elementos da Competência
<p>Conhecimentos</p> <ul style="list-style-type: none"> • Formato de distribuição: especificidades dos principais sistemas operacionais de dispositivos móveis. • Lojas virtuais de aplicativos: conceitos, características e regras; políticas de publicação, tipo de monetização e distribuição de aplicativos. • Visibilidade e impacto da loja de aplicativos: público-alvo do <i>app</i>, tipos de acesso, precificação. • Requisitos da loja para o <i>app</i>: ajustes, ícones, certificado e assinatura digital do aplicativo, otimização de pacote compilado, confecção de tutorial de manipulação do <i>app</i>. • Etapas da publicação: configuração da loja virtual, <i>upload</i> de pacote compilado e disponibilização do aplicativo ao público. • Monitoramento do aplicativo: atualização e versionamento do aplicativo e ferramentas de <i>analytics</i>. <p>Habilidades</p> <ul style="list-style-type: none"> • Interpretar requisitos de projetos. • Interpretar documentos técnicos. • Utilizar termos técnicos na rotina de trabalho. • Realizar processo de empacotamento (<i>build</i>) para dispositivos móveis. • Acompanhar uso do aplicativo publicado. <p>Atitudes/Valores</p> <ul style="list-style-type: none"> • Sigilo no tratamento de dados e informações. • Colaboração no desenvolvimento do trabalho em equipe. • Respeito aos direitos de propriedade intelectual.

Unidade Curricular 5: Projeto Integrador Desenvolvedor *Front-End*

Carga horária: 24 horas

Unidade Curricular 11: Projeto Integrador Desenvolvedor *Back-End*

Carga horária: 32 horas

Unidade Curricular 16: Projeto Integrador Desenvolvedor *Mobile*

Carga horária: 32 horas

O Projeto Integrador é uma Unidade Curricular de Natureza Diferenciada, baseada na

metodologia de ação-reflexão-ação, que se constitui na proposição de situações desafiadoras a serem cumpridas pelo aluno. Esta Unidade Curricular é obrigatória nos cursos de Aprendizagem Profissional Comercial, Qualificação Profissional, Aprendizagem Técnica de Nível Médio, Habilitação Profissional Técnica de Nível Médio, Qualificação Profissional Técnica de Nível Médio e Especialização Técnica de Nível Médio.

O planejamento e execução do Projeto Integrador propiciam a articulação das competências previstas no perfil profissional de conclusão, pois apresentam ao aluno situações que estimulam o seu desenvolvimento profissional ao ter que decidir, opinar e debater com o grupo a resolução de problemas a partir do tema gerador.

Durante a realização do Projeto, portanto, o aluno poderá demonstrar sua atuação profissional pautada pelas marcas formativas do Senac, uma vez que permite o trabalho em equipe e o exercício da ética, da responsabilidade social e da atitude empreendedora.

O Projeto Integrador prevê:

- articulação das competências do curso, com foco no desenvolvimento do perfil profissional de conclusão;
- criação de estratégias para a solução de um problema ou de uma fonte geradora de problemas relacionada com a prática profissional;
- desenvolvimento de atividades em grupos realizadas pelos alunos de maneira autônoma e responsável;
- geração de novas aprendizagens ao longo do processo;
- planejamento integrado entre todos os docentes do curso;
- compromisso dos docentes com o desenvolvimento do Projeto no decorrer das Unidades Curriculares;
- espaço privilegiado para imprimir as Marcas Formativas Senac:
 - domínio técnico-científico;
 - atitude empreendedora;
 - visão crítica;
 - atitude sustentável;
 - atitude colaborativa.

A partir do tema gerador, são necessárias três etapas para a execução do Projeto Integrador:

1ª. Problematização: corresponde ao ponto de partida do Projeto. Na definição do tema gerador, deve-se ter em vista uma situação plausível, identificada no campo de atuação

profissional e que perpassasse as competências do perfil de conclusão. Neste momento, são realizados o detalhamento do tema gerador e o levantamento das questões que nortearão a pesquisa e o desenvolvimento do projeto. As questões devem mobilizar ações que articulem as competências do curso para a resolução do problema.

2ª. Desenvolvimento: para o desenvolvimento do Projeto Integrador, é necessário que os alunos organizem e estruturem um plano de trabalho. Esse é o momento em que são elaboradas as estratégias para atingir os objetivos e dar respostas às questões formuladas na etapa de problematização. O plano de trabalho deve ser realizado conjuntamente pelos alunos e prever situações que extrapolem o espaço da sala de aula, estimulando a pesquisa em bibliotecas, a visita aos ambientes reais de trabalho, a contribuição de outros docentes e profissionais, além de outras ações para a busca da resolução do problema.

3ª. Síntese: momento de organização e avaliação das atividades desenvolvidas e dos resultados obtidos. Nesta etapa, os alunos podem rever suas convicções iniciais à luz das novas aprendizagens e expressar ideias com maior fundamentação teórica e prática, além de gerar produtos de maior complexidade. É importante que a proposta de solução traga aspectos inovadores, tanto no próprio produto quanto na forma de apresentação.

Propostas de Temas Geradores:

UC 5: Projeto Integrador Desenvolvedor *Front-End*

- **Proposta 1: Desenvolvimento de recursos de *front-end* para um sistema multiplataforma com *web services***

Atualmente há uma demanda em compartilhar dados nas mais diversas plataformas como computador, no celular ou no *tablet*, o que traz a necessidade de desenvolvimento de recursos para esta finalidade.

A partir do tema gerador, o docente deverá propor aos grupos desafios que envolvam a construção de um projeto de sistema *web*, que conte com interações via *website* e via *mobile*. Este projeto poderá ser com base em uma situação real, prioritariamente, porém, caso não seja possível, pode-se trabalhar com uma necessidade fictícia de uma empresa, organização ou comunidade. É importante que o docente apresente diversas situações de organizações e segmentos para que os alunos tenham como referência diferentes contextos de mercado.

Na produção do *front-end*, o aluno criará layouts, construirá páginas e montará seu estilo a partir dos requisitos levantados, que pode ser realizado com base em visitas técnicas e entrevistas para esse levantamento.

Para o desenvolvimento do projeto, o aluno utilizará interação com linguagem de *script* e recursos assíncronos, podendo ser utilizados os serviços *mock* de *back-end*, sem as funcionalidades definitivas dos serviços e montados apenas para completar os testes. O docente deverá instigar o aluno a observar a responsividade e a acessibilidade das páginas e das funções incluídas no sistema. Poderá, ainda, buscar, selecionar e utilizar ferramentas (*frameworks*) para estilização e interação das páginas, como Bootstrap e jQuery, por exemplo, porém, sem a necessidade de utilização de *Content Management System* – CMS.

É importante lembrar que o projeto integrador deve ser iniciado com sua documentação. Desta forma, os alunos iniciarão confeccionando documentos importantes para o desenvolvimento do produto final, como por exemplo o *wireframe*, *moke-up*, levantamento de requisitos, logo, paletas de cores e outros que o docente irá estimulando durante o desenvolvimento do projeto.

Seguindo procedimentos de mercado, os testes serão realizados internamente e com o cliente, tornando-se subsídio para ajustes e melhorias no sistema.

- **Proposta 2: Desenvolvimento de um *website* para um sistema de economia compartilhada**

Nos últimos anos, devido às mudanças climáticas e desastres naturais, houve uma crescente preocupação com a possibilidade de escassez de nossos recursos naturais e, conseqüentemente, uma conscientização de que o modo como consumimos produtos e serviços deve ser revisto e conduzido de uma maneira mais sustentável. Considerando o contexto, foram criadas iniciativas que promovem a economia compartilhada, como serviços de transporte, hospedagem em domicílios, aluguel de roupas, etc. Esses serviços são geralmente escaláveis pela internet e precisam de um profissional de desenvolvimento de sistemas *web* e *mobile* capaz de criar *sites* e aplicativos que conectem as pessoas aos serviços.

A partir dessa ideia, o docente deve apresentar ao aluno a contextualização do tema gerador e incentivá-lo a desenvolver um serviço de economia compartilhada pela internet. Primeiramente, o aluno deve definir um problema a ser resolvido e conversar com pessoas que possuam aquele problema. A partir das necessidades propostas pelas pessoas (clientes), os alunos devem definir uma lista de requisitos iniciais que serão utilizados para desenvolver o sistema.

No desenvolvimento da interface *web*, o aluno deve criar as telas que compõem os requisitos apresentados pelo cliente, considerando aspectos de usabilidade e responsividade. Sugere-se, ainda, que o aluno desenvolva um *website* de apresentação do

projeto/organização. Serão utilizadas linguagens de marcação, estilo e *script*, que poderá ser desde as telas de um sistema ou o *website* da apresentação do projeto.

É importante lembrar que o projeto integrador deve ser iniciado com sua documentação. Desta forma, os alunos iniciarão confeccionando documentos importantes para o desenvolvimento do produto final, como por exemplo o *wireframe*, *moke-up*, levantamento de requisitos, logo, paletas de cores e outros que o docente irá estimulando durante o desenvolvimento do projeto.

Durante o processo de desenvolvimento do sistema, poderão ser realizados testes de usabilidade que verificarão a facilidade de acesso e uso do sistema.

UC 11: Projeto Integrador Desenvolvedor *Back-End*

- **Proposta 1: Desenvolvimento de recursos *back-end* para um sistema multiplataforma com *web services***

Tendo em vista a demanda em compartilhar dados nas mais diversas plataformas, a proposta deste tema gerador é dar seguimento ao tema desenvolvido no Desenvolvedor *Front-End*. Partindo do escopo elaborado na etapa de *front-end* do projeto, indica-se criar um sistema com base em serviços que realizem as manipulações de dados e as regras de negócio previstas nos requisitos, levantados na etapa anterior.

Neste contexto, o docente deverá propor estratégias para que a solução desenvolvida em *back-end* se integre com as requisições e as páginas já desenvolvidas no *front-end*. Algumas adaptações, no entanto, podem ser necessárias de maneira a comunicar adequadamente as páginas e os recursos assíncronos com os de *back-end* desenvolvidos. Sugere-se a utilização de *web services* REST, de maneira a completar, nesta etapa, o ciclo de desenvolvimento de uma aplicação RESTful funcional via *web*.

É importante lembrar que o projeto integrador deve ser iniciado com sua documentação. Desta forma, os alunos iniciarão confeccionando documentos importantes para o desenvolvimento do aplicativo final, como por exemplo UML (caso de uso, digrama de classe, diagrama de sequência), fluxogramas, notação básica de caso de uso, *wireframe*, *moke-up*, DER-M, dicionário de dados, *workflow* de API e outros que o docente irá estimulando durante o desenvolvimento do projeto.

Seguindo procedimentos de mercado, testes serão realizados internamente e com o cliente, tornando-se subsídio para ajustes e melhorias no *website* e contribuindo para a qualidade do serviço oferecido.

- **Proposta 2: Desenvolvimento de recursos de *back-end* para um sistema de economia compartilhada**

Nos últimos anos, devido às mudanças climáticas e desastres naturais, houve uma crescente preocupação com a possibilidade de escassez de nossos recursos naturais e, conseqüentemente, uma conscientização de que o modo como consumimos produtos e serviços deve ser revisto e conduzido de uma maneira mais sustentável. Considerando o contexto, foram criadas iniciativas que promovem a economia compartilhada, como serviços de transporte, hospedagem em domicílios, aluguel de roupas, etc. Esses serviços são geralmente escaláveis pela internet e precisam de um profissional de desenvolvimento de sistemas *web* e *mobile* capaz de criar *sites* e aplicativos que conectem as pessoas aos serviços.

A partir dessa ideia, o docente deve apresentar ao aluno a contextualização do tema gerador e incentivá-lo a desenvolver um serviço de economia compartilhada pela internet. Primeiramente, o aluno deve definir um problema a ser resolvido e conversar com pessoas que possuam aquele problema. A partir das necessidades propostas pelas pessoas (clientes), os alunos devem definir uma lista de requisitos iniciais que serão utilizados para desenvolver o sistema.

Os alunos, em equipes, devem desenvolver recursos de *back-end* que englobem os requisitos do sistema. Primeiramente deve ser desenvolvido um banco de dados, que precisará ter estrutura para armazenar todos os dados necessários para o funcionamento do sistema. Além disso, também devem ser desenvolvidas funcionalidades em linguagem de *back-end* que sejam responsáveis por operações administrativas do projeto, como cadastros, relatórios e manipulações de dados.

Durante o processo de desenvolvimento do sistema, poderão ser realizados testes unitários que verificarão o funcionamento dos métodos do sistema, contribuindo para a qualidade dos códigos estruturados.

É importante lembrar que o projeto integrador deve ser iniciado com sua documentação. Desta forma, os alunos iniciarão confeccionando documentos importantes para o desenvolvimento do aplicativo final, como por exemplo UML (caso de uso, digrama de classe, diagrama de sequência), fluxogramas, notação básica de caso de uso, *wireframe*, *moke-up*, DER-M, dicionário de dados, *workflow* de API e outros que o docente irá estimulando durante o desenvolvimento do projeto.

UC 16: Projeto Integrador Desenvolvedor *Mobile*

- **Proposta 1: Desenvolvimento de aplicativo para um sistema multiplataforma com *web services***

Tendo em vista a demanda em compartilhar dados nas mais diversas plataformas, a

proposta deste tema gerador é dar seguimento ao tema realizado no Desenvolvedor *Front-End* e no Desenvolvedor *Back-End*. Após a etapa de *back-end*, já com os serviços *web* criados e comprovadamente funcionais (uma vez que as páginas já estão em operação), deve-se construir, como alternativa e complemento, uma aplicação *mobile* que faça uso desses serviços. Nesta etapa, serão acrescentadas ao sistema novas possibilidades de interação e funcionalidades, como por exemplo notificações e geolocalização.

Para o desenho da aplicação e definição de sua navegação e funcionalidades, um novo projeto de leiaute deve ser desenvolvido, pensando especificamente nesta tecnologia, mas com base em pontos comuns já em vigor no sistema *web*. Deve-se observar com atenção as questões de responsividade e acessibilidade.

É importante lembrar que o projeto integrador deve ser iniciado com sua documentação. Desta forma, os alunos iniciarão confeccionando documentos importantes para o desenvolvimento do aplicativo final, como por exemplo UML (caso de uso, digrama de classe, diagrama de sequência), fluxogramas, notação básica de caso de uso, *wireframe*, *moke-up*, DER-M, dicionário de dados, *workflow* de API e outros que o docente irá estimulando durante o desenvolvimento do projeto.

Seguindo procedimentos de mercado, testes serão realizados internamente e também com o cliente, tornando-se subsídio para ajustes e melhorias no sistema.

- **Proposta 2: Desenvolvimento de um aplicativo *mobile* de economia compartilhada**

Nos últimos anos, devido às mudanças climáticas e desastres naturais, houve uma crescente preocupação com a possibilidade de escassez de nossos recursos naturais e, consequentemente, uma conscientização de que o modo como consumimos produtos e serviços deve ser revisto e conduzido de uma maneira mais sustentável. Considerando o contexto, foram criadas iniciativas que promovem a economia compartilhada, como serviços de transporte, hospedagem em domicílios, aluguel de roupas, etc. Esses serviços são geralmente escaláveis pela internet e precisam de um profissional de desenvolvimento de sistemas *web* e *mobile* capaz de criar *sites* e aplicativos que conectem as pessoas aos serviços.

A partir dessa ideia, o docente deve apresentar ao aluno a contextualização do tema gerador e incentivá-lo a desenvolver um serviço de economia compartilhada pela internet. Primeiramente, o aluno deve definir um problema a ser resolvido e conversar com pessoas que possuam aquele problema. A partir das necessidades propostas pelas pessoas (clientes), os alunos devem definir uma lista de requisitos iniciais que serão utilizados para

desenvolver o sistema.

No desenvolvimento do aplicativo, devem ser criadas telas que englobem todos os requisitos levantados junto aos clientes. Além disso, caso necessário, podem ser utilizados recursos nativos do celular, como câmera ou GPS.

Após a criação das telas, os alunos devem desenvolver um banco de dados que seja armazenado em um servidor *web* e possibilite a interação com o aplicativo. O aplicativo deve se conectar ao serviço *web*, consumindo-o de modo que todos os dados sejam compartilhados entre os diversos usuários da aplicação.

É importante lembrar que o projeto integrador deve ser iniciado com sua documentação. Desta forma, os alunos iniciarão confeccionando documentos importantes para o desenvolvimento do aplicativo final, como por exemplo UML (caso de uso, digrama de classe, diagrama de sequência), fluxogramas, notação básica de caso de uso, *wireframe*, *moke-up*, DER-M, dicionário de dados, *workflow* de API e outros que o docente irá estimulando durante o desenvolvimento do projeto.

Durante o processo de desenvolvimento do sistema, poderão ser realizados testes que verificarão o funcionamento do aplicativo, contribuindo para a qualidade do serviço oferecido.

Com a realização de uma das propostas apresentadas, o aluno poderá demonstrar sua atuação profissional pautada pelas marcas formativas do Senac, uma vez que permite o trabalho em equipe e o exercício da ética, da responsabilidade social e da atitude empreendedora.

Outros temas geradores podem ser definidos em conjunto com os alunos, desde que constituam uma situação-problema e atendam aos indicadores para avaliação.

Indicadores para avaliação:

Para avaliação do Projeto Integrador, são utilizados os seguintes indicadores:

1. Cumpre as atividades previstas no plano de ação, conforme desafio identificado no tema gerador.
2. Apresenta resultados ou soluções, de acordo com as problemáticas do tema gerador e objetivos do Projeto Integrador.

6. Orientações Metodológicas

As orientações metodológicas deste curso, em consonância com a Proposta Pedagógica do Senac, pautam-se pelo princípio da aprendizagem com autonomia e pela metodologia de desenvolvimento de competências, entendidas como *ação/fazer profissional observável*,

potencialmente criativo(a), que articula conhecimentos, habilidades e atitudes/valores e que permite desenvolvimento contínuo.

As competências que compõem a organização curricular do curso foram definidas com base no perfil profissional de conclusão, considerando a área de atuação e os processos de trabalho deste profissional. Para o desenvolvimento das competências, foi configurado um percurso metodológico que privilegia a prática pedagógica contextualizada, colocando o aluno diante de situações de aprendizagem que possibilitam o exercício contínuo da mobilização e articulação dos saberes necessários para a ação e para a solução de questões inerentes à natureza da ocupação.

A mobilização e a articulação dos elementos da competência requerem a proposição de situações desafiadoras de aprendizagem, que apresentem níveis crescentes de complexidade e se relacionem com a realidade do aluno e com o contexto da ocupação.

As atividades relacionadas ao planejamento de carreira dos alunos devem ocorrer de forma concomitante ao desenvolvimento da marca formativa Atitude Empreendedora. Recomenda-se que o tema seja abordado no início das primeiras Unidades Curriculares do curso e revisitado no decorrer de toda a formação. A partir da reflexão sobre si mesmo e sobre a própria trajetória profissional, os alunos podem reconhecer possibilidades de atuação na perspectiva empreendedora e elaborar estratégias para identificar oportunidades e aprimorar cada vez mais suas competências. O docente pode abordar com os alunos o planejamento de carreira a partir dos seguintes tópicos: i) *ponto de partida*: momento de vida do aluno, suas possibilidades de inserção no mercado, fontes de recrutamento e seleção, elaboração de currículo, remuneração oferecida pelo mercado, competências que possui e seu histórico profissional; ii) *objetivos*: o que o aluno pretende em relação à sua carreira a curto, médio e longo prazo; e iii) *estratégias*: o que o aluno deve fazer para alcançar seus objetivos.

Esse plano de ação tem como foco a iniciativa, a criatividade, a inovação, a autonomia e o dinamismo, na perspectiva de que os alunos possam criar soluções e buscar formas diferentes de atuar em seu segmento.

No que concerne às orientações metodológicas para a Unidade Curricular Projeto Integrador (UCPI), recomenda-se que o docente apresente aos alunos o tema gerador da UCPI na primeira semana do curso, possibilitando-lhes modificar e/ou substituir a proposta inicial. Para a execução da UCPI o docente deve atentar para as fases que a compõem: a) problematização (detalhamento do tema gerador); b) desenvolvimento (elaboração das estratégias para atingir os objetivos e dar respostas às questões formuladas na etapa de problematização); e c) síntese (organização e avaliação das atividades desenvolvidas e dos resultados obtidos).

Ressalta-se que o tema gerador deve se basear em problemas da realidade da ocupação, propiciando desafios significativos que estimulem a pesquisa a partir de diferentes temas e ações relacionadas ao setor produtivo ao qual o curso está vinculado. Neste sentido, a proposta deve contribuir para o desenvolvimento de projetos consistentes, que ultrapassem a mera sistematização das informações trabalhadas durante as demais unidades curriculares.

No tocante à apresentação dos resultados o docente deve retomar a reflexão sobre a articulação das competências do perfil profissional e o desenvolvimento das Marcas Formativas, correlacionando-os ao fazer profissional. Deve, ainda, incitar o compartilhamento dos resultados do Projeto Integrador com todos os alunos e a equipe pedagógica, zelando para que a apresentação estabeleça uma aproximação ao contexto profissional. Caso o resultado não atenda aos objetivos iniciais do planejamento, não há necessidade de novas entregas, mas o docente deve propor que os alunos reflitam sobre todo o processo de aprendizagem com o intuito de verificar o que acarretou o resultado obtido.

Orientações metodológicas específicas por Unidade Curricular

UC 1: Elaborar projetos de aplicações para *web*

Nesta unidade curricular, sugere-se a experimentação do fluxo de planejamento de um *website*, desde as análises de mercado, público-alvo, levantamento de informações através do *briefing* e definição de arquitetura da informação, utilizando de exercícios práticos e estudos de caso que contemplem, preferencialmente, situações reais de mercado.

UC 2: Desenvolver aplicações para *websites*

Nesta unidade curricular, sugere-se a experimentação de técnicas de composição visual e codificação para o desenvolvimento de *websites*, através de exercícios que evidenciem conceitos como imagem digital, sistemas de cor, tipografia, *grids*, alinhamentos, *wireframes*, *design* responsivo, linguagens de marcação e estilo e padrões *web*.

Esta UC já traz a necessidade de execução de testes que serão realizados ao longo das demais competências do curso.

As atividades ao longo do curso com base em situações reais devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos.

UC 3: Codificar *front-end* de aplicações *web*

Nesta unidade curricular, sugere-se atividades de codificação utilizando as melhores práticas de desenvolvimento de aplicações para *websites*, com o JavaScript, estimulando sempre o

aprender fazendo e aproximando o curso da realidade na qual os conhecimentos aprendidos poderão ser utilizados. Outra pesquisa poderá ser feita em bibliotecas que utilizem esta linguagem, porém é preciso verificar as bibliotecas atuais e que tenham demanda na região/localidade.

O trabalho com algoritmo que será realizado na UC 6, “Desenvolver algoritmos”, deve iniciar sua contextualização nesta UC, pois será trabalhada uma parte de programação e lógica para seu uso na programação em JavaScript, de maneira que o aluno esteja habilitado a programar, ao menos, pequenas rotinas em *script*.

Indica-se que, para os estudos de funcionalidades assíncronas, sejam disponibilizados recursos previamente programados em *back-end* que sejam acessíveis e que retornem valores adequados para teste. As competências que envolvem *back-end* serão desenvolvidas posteriormente.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso.

UC 4: Publicar aplicações web

Nesta unidade curricular, sugere-se a prática de atividades de pesquisa de servidores de hospedagem que atendam às necessidades do projeto, bem como a prática de transferência de arquivos locais para os servidores de hospedagem via FTP.

UC 6: Desenvolver algoritmos

O trabalho com algoritmos iniciou em “Codificar *front-end* de aplicações web”. Entretanto, será nesta competência que o aluno irá aprofundar a experiência com algoritmos de maneira geral, antes do desenvolvimento do *back-end*.

Por isso, é importante que as práticas e dinâmicas estejam presentes ao longo das aulas. Para isso, propõe-se o uso de jogos e exercícios lúdicos, com tabuleiros, cartas ou ferramentas como o *scratch*, para reforçar a prática e os conceitos da lógica de programação (SCRATCH. Disponível em: scratch.mit.edu. Acesso em: 23 jan. 2020).

Nesta UC serão elaborados pequenos projetos de *softwares*, tendo em vista que, neste momento, a estrutura de algoritmos será reduzida e a proposta é que os alunos aprofundem seu desenvolvimento gradativamente.

A presença de UML se justifica na necessidade de abordar aspectos que extrapolam/complementam o que foi visto na UC 1, “Elaborar projetos de aplicações para web”. Ao invés de *briefing*, aqui o planejamento será de projeto de *software* propriamente dito, mas em linhas gerais, passado de maneira dinâmica ao aluno.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso.

As atividades ao longo do curso com base em situações reais devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos.

UC 7: Codificar *back-end* de aplicações *web*

Sugere-se que, ao apresentar as técnicas e características da programação para *back-end*, o docente incentive os alunos a buscarem situações semelhantes em *websites* já em produção, questionando a tecnologia usada, os conceitos empregados e as funcionalidades desenvolvidas. Propõe-se ainda que o docente forneça projetos que ajudem o aluno a ter exemplos de implementação e que sejam desenvolvidas novas funcionalidades no projeto a partir dos conhecimentos estudados. Também é interessante incluir situações de manutenção para esses projetos (erros a serem corrigidos e adaptações necessárias), permitindo ainda o trabalho juntamente com sistema de versionamento de código.

Indica-se, também, que as tarefas utilizem rotinas previstas em metodologias (como Scrum, XP ou outra à escolha do docente) de maneira a simular um ambiente corporativo.

É possível apresentar modelos que já foram utilizados e testados anteriormente (*Design Patterns*), no sentido de contribuir para as primeiras experiências dos alunos e para organização/manutenção de projetos, além da demonstração de padronização de código.

As atividades ao longo do curso, com base em situações reais, devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos. Estas contribuirão para que, em acordo com a instituição, seja definida a linguagem de programação adotada ao longo do curso.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso. A partir desta unidade curricular o aluno começa a aplicar funcionalidades de versionamento de código, através de um sistema de versionamento selecionado. Esses conceitos e essa prática serão aprofundados no decorrer das UCs que trazem esse tema como conhecimento.

Mesmo o aluno já tendo desenvolvido a competência de publicação, é importante que o docente traga questões para serem aprimoradas/complementadas no desenvolvimento *back-end*.

UC 8: Implementar banco de dados para *web*

Sugere-se entrevistas com profissionais da área ou visitas técnicas a empresas de desenvolvimento que possuam equipe de banco de dados ou um DBA, a fim de que fiquem

claras as atribuições e a importância dos fundamentos da operação com banco de dados. Nesta visita ou entrevista, assim como nas demais atividades do curso, pode-se enfatizar, ainda, as diferentes ferramentas e análises de dados como *Business Intelligence* (BI) e Big Data. A partir de um cenário real, os alunos com apoio do docente, podem, ainda, realizar um trabalho de mapeamento e implementação de um banco de dados. Indica-se que esse banco tenha ligação com projetos desenvolvidos em UCs anteriores, de maneira a conectar as tecnologias e os conhecimentos já aprendidos.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso, assim como sobre o versionamento, iniciado na UC 7.

As atividades ao longo do curso, com base em situações reais, devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos.

Mesmo o aluno já tendo desenvolvido a competência de publicação, é importante que o docente traga questões para serem aprimoradas/complementadas no desenvolvimento *back-end*.

UC 9: Desenvolver serviços *web*

Propõe-se que o docente enumere, junto com os alunos, sistemas que façam utilização de serviços *web*, discutindo as possíveis tecnologias utilizadas. A partir disso, é possível solicitar que os alunos pesquisem projetos e situações que se beneficiam de implementação via serviços *web*, contrapondo com implementações mais tradicionais.

No decorrer do estudo, é importante trazer questões como a segurança do serviço, as vantagens e as desvantagens da abordagem.

O docente pode trabalhar também com os alunos a integração de um sistema *web* a um serviço disponibilizado publicamente, incentivando a leitura e interpretação de documentos técnicos descritivos e analisando os problemas encontrados no decorrer do trabalho.

Mesmo o aluno já tendo desenvolvido a competência de publicação, é importante que o docente traga questões para serem aprimoradas/complementadas no desenvolvimento *back-end*.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso, assim como sobre o versionamento, iniciado na UC 7.

As atividades ao longo do curso, com base em situações reais, devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos.

UC 10: Organizar o processo de trabalho no desenvolvimento de aplicações

Nesta Unidade Curricular indica-se realizar simulações do ambiente de trabalho, trazendo atividades em grupo e desafios que precisem de ações colaborativas para sua resolução. O relacionamento interpessoal, o processo de comunicação e os valores/atitudes da UC são fundamentais para a mobilização das atividades e acompanhamento do cronograma. Este cronograma também poderá ser elaborado ao longo dos projetos e produtos a serem desenvolvidos no curso. Neste momento o olhar é mais administrativo, estimulando o intraempreendedorismo e contribuindo com as premissas e diretrizes das organizações, do que voltado às ações especificamente técnicas do trabalho.

As atividades voltadas ao uso dos *softwares* de escritórios (editor de texto, planilha eletrônica e apresentação de *slides*) têm como objetivo favorecer a criação dos relatórios dessa unidade, bem como gráficos e formas de apresentá-lo.

As simulações de atendimento ao usuário do sistema e os relatórios devem ser elaborados ao longo desta competência, contribuindo e estimulando a comunicação verbal e não verbal.

No elemento respeito à diversidade humana, a proposta é que compreendam a diversidade humana como algo natural. Para tanto, abordagens sobre a diversidade étnica/racial, de gênero e orientações sexuais, biótipos, questões sociais e culturais, entre outras, devem ser trabalhadas por meio de uma abordagem positiva. Indicamos abordar as questões relativas às contribuições da diversidade no ambiente de trabalho. Para isso indicamos:

MCKINSEY & COMPANY. **A diversidade como alavanca de performance**. Disponível em: <https://www.mckinsey.com/business-functions/organization/our-insights/delivering-through-diversity/pt-br>. Acesso em: 12 fev. 2020

UOL. **Jovem trans fatura R\$ 2 milhões ao ano ao levar diversidade a mercado de TI**. Disponível em: <https://www.uol.com.br/ecoa/ultimas-noticias/2019/12/09/jovem-trans-fatura-r-2-milhoes-ao-ano-ao-levar-diversidade-a-mercado-de-ti.htm>. Acesso em: 12 fev. 2020.

ECO.A. **2 mil hackers**. Disponível em: <https://www.uol.com.br/ecoa/reportagens-especiais/causador-vivi-duarte-/index.htm#2-mil-hackers>. Acesso em: 12 fev. 2020.

UOL. **Com 'G10 das favelas' moradores querem atrair investimentos e transformar exclusão em startups**. Disponível em: <https://noticias.uol.com.br/ultimas-noticias/bbc/2019/10/07/com-g-10-das-favelas-moradores-querem-atrair-investimentos-e-transformar-exclusao-em-startups.htm>. Acesso em: 12 fev. 2020.

ALIANÇA NACIONAL LGBTI. Gaylatino. **Manual de comunicação LGBTI+**. Disponível em: <http://www.grupodignidade.org.br/wp-content/uploads/2018/05/manual-comunicacao-LGBTI.pdf>. Acesso em: 14 fev. 2020.

No elemento do conhecimento “Negociação de conflitos”, sugerimos, ainda, uma abordagem que traga a comunicação pela paz (comunicação não violenta). Sugere-se o uso dos materiais a seguir para orientar o planejamento docente:

ITS BRASIL. Secretaria Especial dos Direitos Humanos da Presidência da República. **Módulo II - Violência e não-violência nos direitos humanos**. ITS Brasil; Secretaria Especial dos Direitos Humanos. Disponível em:

http://www.dhnet.org.br/dados/cursos/mediar_conflitos/curso_m_conflitos_modulo_02.pdf.

Acesso em: 14 fev. 2020.

ITS BRASIL. Secretaria Especial dos Direitos Humanos da Presidência da República. **Módulo VI - Formas Não-Violentas de Resolução de Conflitos**. ITS Brasil; Secretaria Especial dos Direitos Humanos. Disponível em:

http://www.dhnet.org.br/dados/cursos/mediar_conflitos/curso_m_conflitos_modulo_06.pdf.

Acesso em: 14 de fev. 2020.

As atividades ao longo do curso com base em situações reais devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos.

UC 12: Desenvolver interface gráfica para dispositivos móveis

Indica-se nesta Unidade Curricular que os alunos aprofundem seus conhecimentos em linguagem de estilo, voltados para comunicação com linguagens de programação (Ionic, Android, Xamarin e outras) para dispositivos móveis, com foco no desenvolvimento das interfaces gráficas.

Recomenda-se, ainda, que os alunos possam pesquisar e discutir sobre as diferenças de um *leiaute mobile* de outros dispositivos. Os alunos, então, farão a implementação a partir de *wireframes* desenvolvidos para dispositivo *mobile*, utilizando recursos visuais das ferramentas e linguagens de programação selecionadas, ao final, emulando ou executando diretamente no dispositivo a fim de realizar testes.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso, assim como sobre o versionamento, iniciado na UC 7.

As atividades ao longo do curso, com base em situações reais, devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos. Estas contribuirão para que, em acordo com a instituição, seja definida a linguagem de programação para dispositivos móveis, adotada ao longo destas UCs.

UC 13: Codificar aplicações para dispositivos móveis

Tendo em vista que o aluno já aprendeu lógica de programação, interface gráfica e JavaScript, o docente deve resgatar estes aprendizados, interligando e aprofundando a linguagem de programação *mobile* no desenvolvimento da aplicação.

Neste momento, o docente deve aprofundar o conhecimento no desenvolvimento do aplicativo, integrando-o ao banco de dados do dispositivo.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso, assim como sobre o versionamento, iniciado na UC 7.

As atividades ao longo do curso, com base em situações reais, devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos, assim como a linguagem de programação que foi definida na UC 12.

UC 14: Codificar acesso a *web services* e recursos de sistemas móveis

Nesta Unidade Curricular, o foco é utilizar *web services* previamente criados (por terceiros ou anteriormente no curso) e integrar o aplicativo aos recursos de *hardware* do dispositivo.

O docente poderá criar serviços genéricos que façam interações com bases de dados e disponibilizá-los aos alunos, para que eles desenvolvam a autenticação e interação através do aplicativo. Na utilização de serviços de autenticação, poderão ser integradas APIs de redes sociais conhecidas.

Na integração de recursos nativos, o docente pode dar ênfase aos mais utilizados, como câmera, arquivos e NFC/*bluetooth*. A ideia é que os alunos entendam como integrá-los e, por meio de exemplos, consigam expandir o conhecimento para os demais recursos do dispositivo.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso, assim como sobre o versionamento, iniciado na UC 7.

As atividades ao longo do curso, com base em situações reais, devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos.

UC 15: Publicar aplicações para dispositivos móveis

Nesta Unidade Curricular, o objetivo é compilar, testar e publicar o aplicativo previamente criado nas demais competências de dispositivos móveis.

O docente deve apresentar as lojas de aplicativos mais utilizadas e as características que as diferem. Em acordo com a Instituição, poderá ser definida uma das lojas, utilizando-a para

exemplificar na prática todo o processo de compilação, criação do executável para publicação final do aplicativo (utilizando chave disponibilizada pela loja), definição de *marketing* (ícone, imagens, descrição) e publicação.

Utilizando o aplicativo do exemplo anterior, também poderão ser publicadas novas versões e analisados dados através de ferramentas de *analytics*, sempre com o foco de apresentar ao aluno um cenário real para que ele aprenda na prática.

A continuidade da aprendizagem na aplicação de testes que iniciou na UC 2 está presente ao longo do desenvolvimento desta e das demais competências do curso, assim como sobre o versionamento, iniciado na UC 7.

As atividades ao longo do curso, com base em situações reais, devem estar pautadas em demandas atuais e da região/localidade onde os alunos estão inseridos.

Orientações metodológicas específicas para as Unidades Curriculares de Natureza Diferenciada

UCs 5, 11 e 16 – Projetos Integradores

O Projeto Integrador deve estimular a pesquisa e a investigação de outras realidades, contextualizadas às demais Unidades Curriculares, assim como a proposição de soluções viáveis aos problemas definidos. É fundamental que o docente responsável pelo desenvolvimento do projeto execute-o em interação com os demais docentes.

Sendo assim, e considerando ainda que o Projeto Integrador é um dos momentos no qual o docente deve trabalhar com as Marcas Formativas Senac, sugere-se que sejam propostos desafios que possibilitem aos alunos a demonstração do domínio técnico-científico, da visão sistêmica e do comportamento investigativo, assim como devem estimular a autonomia, a criatividade e a proatividade nos alunos, fomentar a atitude cidadã e responsável, por meio da reflexão sobre o contexto de trabalho e sobre sua atuação profissional, ajudando-os nas atividades de pesquisa e na sistematização dos dados e contemplando situações de aprendizagem que permitam o trabalho em equipe, no qual irão estabelecer relações interpessoais construtivas.

Os projetos integradores possuem integração por se tratarem de entregas inter-relacionadas que em um ambiente de trabalho envolverão uma equipe de desenvolvimento. Portanto, é possível trabalhar o mesmo projeto que irá evoluir do *front-end* para o *back-end* e *mobile*, ou ainda trabalhar demandas específicas para cada desenvolvimento. O importante é que nas primeiras aulas de cada qualificação o aluno seja levado a entender o contexto de atuação do profissional e possa, em conjunto com os docentes, definir os alinhamentos do tema gerador,

bem como, na sequência, elaborar o cronograma de entregas de cada UC para compor os referidos PIs.

7.

Aproveitamento de Conhecimentos e de Experiências Anteriores

De acordo com a legislação educacional em vigor, é possível aproveitar conhecimentos e experiências anteriores dos alunos, desde que diretamente relacionados com o Perfil Profissional de Conclusão do presente curso.

O aproveitamento de competências anteriormente adquiridas pelo aluno por meio da educação formal, informal ou do trabalho, para fins de prosseguimento de estudos, será feito mediante protocolo de avaliação de competências, conforme as diretrizes legais e orientações organizacionais vigentes.

8.

Avaliação

De forma coerente com os princípios pedagógicos da Instituição, a avaliação tem como objetivos:

- Ser diagnóstica: averiguar o conhecimento prévio de cada aluno e seu nível de domínio das competências, indicadores e elementos, elencar as reais necessidades de aprendizado e orientar a abordagem docente.
- Ser formativa: acompanhar todo o processo de aprendizado das competências propostas neste Plano, constatando se o aluno as desenvolveu de forma suficiente para avançar a outra etapa de conhecimentos e realizando adequações, se necessário.
- Ser somativa: atestar o nível de rendimento de cada aluno, se os objetivos de aprendizagem e competências foram desenvolvidos com êxito e verificar se o aluno está apto a receber seu certificado ou diploma.

8.1. Forma de expressão dos resultados da avaliação

Toda avaliação deve ser acompanhada e registrada ao longo do processo de ensino e aprendizagem. Para tanto, definiu-se o tipo de menção que será utilizada para realizar os registros parciais (ao longo do processo) e finais (ao término da Unidade Curricular/curso).

As menções adotadas no Modelo Pedagógico Senac reforçam o comprometimento com o desenvolvimento da competência e buscam minimizar o grau de subjetividade do processo avaliativo.

De acordo com a etapa de avaliação, foram estabelecidas menções específicas a serem adotadas no decorrer do processo de aprendizagem:

8.1.1. Menção por indicador de competência

A partir dos indicadores que evidenciam o desenvolvimento da competência, foram estabelecidas menções para expressar os resultados de uma avaliação. As menções que serão atribuídas para cada indicador são:

Durante o processo

- Atendido - A
- Parcialmente atendido - PA
- Não atendido - NA

Ao final da Unidade Curricular

- Atendido - A
- Não atendido - NA

8.1.2. Menção por Unidade Curricular

Ao término de cada Unidade Curricular (Competência, Estágio, Prática Profissional, Prática Integrada ou Projeto Integrador), estão as menções relativas a cada indicador. Se os indicadores não forem atingidos, o desenvolvimento da competência estará comprometido. Ao término da Unidade Curricular, caso algum dos indicadores não seja atingido, o aluno será considerado reprovado na unidade. É com base nessas menções que se estabelece o resultado da Unidade Curricular. As menções possíveis para cada Unidade Curricular são:

- Desenvolvida - D
- Não desenvolvida - ND

8.1.3. Menção para aprovação no curso

Para aprovação no curso, o aluno precisa atingir D (desenvolveu) em todas as unidades curriculares (Competências e Unidades Curriculares de Natureza Diferenciada).

Além da menção D (desenvolveu), o aluno deve ter frequência mínima de 75%, conforme legislação vigente. Na modalidade a distância, o controle da frequência é baseado na realização das atividades previstas.

- Aprovado - AP
- Reprovado - RP

8.2. Recuperação

A recuperação será imediata à constatação das dificuldades do aluno, por meio de solução de situações-problema, realização de estudos dirigidos e outras estratégias de aprendizagem que contribuam para o desenvolvimento da competência. Na modalidade de oferta presencial, é possível a adoção de recursos de educação a distância.

9.

Estágio Profissional Supervisionado

O estágio tem por finalidade propiciar condições para a integração dos alunos no mercado de trabalho. É um “ato educativo escolar supervisionado, desenvolvido no ambiente de trabalho, que visa à preparação para o trabalho produtivo de educandos” (Lei nº 11.788/08).

Conforme previsto em legislação vigente, pode integrar ou não a estrutura curricular dos cursos. Será obrigatório quando a legislação que regulamenta a atividade profissional assim o determinar.

Nos cursos em que não for obrigatório, pode ser facultada aos alunos a realização, de acordo com a demanda do mercado de trabalho. Desenvolvido como atividade opcional, a carga horária do estágio é apostilada ao histórico escolar do aluno.

No presente curso, o estágio não é obrigatório.

10.

Instalações, Equipamentos e Recursos Didáticos

10.1. Instalações e equipamentos⁴:

- Para oferta presencial:

Sala de aula convencional mobiliada e equipada com recursos audiovisuais (projektor multimídia ou televisão e caixas de som) e computador.

⁴ É importante que as instalações e equipamentos estejam em consonância com a legislação e atendam às orientações descritas nas normas técnicas de acessibilidade. Estes aspectos, assim como os atitudinais, comunicacionais e metodológicos, buscam atender as orientações da Convenção de Direitos das Pessoas com Deficiência, da qual o Brasil é signatário.

Laboratórios de informática com:

- Computadores com processador com quatro núcleos ou superior, 3,2 GHz ou superior, 16 GB RAM DDR4 2133 MHz, VGA Off-Board com 1 GB, HD 1 TB (recomendável ter um SSD de 128 GB) e suporte à virtualização.
- Celulares ou *tablets* (mínimo Android 6).
- Compiladores/linguagens/tecnologias de programação para o desenvolvimento de *software*: linguagens de programação orientada a objetos, de marcação, de *back-end*, de *script front-end*, folha de estilos e de banco de dados.
- IDEs e *frameworks* de desenvolvimento.
- Ferramentas de desenvolvimento colaborativo e de testes.
- Ferramentas transversais: cliente e servidor de gerenciador de banco de dados relacional; servidor *web* ou emulador de servidor *web*.

- Para oferta a distância:

As configurações de infraestrutura para oferta deste curso a distância serão definidas pelo DR Sede responsável pelo desenvolvimento do título na Rede EaD Senac.

10.2. Recursos didáticos:

O Departamento Regional deve especificar o que será adquirido pelo aluno ou fornecido pelo Senac em caso de alunos do Programa Senac de Gratuidade (PSG).

11.

Perfil do Pessoal Docente e Técnico

O desenvolvimento da oferta ora proposta requer docentes com experiência profissional em desenvolvimento de sistemas e formação em análise e desenvolvimento de sistemas, banco de dados, ciência da computação, engenharia da computação, sistemas de informação ou áreas afins.

Quando da oferta a distância, o DR Sede responsável pela oferta do curso definirá o perfil do tutor.

Unidades Curriculares
<p>UC 1: Elaborar projetos de aplicações para web</p> <p>Carga horária: 36 horas</p> <p><u>Bibliografia Básica</u></p> <p>AGNER, L. Ergodesign e arquitetura de informação: trabalhando com o usuário. São Paulo: Senac, 2018.</p> <p>GEORGE, J.; BEAIRD, J. Princípios do web design maravilhoso. Rio de Janeiro: Alta Books, 2016.</p> <p><u>Bibliografia Complementar</u></p> <p>DIAS, P. R. P. Microsoft Project 2016. São Paulo: Senac, 2016.</p> <p>JOYCE, P.; SILLS, C. Técnicas em Gestalt: aconselhamento e psicoterapia. São Paulo: Vozes, 2016.</p>
<p>UC 2: Desenvolver aplicações para websites</p> <p>Carga horária: 72 horas</p> <p><u>Bibliografia Básica</u></p> <p>GOMES, A. L.; MARTELLI, R. HTML 5 e CSS3. São Paulo: Senac, 2016.</p> <p>SILVA, M. S. CSS Grid Layout: criando layouts CSS profissionais. São Paulo: Novatec, 2017.</p> <p><u>Bibliografia complementar</u></p> <p>ANDRADE, M. S. de. Adobe Photoshop. São Paulo: Senac, 2019.</p> <p>SERAFIM, M. A. Adobe Illustrator. São Paulo: Senac, 2019.</p> <p>TEIXEIRA, F. Introdução e boas práticas em UX Design. São Paulo: Casa do Código, 2014.</p>
<p>UC 3: Codificar front-end de aplicações web</p> <p>Carga horária: 96 horas</p> <p><u>Bibliografia Básica</u></p> <p>ALMEIDA, Flávio. Cangaceiro JavaScript: uma aventura no sertão da programação. São Paulo: Casa do Código, 2017.</p> <p>FREEMAN, Eric; Robson, Elisabeth. Use a cabeça!: programação Javascript. Rio de Janeiro: Alta Books, 2016.</p> <p>IEPSEN, Edécio Fernando. Lógica de programação e algoritmos com JavaScript: uma</p>

Unidades Curriculares
<p>introdução à programação de computadores com exemplos e exercícios para iniciantes. São Paulo: Novatec, 2018.</p> <p><u>Bibliografia Complementar</u></p> <p>ALVES, William Pereira. Desenvolvimento de aplicações web com angular. Rio de Janeiro: Alta Books, 2018.</p> <p>EIS, Diego. Guia front-end: o caminho das pedras para ser um devFront-End. São Paulo: Casa do Código, 2015.</p> <p>MOLINARI, Willian. Desconstruindo a web: as tecnologias por trás de uma requisição. São Paulo: Casa do Código, 2016.</p> <p>NIERADKA, Itamar Pena. Javascript + CSS + DOM: desenvolvimento para web. Rio de Janeiro: Novaterra, 2016.</p>
<p>UC 4: Publicar aplicações web</p> <p>Carga horária: 36 horas</p> <p><u>Bibliografia Básica</u></p> <p>ERINLE, B.; SILVA, A. C. C. [trad.]. Teste de desempenho com Jmeter 3: melhore o desempenho de sua aplicação web. São Paulo: Novatec, 2017.</p> <p>MOLINARI, W. Desconstruindo a web: as tecnologias por trás de uma requisição. São Paulo: Casa do Código, 2016.</p> <p>OLIVEIRA, R.; FERNANDES, N. Apache: instalação, configuração e gestão de servidores web. Portugal: FCA, 2006.</p> <p><u>Bibliografia Complementar</u></p> <p>GOMES, A. L.; MARTELLI, Richard. HTML 5 e CSS3. São Paulo: Senac São Paulo, 2015.</p> <p>ZEMEL, T. Web designer responsivo: páginas adaptáveis para todos os dispositivos. São Paulo: Casa do Código, 2016.</p>
<p>UC 6: Desenvolver algoritmos</p> <p>Carga horária: 96 horas</p> <p><u>Bibliografia Básica</u></p> <p>GUEDES, Gilleanes T. A. UML 2: uma abordagem prática. 3. ed. São Paulo: Novatec, 2018.</p> <p>PIVA JUNIOR, Dilermando; NAKAMITI, Gilberto; ENGELBRECHT, Angela de Mendonça; BIANCHI, Francisco. Algoritmo e programação de computadores. 2. ed. Rio de Janeiro: Elsevier, 2019.</p>

Unidades Curriculares
<p>XAVIER, Gley Fabiano Cardoso. Lógica de programação. 13. ed. São Paulo: Senac São Paulo, 2018.</p> <p><u>Bibliografia Complementar</u></p> <p>ALVES, William Pereira. Desenvolvimento de aplicações web com Wordpress. São Paulo: Érica, 2018.</p> <p>BHARGAVA, Aditya Y. Entendendo algoritmos: um guia ilustrado para programadores e outros curiosos. São Paulo: Novatec, 2017.</p> <p>BOOCH, Grady; JACOBSON, Ivar; RUMBAUGH, James. Uml: guia do usuário. Rio de Janeiro: Campus, 2017. (<i>eBook Kindle</i>).</p> <p>MANZANO, José Augusto N. G.; OLIVEIRA, Jayr Figueiredo de. Algoritmos: lógica para desenvolvimento de programação de computadores. 28. ed. São Paulo: Érica, 2016.</p>
<p>UC 7: Codificar <i>back-end</i> de aplicações web</p> <p>Carga horária: 120 horas</p> <p><u>Bibliografia Básica</u></p> <p>ARAÚJO, Everton Coimbra de. ASP.NET Core MVC: aplicações modernas em conjunto com o Entity Framework. São Paulo: Casa do Código, 2018.</p> <p>MOLINARI, William. Desconstruindo a web: as tecnologias por trás de uma requisição. São Paulo: Casa do Código, 2016.</p> <p>MUELLER, John Paul. Segurança para desenvolvedores web. São Paulo: Novatec, 2016.</p> <p><u>Bibliografia Complementar</u></p> <p>ANDRADE, Sidney da Silva. Aprenda Java EE8: aplicações para web com Spring MVC e Hibernate. São Paulo: Senai Editora, 2018.</p> <p>DALL’OGLIO, Pablo. PHP: programando com orientação a objetos. 4. ed. São Paulo: Novatec, 2018.</p> <p>DEITEL, Harvey; DEITEL, Paul. Java: como programar. 10. ed. São Paulo: Pearson, 2016.</p> <p>POWERS, Shelley. Aprendendo Node: usando JavaScript no servidor. São Paulo: Novatec, 2017.</p> <p>STAUFFER, Matt. Desenvolvendo com Laravel. São Paulo: Novatec, 2017.</p>
<p>UC 8: Implementar banco de dados para web</p> <p>Carga horária: 84 horas</p> <p><u>Bibliografia Básica</u></p>

Unidades Curriculares
<p>CABRAL, Alex de L.; SANTANA FILHO, Ozeas V.; MARTELLI, Richard. Modelagem e banco de dados. São Paulo: Senac São Paulo, 2017.</p> <p>ELMASRI, Ramez; NAVATHE, Shamkant B. Sistemas de banco de dados. São Paulo: Pearson, 2019.</p> <p><u>Bibliografia Complementar</u></p> <p>ARAÚJO, Everton Coimbra de. ASP.NET Core MVC: aplicações modernas em conjunto com o Entity Framework. São Paulo: Casa do Código, 2018.</p> <p>MILANI, André. Construindo aplicações web com PHP e MySQL. 2. ed. São Paulo: Novatec, 2016.</p> <p>PANIZ, David. NoSQL: como armazenar os dados de uma aplicação moderna. São Paulo: Casa do Código, 2016.</p>
<p>UC 9: Desenvolver serviços web</p> <p>Carga horária: 48 horas</p> <p><u>Bibliografia Básica</u></p> <p>LECHETA, Ricardo R. Web services RESTful. São Paulo: Novatec, 2015.</p> <p><u>Bibliografia Complementar</u></p> <p>GABARDO, Ademir C. Laravel para ninjas. São Paulo: Novatec, 2017.</p> <p>MENDES, Marta Talitha; MENDES, Warley Rocha. Java RESTful na prática com JAX –RS. Rio de Janeiro: Ciência Moderna, 2016.</p> <p>SIÉCOLA, Paulo. Web services REST com ASP .NET Web API e Windows Azure. São Paulo: Casa do Código, 2016.</p> <p>SKLAR, David. Aprendendo PHP: introdução amigável à linguagem mais popular da web. São Paulo: Novatec, 2016.</p> <p>WITTIG, Andreas; WITTIG, Michael. Amazon Web Services em ação. São Paulo: Novatec, 2015.</p>
<p>UC 10: Organizar o processo de trabalho no desenvolvimento de aplicações</p> <p>Carga horária: 48 horas</p> <p><u>Bibliografia Básica</u></p> <p>CARVALHAL, Eugenio; NETO, Antonio André. Negociação e administração de conflitos. 5. ed. Rio de Janeiro: FGV, 2017.</p> <p>FARRELL, Larry, C. Atitude empreendedora. Rio de Janeiro: Sextante, 2019.</p>

Unidades Curriculares
<p>TOMASI, C.; MEDEIROS, J. B. Comunicação empresarial. 4. ed. São Paulo: Atlas, 2014.</p> <p><u>Bibliografia Complementar</u></p> <p>FALCONI, Vicente. Gerenciamento da rotina do trabalho do dia a dia. 9. ed. Belo Horizonte: Falconi, 2013.</p> <p>HASHIMOTO, Marcos. Espírito empreendedor nas organizações: aumentando a competitividade através do intra-empendedorismo. São Paulo: Saraiva, 2013.</p>
<p>UC 12: Desenvolver interface gráfica para dispositivos móveis</p> <p>Carga horária: 60 horas</p> <p><u>Bibliografia Básica</u></p> <p>AQUILES, Alexandre; FERREIRA, Rodrigo. Controlando versões com Git e GitHub. São Paulo: Casa do Código, 2014.</p> <p>LOPES, Sergio. A web mobile: design responsivo e além para uma web adaptada ao mundo mobile. São Paulo: Casa do Código, 2013.</p> <p>ZEMEL, Tarcio. Web design responsivo: páginas adaptáveis para todos os dispositivos. São Paulo: Casa do Código, 2015.</p> <p><u>Bibliografia Complementar</u></p> <p>CYBIS, Walter; BETIOL, Adriana Holtz; FAUST, Richard. Ergonomia e usabilidade. 3. ed. São Paulo: Novatec, 2015.</p> <p>SILVA, Maurício Samy. Web design responsivo: aprenda a criar sites que se adaptam automaticamente a qualquer dispositivo, desde desktops até telefones celulares. São Paulo: Novatec, 2014.</p>
<p>UC 13: Codificar aplicações para dispositivos móveis</p> <p>Carga horária: 120 horas</p> <p><u>Bibliografia Básica</u></p> <p>LECHETA, Ricardo R. Google Android: aprenda a criar aplicações para dispositivos móveis com Android SDK. 5. ed. rev. e ampl. São Paulo: Novatec, 2015.</p> <p>MOLINARI, Leonardo. Testes de aplicações mobile: qualidade e desenvolvimento em aplicativos móveis. São Paulo: Érica, 2018.</p> <p>QUERINO FILHO, Luiz Carlos. Desenvolvendo seu primeiro aplicativo. São Paulo: Novatec, 2017.</p> <p><u>Bibliografia Complementar</u></p>

Unidades Curriculares
<p>ALVES, William Pereira. Desenvolvendo aplicações com Xamarin. São Paulo: Novatec, 2017.</p> <p>GOIS, Adrian. Ionic Framework: construa aplicativos para todas as plataformas mobile. São Paulo: Casa do Código, 2017.</p> <p>HARMES, Dan. Desenvolvimento de aplicativos móveis com Xamarin. São Paulo: Novatec, 2015.</p>
<p>UC 14: Codificar acesso a web services e recursos de sistemas móveis</p> <p>Carga horária: 60 horas</p> <p><u>Bibliografia Básica</u></p> <p>LECHETA, Ricardo R. Android essencial. São Paulo: Novatec, 2016.</p> <p>LECHETA, Ricardo R. Google Android: aprenda a criar aplicações para dispositivos móveis com Android SDK. 5. ed. rev. e ampl. São Paulo: Novatec, 2015.</p> <p>QUERINO FILHO, Luiz Carlos. Desenvolvendo seu primeiro aplicativo. São Paulo: Novatec, 2017.</p> <p><u>Bibliografia Complementar</u></p> <p>FOX, Armando; PATTERSON, David. Construindo software como serviço (saas): uma abordagem ágil usando computação em nuvem. Tradução de Fabio Kon e Daniel Cordeiro. San Francisco: Strawberry Canyon LLC, 2015.</p> <p>GOIS, Adrian. Ionic Framework: construa aplicativos para todas as plataformas mobile. São Paulo: Casa do Código, 2017.</p> <p>LECHETA, Ricardo R. Web services Restful: aprenda a criar web services RESTful em Java na nuvem do Google. São Paulo: Novatec, 2015.</p> <p>MITCHELL, Lorna Jane. Web services em PHP: APIs para a web moderna. Tradução de Lúcia Ayako Kinoshita. São Paulo: Novatec, 2013.</p> <p>MOLINARI, Leonardo. Testes de aplicações mobile: qualidade e desenvolvimento em aplicativos móveis. São Paulo: Érica, 2018.</p>
<p>UC 15: Publicar aplicações para dispositivos móveis</p> <p>Carga horária: 36 horas</p> <p><u>Bibliografia Básica</u></p> <p>SILVA, Roger. Entrega contínua em Android: como automatizar a distribuição de apps. São Paulo: Casa do Código, 2016.</p> <p><u>Bibliografia Complementar</u></p>

Unidades Curriculares
<p>GLAUBER, Nelson. Dominando o Android: do básico ao avançado. 2. ed. São Paulo: Novatec, 2015.</p> <p>GOIS, Adrian. Ionic Framework: construa aplicativos para todas as plataformas mobile. São Paulo: Casa do Código, 2017.</p>

13.

Certificação

Àquele que concluir com aprovação as unidades curriculares de 1 a 5 será conferido o certificado de **Qualificação Profissional Técnica de Desenvolvedor *Front-End***, com validade nacional.

Àquele que concluir com aprovação as unidades curriculares 1, 4 e de 6 a 11 será conferido o certificado de **Qualificação Profissional Técnica de Desenvolvedor *Back-End***, com validade nacional.

Àquele que concluir com aprovação as unidades curriculares 1, 10 e de 12 a 16 será conferido o certificado de **Qualificação Profissional Técnica de Desenvolvedor *Mobile***, com validade nacional.

Àquele que concluir com aprovação todas as unidades curriculares que compõem a organização curricular desta Habilitação Profissional Técnica de Nível Médio e comprovar a conclusão do Ensino Médio será conferido o Diploma de **Técnico em Informática para Internet**, com validade nacional.