

REFERENCIAL DE COMPETÊNCIAS

DA QUALIFICAÇÃO

Técnico/a de Desenvolvimento de Software

ÁREA DE EDUCAÇÃO E FORMAÇÃO: 481 – Ciências Informáticas

CÓDIGO DA QUALIFICAÇÃO:

NÍVEL DE QUALIFICAÇÃO: 4

PONTOS DE CRÉDITO: 99,00

PUBLICAÇÕES E ATUALIZAÇÕES:

OBSERVAÇÕES:

DESCRIÇÃO GERAL DA QUALIFICAÇÃO (MISSÃO):

Efetuar a análise de sistemas, a gestão de base de dados, desenvolver aplicações e proceder à instalação e manutenção de equipamentos e aplicações informáticas, assegurando a otimização do seu funcionamento e respeitando as normas de segurança e saúde no trabalho e de proteção do ambiente.

ATIVIDADES PRINCIPAIS:

1. Instalar e configurar equipamentos informáticos isolados ou em rede, bem como proceder à sua manutenção, de acordo com as suas especificações técnicas.
2. Instalar, configurar, parametrizar e gerir sistemas operativos e *software* complementar.
3. Desenvolver e gerir bases de dados, de acordo com as técnicas de análise de sistemas de informação.
4. Utilizar metodologias de planeamento, de desenvolvimento e de gestão de *software*.
5. Desenvolver, distribuir, instalar e efetuar a manutenção de aplicações informáticas, utilizando ambientes e diferentes tipos de linguagens de programação.
6. Desenvolver, instalar e efetuar a manutenção de sistemas de informação baseados nas tecnologias *web* e aplicações para dispositivos móveis.
7. Elaborar documentos e relatórios necessários no desempenho das suas funções.

UNIDADES DE COMPETÊNCIA (UC)

UC OBRIGATÓRIAS

CÓDIGO UC ¹	N.º UC	UNIDADES DE COMPETÊNCIA	PONTOS DE CRÉDITO
UC01989	1	Montar, instalar e manter equipamentos informáticos	4,50
COMUM	2	Instalar, configurar e parametrizar sistemas operativos	4,50
UC01990	3	Instalar, configurar e atualizar software complementar aos sistemas operativos	2,25
COMUM	4	Instalar e interligar redes	2,25
UC00241	5	Configurar protocolos e serviços de rede	4,50
UC00613	6	Gerir políticas de segurança em sistemas informáticos	2,25
UC00602	7	Modelar bases de dados relacionais	2,25
COMUM	8	Desenvolver uma base de dados com linguagem SQL	4,50
	9	Desenvolver uma interface para a base de dados com uma linguagem de programação	4,50
COMUM	10	Planear, instalar e configurar um sistema de gestão de conteúdos	4,50
UC00245	11	Desenvolver algoritmos	2,25
UC00606	12	Desenvolver programas em linguagem estruturada	4,50
UC00607	13	Desenvolver programas complexos em linguagem de programação estruturada	4,50
UC00608	14	Desenvolver programas em linguagem orientada a objetos	4,50
	15	Conceber aplicações para a <i>web</i> na vertente <i>frontend</i>	4,50
	16	Desenvolver programação para a <i>web</i>	4,50
	17	Desenvolver aplicações para dispositivos móveis	4,50
	18	Programar parâmetros de segurança no desenvolvimento de <i>software</i>	2,25
	19	Desenvolver projetos de programação	4,50
COMUM	20	Produzir relatórios e documentos técnicos	2,25
UC00616	21	Implementar as normas de segurança e saúde no trabalho no setor de Informática	2,25
Total de pontos de crédito de UC Obrigatórias			76,50

¹ Os códigos assinalados a preto correspondem a UC específicas de uma qualificação. Os códigos assinalados a laranja correspondem a UC comuns a duas ou mais qualificações.

Para obter a qualificação de Técnico/a de Desenvolvimento de Software, para além das UC Obrigatórias, terão também de ser realizadas UC Opcionais correspondentes ao total de 22,50 pontos de crédito.

UC OPCIONAIS

CÓDIGO UC ²	N.º UC	UNIDADES DE COMPETÊNCIA	PONTOS DE CRÉDITO
UC00033	01	Comunicar e interagir em contexto profissional	4,50
UC00034	02	Colaborar e trabalhar em equipa	4,50
UC00599	03	Interagir em inglês nas atividades do setor da informática	4,50
UC00627	04	Instalar e configurar servidores web	2,25
UC00664	05	Criar bases de dados noSQL	2,25
	06	Desenvolver aplicações com plataformas <i>low code</i>	2,25
UC00669	07	Conceber programas em linguagem C/C++	4,50
	08	Conceber programas complexos em linguagem C/C++	4,50
UC00618	09	Criar aplicações em Linguagem de Programação Python	4,50
UC00619	10	Desenvolver aplicações em linguagem JAVA	4,50
UC00620	11	Desenvolver aplicações em C#	4,50
UC00621	12	Criar aplicações em React	2,25
	13	Criar aplicações em COBOL	2,25
UC00622	14	Criar aplicações em HTML5	2,25
	15	Programar videojogos	4,50
	16	Programar artefactos tangíveis	2,25
UC00623	17	Programar com sistemas de Inteligência Artificial	2,25
UC00643	18	Programar com sistemas baseados na Internet das Coisas (IoT)	2,25
UC00617	19	Utilizar serviços Git e GitHub	2,25
COMUM	20	Editar imagens bitmap	2,25
Total de pontos de crédito da Componente Tecnológica			99,00

² Os códigos assinalados a preto correspondem a UC específicas de uma qualificação. Os códigos assinalados a laranja correspondem a UC comuns a duas ou mais qualificações.

CÓDIGO UC ²	N.º UC	UNIDADES DE COMPETÊNCIA	PONTOS DE CRÉDITO
------------------------	-----------	-------------------------	----------------------

UC 0001/UC01989 Montar, instalar e manter equipamentos informáticos
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Efetuar a instalação dos componentes de um computador e outros equipamentos informáticos. 2. Realizar a manutenção preventiva e corretiva dos equipamentos informáticos.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Informática – fundamentos; terminologia; evolução e marcos tecnológicos relevantes. 2. Manuais do fabricante, guiões e tutorais técnicos. 3. Componentes de <i>hardware</i> – <i>motherboard</i> ; processador; memórias; placa de rede; placas de vídeo; portas de comunicação e fonte de alimentação; periféricos. 4. Outros componentes (cabos, fichas, etc.). 5. Componentes de armazenamento - tipos e características. 6. Orientações e procedimentos de montagem e instalação de equipamento informático. 7. Práticas seguras. 8. Ferramentas de diagnóstico de <i>hardware</i> - multímetros, utilitários de sistema, <i>software</i> de teste de memória e discos. 9. Manutenção preventiva do <i>hardware</i> - limpeza do equipamento; verificação de cablagens e de componentes. 10. Normas de gestão de resíduos. 11. Normas de proteção ambiental. 12. Regulamento geral de proteção de dados (RGPD). 13. Normas de segurança e saúde no trabalho.	1. Interpretar manuais do fabricante, guiões e tutorais técnicos dos diversos componentes de <i>hardware</i> e de montagem de equipamentos informáticos. 2. Utilizar as orientações técnicas de manipulação de ferramentas e equipamentos. 3. Selecionar e reunir ferramentas e equipamentos. 4. Aplicar técnicas e orientações de montagem, instalação, atualização e configuração de equipamentos informáticos. 5. Instalar e ligar periféricos. 6. Instalar o <i>software</i> necessário para o funcionamento dos componentes de <i>hardware</i> . 7. Efetuar a limpeza interna e externa do equipamento. 8. Verificar o estado da cablagem e dos componentes. 9. Substituir a massa térmica. 10. Aplicar procedimentos de segurança no manuseamento e na montagem dos componentes, dos equipamentos informáticos. 11. Usar aplicações para analisar e medir o desempenho do computador. 12. Testar o funcionamento de componentes instalados. 13. Aplicar regras de segurança na montagem e manutenção dos componentes dos equipamentos informáticos.	1. Responsabilidade partilhada pelas suas ações. 2. Autonomia limitada no âmbito das suas funções. 3. Cooperação com a equipa. 4. Empenho e persistência na resolução de problemas. 5. Iniciativa. 6. Sentido crítico. 7. Sentido de organização. 8. Respeito pelas regras e normas definidas.

	14. Aplicar as normas de gestão de resíduos. 15. Aplicar normas de proteção ambiental. 16. Aplicar o Regulamento Geral de Proteção de Dados. 17. Aplicar as normas de segurança e saúde no trabalho.	
--	--	--

CRITÉRIOS DE DESEMPENHO

Montar, instalar e manter equipamentos informáticos:

1. Garantindo os requisitos técnicos e de compatibilidade de componentes.
2. Seguindo as técnicas e orientações de instalação dos componentes.
3. Cumprindo os procedimentos de manuseamento e segurança.
4. Testando o funcionamento para deteção e correção de problemas.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática, redes e telecomunicações
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Dispositivos eletrónicos com acesso à *internet*.
2. Manuais do fabricante, guiões e tutorais técnicos.
3. Software de medição de desempenho do equipamento informático.
4. Computadores, equipamentos, ferramentas e materiais do laboratório de eletrónica.
5. Componentes de equipamentos informáticos e outros dispositivos.
6. Ferramentas e máquinas para montagem de componentes.
7. Normas, regras técnicas e legislação aplicável.

OBSERVAÇÕES

UC 0002/0000 Instalar, configurar e parametrizar sistemas operativos
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Instalar sistemas operativos nos equipamentos informáticos. 2. Parametrizar os recursos dos sistemas operativos		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Manuais do fabricante, guiões e tutoriais técnicos. 2. Sistemas operativos - requisitos e funcionalidades. 3. Equipamento: características. 4. Instalação do sistema operativo a partir de periféricos. 5. Parâmetros na instalação dos sistemas operativos. 6. Periféricos - definição e configuração. 7. Normas de gestão da segurança da informação. 8. Ambiente de trabalho. 9. Gestão de processos e de memória. 10. Sistema de ficheiros. 11. Gestão de recursos. 12. Privilégios de acesso. 13. Normas e regulamentos aplicáveis	1. Interpretar manuais e tutoriais técnicos de configuração dos sistemas operativos e dos diversos componentes. 2. Operar com diferentes sistemas operativos. 3. Reconhecer a sequência de operações a realizar. 4. Definir as configurações de arranque da máquina. 5. Definir as opções de configuração do sistema operativo e a forma de interação com periféricos e outros componentes do equipamento. 6. Consultar o gestor de dispositivos para a visualização do estado dos componentes. 7. Utilizar os recursos do sistema para otimizar o rendimento das aplicações 8. Definir e gerir como os dados são escritos, armazenados e recuperados. 9. Definir e gerir a partilha de recursos. 10. Definir e gerir a partilha de ficheiros. 11. Definir os privilégios de acesso para diferentes perfis de utilizador. 12. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções 3. Assertividade na comunicação. 4. Cooperação com a equipa. 5. Empenho e persistência na resolução de problemas. 6. Iniciativa. 7. Sentido crítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Instalar, configurar e parametrizar sistemas operativos:

1. Cumprindo orientações e procedimentos técnicos nas diversas fases.
2. Cumprindo com as recomendações técnicas e as necessidades do utilizador.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

Manuais, guiões e tutoriais técnicos.

Equipamentos informáticos

Sistemas operativos.

Normas nacionais ou internacionais de segurança da informação.

OBSERVAÇÕES

UC 0003/UC01990 Instalar, configurar e atualizar *software* complementar aos sistemas operativos

PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Configurar as ferramentas de sistema. 2. Executar <i>software</i> para criar imagens de restauro e <i>backup</i> do sistema. 3. Executar a manutenção, a atualização e o restauro do sistema. 4. Configurar mecanismos de segurança e proteção nos sistemas operativos.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Manuais do fabricante, guiões e tutoriais técnicos. 2. Ferramentas de manutenção e otimização do sistema operativo - tipos; funções; configuração 3. Backup - definição; tipos; ferramentas de backup; agendamento e execução de backups. 4. Procedimentos de restauração do sistema. 5. Atualizações do sistema - gestão e configuração de atualizações automáticas. 6. Mecanismos de segurança - ferramentas, políticas de proteção e combate a ameaças informáticas. 7. Antivírus - tipos; instalação; configuração. 8. Ameaças informáticas - Tipos; deteção e eliminação de vírus. 9. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Utilizar as funcionalidades para configurar as ferramentas de sistema num equipamento informático. 3. Configurar as opções de segurança. 4. Criar imagens de restauro. 5. Configurar agendamentos para garantir <i>backups</i> automáticos. 6. Restaurar o sistema a partir de <i>backups</i> ou imagens de restauro. 7. Instalar um <i>software</i> antivírus. 8. Configurar o antivírus e <i>firewall</i>. 9. Detetar a existência de vírus e outras ameaças. 10. Aplicar normas e regulamentos aplicáveis.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Cooperação com a equipa. 4. Empenho e persistência na resolução de problemas. 5. Iniciativa. 6. Sentido crítico. 7. Sentido de organização. 8. Rigor. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Instalar, configurar e atualizar software complementar aos sistemas operativos:

1. Garantindo os requisitos técnicos e de compatibilidade.
2. Seguindo as orientações e procedimentos técnicos.
3. Testando o funcionamento para deteção e correção de problemas.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Equipamentos informáticos.
3. Sistema operativo instalado.
4. *Software* de gestão de cópias de sistema.

OBSERVAÇÕES

UC 0004/0000 Instalar e interligar redes

PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Preparar cablagem de redes. 2. Montar equipamentos passivos de rede. 3. Ligar e testar equipamentos passivos de rede.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Manuais do fabricante, guíões e tutoriais técnicos. 2. Redes – tipos, características, topologias, elementos, mapas físicos e lógicos. 3. Equipamentos ativos e passivos de uma rede. 4. Técnicas para cravar fichas. 5. Técnicas para montar calhas e tomadas de rede. 6. Técnicas de fusão de fibra ótica. 7. Protocolos de comunicação. 8. Transmissão de dados. 9. Testes de continuidade e desempenho dos cabos. 10. Instalação de equipamentos passivos de rede. 11. Normas de segurança e saúde no trabalho. 12. Normas de proteção ambiental. 13. Normas e regulamentações na instalação de redes.	1. Interpretar manuais e tutoriais técnicos. 2. Distinguir diferentes tipos de cablagem. 3. Descarnar cabos e cravar fichas de rede. 4. Realizar a fusão de fibra ótica. 5. Selecionar os equipamentos para criar uma rede. 6. Preparar a infraestrutura de rede (calhas, tomadas, régua, bastidores, entre outros equipamentos). 7. Operar com ferramentas e maquinaria necessária para a instalação e montagem de equipamentos de rede. 8. Instalar os componentes físicos da rede. 9. Ligar e testar os componentes físicos da rede. 10. Aplicar as normas de segurança e saúde no trabalho. 11. Aplicar as normas de proteção ambiental. 12. Aplicar as normas e regulamentações na instalação de redes.	1. Responsabilidade partilhada pelas suas ações. 2. Autonomia limitada no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido de organização. 6. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Instalar e interligar redes:

1. Assegurando a compatibilidade entre os equipamentos.
2. Cumprindo as orientações técnicas para preparar e montar equipamentos de rede.
3. Cumprindo as orientações técnicas para ligar e testar equipamentos de rede.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática, redes e telecomunicações.

2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Placas de rede, routers, *switchs*, *entre outros*.
3. Cabos
4. Fichas
5. Ferramentas e máquinas
6. Computadores
7. Outros equipamentos

OBSERVAÇÕES

UC0005/UC00241

Configurar protocolos e serviços de rede

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Configurar os parâmetros dos protocolos de rede. 2. Configurar os serviços de rede. 3. Configurar os serviços da Internet.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Manuais do fabricante, guiões e tutoriais técnicos. 2. Redes – tipos, características, topologias, elementos, mapas físicos e lógicos. 3. Equipamentos ativos e passivos de uma rede. 4. Protocolos de rede (TCP/IP, HTTPS, FTP, SMTP, DNS, SSL/TLS, UDP). 5. Modelo OSI 6. Modelo TCP/IP 7. Endereçamento IP 8. Redes, sub-redes e máscaras de rede 9. Procedimentos de configuração 10. Serviços de rede. 11. Utilizadores: perfis e gestão. 12. Regulamento Geral de Proteção de Dados (RGPD) 13. Normas e regulamentos na instalação de redes	14. Interpretar manuais, guiões e tutoriais técnicos. 15. Identificar o modelo OSI 16. Identificar o modelo TCP/IP 17. Utilizar endereçamento IP 18. Configurar roteamento 19. Configurar a gestão dos protocolos 20. Instalar o serviço de domínio. 21. Instalar o servidor de <i>email</i> . 22. Configurar o servidor de <i>email</i> . 23. Criar grupos de trabalho. 24. Definir e gerir utilizadores. 25. Aplicar as normas e regulamentos na instalação de redes.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas 4. Iniciativa. 5. Respeito pelas normas de segurança. 6. Sentido crítico. 7. Sentido de organização.

CRITÉRIOS DE DESEMPENHO

Configurar protocolos e serviços de rede:

1. Selecionando opções para os serviços de rede.
3. Cumprindo orientações técnicas.
4. Cumprindo as normas e regulamentos na instalação de redes

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática, redes e telecomunicações.
2. Lojas de informática.
3. Serviços de apoio técnico.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Computadores e outros equipamentos de rede.
3. Sistema operativo de rede

OBSERVAÇÕES

UC0006/UC00613 Gerir políticas de segurança em sistemas informáticos

PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Configurar mecanismos de segurança no sistema de rede. 2. Monitorizar a segurança do sistema. 3. Realizar cópias de segurança.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Segurança da informação – conceitos, vulnerabilidades, ameaças e ataques; políticas e mecanismos de segurança; segurança em sistemas distribuídos; criptografia, gestão de chaves. 2. <i>Firewall</i> – tipologia, implementação. 3. Sistemas de deteção de intrusões (IDS) – arquitetura, classificação e aplicação. 4. Redes privadas virtuais (VPN) – tipos, dispositivos, túneis e protocolos, estabelecimentos. 5. Regulamento geral de proteção de dados. 6. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Instalar ferramentas de monitorização. 3. Utilizar ferramentas de diagnóstico e análise de redes. 4. Configurar parâmetros de rede. 5. Aplicar autenticação e criptografia a ligações de rede. 6. Executar as operações técnicas necessárias para a segurança dos equipamentos ligados à rede. 7. Gerir uma <i>firewall</i>. 8. Planear e configurar a realização de <i>backups</i> de segurança. 9. Configurar redes privadas virtuais. 10. Detetar anomalias decorrentes de ataques ou tentativas de ataques. 11. Bloquear conteúdos indevidos. 12. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas 4. Iniciativa. 5. Respeito pelas normas de segurança. 6. Sentido crítico. 7. Sentido de organização.

CRITÉRIOS DE DESEMPENHO

Gerir políticas de segurança em sistemas informáticos:

1. Aplicando medidas de segurança relativas à confidencialidade, integridade e disponibilidade de dados.
2. Operando com as ferramentas de análise do estado e monitorização do sistema.
3. Planeando a realização de cópias de segurança com regularidade.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática e redes.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.

2. Dispositivos tecnológicos com acesso à Internet.
3. Servidor.
4. Software de monitorização.

OBSERVAÇÕES

UC0007/UC00602 Modelar bases de dados relacionais
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Definir a estrutura de uma base dados relacionais 2. Representar modelos relacionais. 3. Normalizar dados não normalizados.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Sistemas de Gestão de Bases de Dados (SGBD) – conceitos. 2. Bases de dados - conceito de dados e de modelo de dados; arquitetura; ficheiros. 3. Arquitetura de um sistema de gestor de base de dados. 4. Modelo relacional- estrutura de dados relacional; regras de integridade; gestão de dados. 5. Tabelas, registos, campos e chaves. 6. Modelo de entidades e relações: conceitos e tipo de atributos. 7. Modelos físicos de dados 8. Representação das fronteiras do sistema 9. Representação do comportamento do sistema 10. Representação da implementação do sistema 11. Normalização - representação na forma não normalizada; tipo de notação; integridade da informação. 12. Regime Geral de Proteção de Dados 13. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os conceitos sobre a gestão da informação em bases de dados. 3. Identificar as entidades e os seus atributos. 4. Utilizar modelos relacionais para modelar e representar a informação. 5. Utilizar modelos de diagramas para modelar e representar a informação. 6. Usar regras e notações para representação das fases da normalização de dados. 7. Aplicar as regras que contribuem para a integridade da informação. 8. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Modelar bases de dados relacionais:

1. Seguindo orientações metodológicas para modelar a informação.
2. Cumprindo as fases de normalização de dados em diversas notações.
3. Cumprindo as normas e regulamentos aplicáveis.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática, redes e telecomunicações.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

1. Manuais, guíões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações para a criação de modelos relacionais e normalização de dados.

OBSERVAÇÕES

UC 0008/0000
Desenvolver uma base de dados com linguagem SQL
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear a estrutura de uma base de dados. 2. Criar e manipular tabelas relacionais e consultas. 3. Manipular registos na base de dados.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Sistemas de Gestão de Bases de Dados (SGBD) - conceitos. 2. Modelo relacional. 3. Modelação de informação. 4. Servidor virtual. 5. Linguagem SQL - comandos; funções; operadores 6. Base de dados- tabelas e integridade de dados; fundamentos de <i>transact</i> SQL; filtragem, ordenação e agrupamento de dados. 7. Funcionalidades de um editor de texto. 8. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Utilizar modelos relacionais para modelar a informação. 3. Elaborar diagramas para modelar a informação. 4. Instalar e executar o servidor local. 5. Utilizar os comandos, funções e operadores de uma linguagem SQL. 6. Criar tabelas. 7. Definir os campos das tabelas e tipo de dados. 8. Inserir, consultar, alterar e eliminar dados. 9. Definir o tipo de relações entre tabelas. 10. Criar consultas com critérios e filtros. 11. Criar guião técnico e o manual de utilizador da base de dados.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Desenvolver uma base de dados com linguagem SQL:

1. Cumprindo especificações metodológicas para estruturar a informação.
2. Cumprindo especificações para criar tabelas e definir as relações entre tabelas.
3. Garantindo a integridade dos dados.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.

3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações para a criação de base de dados.
4. Aplicações para a criação de diagramas.
5. Servidor.
6. Editor de texto.

OBSERVAÇÕES

UC 0009/0000

 Desenvolver uma *interface* para a base de dados com uma linguagem de programação

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Explorar funções, instruções e sintaxe da linguagem de programação. 2. Planear a estrutura da <i>interface</i> para a base de dados. 3. Criar o <i>layout</i> de acesso à base de dados. 4. Ligar a <i>interface</i> com a base de dados.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. <i>Websites</i> dinâmicos - conceitos. 2. Estrutura de <i>layouts/templates</i> . 3. Ambiente de desenvolvimento de programação. 4. Linguagem de programação - conceitos, funções, instruções e sintaxe. 5. <i>Frameworks</i> . 6. Métodos de ligação à base de dados. 7. Arquitetura de componentes de acesso base de dados. 8. <i>Interface</i> de acesso a dados. 9. Manipulação de registos - adicionar; eliminar; atualizar; procurar. 10. Segurança - práticas de proteção e autenticação. 11. Funcionalidades de um editor de texto. 12. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Utilizar as metodologias de planeamento de <i>websites</i> para definir a estrutura da <i>interface</i> . 3. Reconhecer linguagens de programação para o desenvolvimento de <i>interfaces</i> para bases de dados. 4. Aplicar as regras de programação, funções, instruções e sintaxe no desenvolvimento da <i>interface</i> . 5. Configurar e implementar conexões entre a <i>interface</i> e a base de dados. 6. Manipular registos na base de dados. 7. Utilizar <i>frameworks</i> no desenvolvimento de <i>software</i> para <i>web</i> . 8. Testar a <i>interface</i> desenvolvida. 9. Detetar e corrigir erros. 10. Criar um guião técnico ou manual de utilizador do programa. 11. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

 Desenvolver uma *interface* para a base de dados com uma linguagem de programação:

- Cumprindo as especificações técnicas na definição da estrutura dos *layouts*.
- Cumprindo as regras de programação, funções, instruções e a sintaxe.

3. Assegurando que todas as conexões entre a interface e a base de dados estejam protegidas por mecanismos de segurança adequados.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações para fazer base de dados.
4. Ambientes integrados de desenvolvimento.
5. Aplicações para desenhar *layout*.
6. *Frameworks*
7. Servidor.
8. Editor de texto.

OBSERVAÇÕES

UC 0010/0000 Planear, instalar e configurar um sistema de gestão de conteúdos

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear a estrutura de páginas e separadores num sistema de gestão de conteúdos (CMS). 2. Configurar temas/<i>templates</i>, <i>plugins</i> e <i>widgets</i> num CMS. 3. Manusear páginas, menus e artigos num CMS. 4. Inserir, atualizar e remover conteúdos nas páginas do CMS.		
CONHECIMENTOS	APTIDÕES	ATITUDES
1. Metodologias de planeamento e prototipagem de <i>websites</i>. 2. Estrutura de uma página <i>web</i>. 3. Domínio <i>web</i> - conceitos; tipos de domínios. 4. Serviço de alojamento <i>web</i>. 5. Servidor virtual. 6. Criação e publicação de <i>websites</i>: conceitos. 7. Sistemas de Gestão de Conteúdos (CMS). 8. Painel de controlo um CMS - <i>templates</i> /temas; personalização; ferramentas; menus/separadores; páginas; artigos; multimédia; <i>plugins</i>; <i>Widgets</i>; ligações às redes sociais. 9. Conteúdos para a <i>web</i> - ícones; botões; texto; imagens; vídeo; formulários e mapas. 10. Normas e práticas de referênciação e licenciamento dos direitos de autor, propriedade intelectual e <i>copyright</i>. 11. Cópias de segurança. 12. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Utilizar metodologia de organização de <i>websites</i> para desenhar o diagrama com o mapa do site. 3. Utilizar a técnica de <i>wireframing</i> para criação de um protótipo do <i>layout</i> das páginas e organização dos conteúdos. 4. Registrar um domínio na <i>web</i>. 5. Criar conta e configurar um alojamento <i>web</i>. 6. Instalar um CMS. 7. Selecionar, ativar, personalizar e configurar o tema. 8. Adicionar e ativar <i>plugins</i>. 9. Utilizar as funcionalidades do painel de controlo para configurar e editar <i>widgets</i>. 10. Adicionar, editar e remover páginas. 11. Configurar os menus. 12. Inserir, atualizar e remover conteúdos (textos, imagens, vídeo, formulários, mapas, documentos, etc...) nas páginas. 13. Atribuir a autoria na utilização de textos, imagens, sons, vídeos e outros artefactos digitais. 14. Efetuar cópias de segurança de um CMS.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

	15. Aplicar normas e regulamentos.	
--	------------------------------------	--

CRITÉRIOS DE DESEMPENHO

Planear, instalar e configurar um sistema de gestão de conteúdos:

1. Cumprindo as especificações técnicas definidas.
2. Testando o funcionamento das principais funcionalidades.
3. Cumprindo o licenciamento dos direitos de autor de acordo com a finalidade dos conteúdos.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Domínio na *web*.
4. Alojamento *web*.
5. Servidor virtual.
6. Sistema de Gestão de Conteúdos (CMS).
7. *Software* de edição de imagem.
8. *Software* de edição de vídeo.

OBSERVAÇÕES

UC0011/UC00245 Desenvolver algoritmos
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Definir o problema. 2. Planejar as etapas de criação do algoritmo. 3. Estruturar algoritmos em pseudocódigo. 4. Desenhar algoritmos em fluxograma. 5. Testar e depurar algoritmos.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios. 2. Algoritmo – conceitos, noções de ação e estado da ação; etapas e desenvolvimento. 3. Tipos de dados – constantes e variáveis. 4. Entrada e saída de dados – elementos de linguagem. 5. Estruturas lógicas básicas – estrutura sequencial, alternativa e repetitiva; condições e regras de inicialização e alteração; estruturas diagramáticas como representação algorítmica. 6. Técnicas de construção/desenho de algoritmos – contadores, totalizadores, expressões aritméticas, funções predefinidas, validação de dados.	1. Reconhecer os princípios do pensamento computacional. 2. Definir os inputs e os outputs esperados, as restrições e as condições que o algoritmo deve cumprir. 3. Decompor um problema em subproblemas ou etapas menores. 4. Aplicar estruturas de dados, estruturas lógicas e técnicas de construção de algoritmos. 5. Utilizar aplicações de desenho de algoritmos. 6. Utilizar métodos de teste e depuração de algoritmos. 7. Aplicar estratégias de otimização de algoritmos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas 4. Iniciativa. 5. Rigor 6. Sentido analítico. 7. Sentido de organização. 8. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Desenvolver algoritmos:

1. Aplicando as técnicas de construção
2. Utilizando aplicações de representação diagramática.
3. Garantindo a resolução do problema.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.

2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Editor de texto.
4. Ambientes integrados de desenvolvimento.
5. Compiladores.
6. Aplicações de desenho de algoritmos e fluxogramas.

OBSERVAÇÕES

UC0012/UC00606 Desenvolver programas em linguagem estruturada
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento de um programa. 2. Criar programas com funções e estruturas de controlo. 3. Testar e depurar os programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Ciclo de vida do software. 2. Pensamento computacional - princípios. 3. Desenvolvimento de software - metodologias. 4. Algoritmos. 5. Linguagem estruturada- conceitos; características; estrutura de um programa. 6. Dados - variáveis, declarações e expressões; constantes; tipos de dados simples. 7. Estruturas de controlo: sequência; seleção; repetição 8. Subprogramas - estrutura (funções e procedimentos); variáveis locais e globais; passagem de variáveis por parâmetros. 9. Funcionalidades de um editor de texto. 10. Regulamento geral de proteção de dados. 11. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios e conceitos relacionados com algoritmia. 3. Interpretar os princípios e conceitos relacionados com programação estruturada. 4. Utilizar orientações metodológicas para planear as etapas de criação do programa. 5. Instalar e configurar o ambiente de programação. 6. Utilizar a sintaxe da linguagem no desenvolvimento do programa. 7. Definir os tipos de dados, operadores, constantes, variáveis e estruturas de controlo no desenvolvimento. 8. Definir funções e estruturas de dados. 9. Testar e depurar o programa. 10. Detetar e corrigir os erros identificados. 11. Criar um guião técnico ou manual de utilizador do programa. 12. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Sentido critico. 7. Sentido de organização. 8. Rigor. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Desenvolver programas em linguagem estruturada:

1. Seguindo as orientações técnicas e metodológicas no desenvolvimento de *software*.
2. Utilizando o ambiente de programação.

3. Cumprindo as regras de programação.
4. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações de programação estruturada.
4. Ambientes integrados de desenvolvimento.
5. Compiladores.
6. Editor de texto.

OBSERVAÇÕES

UC0013/UC00607

Desenvolver programas complexos em linguagem de programação estruturada

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento de software. 2. Criar programas com estruturas de dados. 3. Testar e depurar os programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional: Princípios 2. Algoritmos. 3. Desenvolvimento de <i>software</i> : Metodologias. 4. Ambiente de desenvolvimento. 5. Estruturas de dados estáticas - <i>strings</i> ; <i>Arrays</i> ; <i>Arrays</i> multidimensionais. 6. Estruturas de dados compostas - conceitos fundamentais; acesso e manipulação. 7. Estruturas de dados dinâmicas - apontadores; acesso e manipulação. 8. Ficheiros - acesso e manipulação. 9. Funcionalidades de um editor de texto. 10. Regulamento geral de proteção de dados. 11. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais de programação. 2. Interpretar os princípios do pensamento computacional. 3. Reconhecer as características da programação estruturada. 4. Utilizar metodologias de desenvolvimento de <i>software</i> . 5. Utilizar os tipos de dados, operadores, funções, procedimentos e variáveis. 6. Utilizar estruturas de dados estáticas. 7. Utilizar estruturas de dados compostas. 8. Utilizar estruturas de dados dinâmicas. 9. Definir o acesso e manipulação de ficheiros. 10. Testar e depurar o programa. 11. Corrigir os erros identificados. 12. Criar um guião técnico ou manual de utilizador do programa. 13. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações e pelas de terceiros. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Sentido crítico. 7. Sentido de organização. 8. Rigor. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Desenvolver programas complexos em linguagem de programação estruturada:

1. Seguindo as orientações metodológicas no desenvolvimento de *software*.
2. Cumprindo a especificidade da linguagem.
3. Cumprindo as regras de programação na utilização de estruturas de dados complexas.
4. Executando a programação e corrigindo os erros detetados.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.

3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações de programação estruturada.
4. Ambientes de desenvolvimento de programação.
5. Compiladores.
6. Editor de texto.

OBSERVAÇÕES

UC0014/UC00608	Desenvolver programas em linguagem orientada a objetos
----------------	--

PONTOS DE CRÉDITO: 4,5

REALIZAÇÕES		
1. Aplicar metodologias de desenvolvimento de <i>software</i> . 2. Estruturar os programas. 3. Aplicar a sintaxe da linguagem de programação. 4. Testar e depurar programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios 2. Algoritmos. 3. Desenvolvimento de <i>software</i> - metodologias. 4. Ambiente de desenvolvimento. 5. Programação orientada por objetos – paradigma; características (encapsulamento; partilha de comportamento; evolução) 6. Classes, objetos, métodos, atributos e mensagens entre objetos. 7. Relações entre classes. 8. Relações entre Objetos. 9. Herança e Polimorfismo. 10. Classes Genéricas. 11. Interfaces. 12. Funcionalidades de um editor de texto. 13. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Criar algoritmos. 4. Utilizar orientações metodológicas para planear as etapas de desenvolvimento de uma aplicação. 5. Reconhecer as características da programação orientada por objetos. 6. Utilizar elementos e sintaxe de uma linguagem de programação orientada por objetos para desenvolver aplicações. 7. Desenhar as interfaces. 8. Testar e corrigir os erros identificados durante os testes ao programa. 9. Criar um guião técnico ou manual de utilizador da aplicação. 10. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor. 7. Sentido analítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Desenvolver programas em linguagem orientada a objetos:

- Cumprindo as fases metodológicas no desenvolvimento de *software*.
- Cumprindo a especificidade da linguagem
- Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

- Empresas do setor da informática.

2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambiente de desenvolvimento de programação.
4. Máquina virtual.
5. Compiladores.
6. Editor de texto.

OBSERVAÇÕES

UC 0015/0000

Conceber aplicações para a *web* na vertente *frontend*

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
- Definir parâmetros da estrutura da página <i>web</i> com a linguagem de marcação <i>web</i>. - Programar estilos CSS (<i>Cascadin Style Sheets</i>) - Aplicar a linguagem de programação JavaScript à manipulação do DOM. - Desenhar e desenvolver o <i>frontend</i> de uma aplicação <i>web</i> responsiva.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
- Desenvolvimento <i>web</i> - história e tecnologias associadas. - Páginas <i>web</i> - conceitos, organização, desenvolvimento, criação e publicação; responsividade - Construção base de páginas <i>Web</i> - estrutura de páginas <i>web</i>; etiquetas comuns em páginas <i>web</i>; hiperligações; integração de imagens; propriedades e formatação de páginas <i>web</i>. - Tabelas. - Formulários. <i>Cascading Style Sheets</i> (CSS). <i>Javascript</i> aplicados a páginas <i>Web</i>. - Bibliotecas <i>frontend</i>. - Acessibilidade <i>web</i>. - Domínio <i>web</i>. - Alojamento <i>web</i>. - Ferramenta de FTP. - Normas e regulamentos aplicáveis.	- Interpretar manuais, guiões e tutoriais técnicos. - Aplicar metodologias de organização de websites para desenvolver e estruturar um diagrama de mapa do site. - Utilizar a técnica de <i>wireframing</i> para criação de um protótipo dos <i>layouts</i> das páginas e organizar o tipo de conteúdos. - Aplicar as regras de organização de páginas <i>web</i>, para criar a estrutura em HTML. - Programar marcadores em HTML para integrar os vários elementos na página <i>web</i>. - Programar hiperligações em HTML para criar ligações entre as páginas de um <i>website</i>. - Utilizar as regras de sintaxe para definir estilos CSS. - Utilizar procedimentos técnicos para integrar a linguagem de estilos CSS em documentos HTML. - Utilizar <i>frameworks</i> <i>web</i>. - Programar com elementos e sintaxe da linguagem de programação JavaScript. - Aplicar regras de acessibilidade na programação de <i>interfaces web</i>.	- Responsabilidade pelas suas ações. - Autonomia no âmbito das suas funções. - Empenho e persistência na resolução de problemas. - Iniciativa. - Rigor. - Sentido analítico. - Sentido de organização. - Respeito pelas regras e normas definidas.

	12. Registrar um domínio na <i>web</i>. 13. Criar conta e configurar um alojamento <i>web</i>. 14. Instalar uma ferramenta <i>File Transfer Protocol</i> (FTP). 15. Executar <i>upload</i> de ficheiros via <i>ftp</i> para o alojamento. 16. Testar a aplicação. 17. Corrigir os erros identificados durante os testes. 18. Aplicar normas e regulamentos.	
--	---	--

CRITÉRIOS DE DESEMPENHO

Conceber aplicações para a web na vertente frontend:

1. Cumprindo as regras de programação de marcadores em HTML para inserir os vários elementos.
2. Cumprindo as regras e a sintaxe na programação de estilos *em CSS*.
3. Cumprindo as regras e a sintaxe na programação de *scripts em JavaScript*.
4. Executando o *frontend* de uma aplicação *web responsiva* e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Bloco de notas.
4. Ambiente de desenvolvimento do Javascript.
5. Domínio na web.
6. Alojamento web.
7. Ferramenta de FTP

OBSERVAÇÕES

UC 0016/0000 Desenvolver programação para a web
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Criar conteúdo dinâmico e ativo com recurso a uma linguagem de <i>scripting server-side</i> . 2. Manipular bases de dados através da <i>web</i> .		
CONHECIMENTOS	APTIDÕES	ATITUDES
1. Pensamento computacional - princípios. 2. Páginas <i>web</i> - conceitos, organização, desenvolvimento, criação e publicação. 3. Programação para a <i>Web</i> - conceitos; arquitetura das aplicações para a <i>web</i> . 4. Linguagens de <i>Scripting</i> - diferenças da linguagem face ao padrão; validações locais (no cliente); animações (no cliente); programação orientada a eventos; processamento de informação no cliente (<i>cookies</i> e outras técnicas). 5. Interface com Bases de Dados - ferramentas de acesso a base de dados; SQL em consultas, inserção, modificação e eliminação de dados; saída da base de dados num <i>browser</i> . 6. Conteúdo dinâmico no servidor. 7. Conteúdo ativo no cliente. 8. Funcionalidades de um editor de texto.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Reconhecer as tecnologias <i>web</i> do lado servidor. 3. Distinguir entre código <i>server-side</i> e <i>cliente-side</i> . 4. Aplicar conceitos e técnicas de programação para a <i>web</i> . 5. Utilizar elementos e a sintaxe de uma linguagem de <i>scripting server-side</i> para criar conteúdo dinâmico. 6. Utilizar ferramentas de acesso a bases de dados. 7. Utilizar SQL em consultas, inserção, modificação e eliminação de dados. 8. Configurar a saída de base de dados num <i>browser</i> . 9. Criar conteúdo dinâmico do lado servidor. 10. Integrar conteúdo ativo no cliente. 11. Criar um guião técnico ou manual de utilizador do programa.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Rigor. 6. Sentido analítico. 7. Sentido de organização. 8. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Desenvolver programação para a web:

1. Identificando e corrigindo erros de execução e sintaxe.
2. Estabelecendo a ligação segura entre script e a base de dados.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.

3. Serviços de apoio técnico.

4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambiente de desenvolvimento de programação.

OBSERVAÇÕES

UC 0017/0000 Desenvolver aplicações para dispositivos móveis
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento da <i>app</i> . 2. Instalar e configurar o ambiente de desenvolvimento de programação. 3. Criar <i>interfaces</i> e conteúdos para dispositivos móveis. 4. Executar testes de navegação e de compatibilidade.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Desenvolvimento de software - ciclo de vida. 2. Pensamento computacional: princípios. 3. Desenvolvimento de apps - princípios; conceitos. 4. <i>Wireframing</i> . 5. Arquitetura de <i>Apps</i> . 6. Plataforma de conceção e desenvolvimento - principais plataformas de desenvolvimento (Android, IOS, etc.); ambientes de desenvolvimento; ferramenta de desenvolvimento. 7. Aplicações - desenho de interfaces da aplicação; programação da aplicação; conteúdos do dispositivo móvel; interatividade na aplicação; gestão de dados. 8. Testes, exportação e publicação: navegação e de compatibilidade; exportação para diferentes plataformas; publicação e disponibilização da aplicação. 9. Funcionalidades de um editor de texto. 10. Normas e regulamentos aplicáveis.	1. Interpretar os princípios do pensamento computacional. 2. Utilizar as metodologias de planeamento de apps para planear a estrutura. 3. Utilizar a técnica de <i>wireframing</i> para criação de um protótipo dos layouts dos ecrãs, organizar o tipo de conteúdos e ligações. 4. Utilizar orientações técnicas na instalação e configuração do ambiente de programação. 5. Adotar técnicas de programação aos recursos de um smartphone. 6. Criação interfaces para dispositivos móveis. 7. Utilizar as regras, os elementos e a sintaxe da linguagem de programação. 8. Adaptar rotinas de programação para diferentes fins (comunicação, gráficos, sensores, localização, etc.). 9. Implementar elementos interativos na aplicação. 10. Usar orientações técnicas para a realização de testes de navegação e compatibilidade. 11. Corrigir erros de programação que surgem durante os testes. 12. Exportar a aplicação para diferentes plataformas.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Rigor. 6. Sentido analítico. 7. Sentido de organização. 8. Respeito pelas regras e normas definidas.

	13. Criar um guião técnico ou manual de utilizador do programa. 14. Aplicar normas e regulamentos.	
--	--	--

CRITÉRIOS DE DESEMPENHO

Desenvolver aplicações para dispositivos móveis:

1. Cumprindo o planeamento das diferentes etapas de desenvolvimento da *app*.
2. Manuseando o ambiente de desenvolvimento de *apps*.
3. Cumprindo regras no uso dos elementos e sintaxe da programação.
4. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Dispositivos móveis.
4. Ambiente de programação.
5. Ambiente de desenvolvimento integrado.
6. Emulador.
7. Aplicação de edição de imagem.
8. Editor de texto.

OBSERVAÇÕES

UC 0018/0000 Programar parâmetros de segurança no desenvolvimento de *software*
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Aplicar os princípios de segurança no desenvolvimento de <i>software</i> . 2. Executar testes de segurança de <i>software</i> .		
Conhecimentos	Aptidões	ATITUDES
1. Desenvolvimento de <i>software</i> - ciclo de vida. 2. Desenho de código seguro. 3. Riscos de segurança em aplicações <i>web</i> - injeção; quebras de autenticação; exposição de dados sensíveis; entidades externas XML; controle em quebras de acessos. 4. Configuração incorreta de segurança. 5. Outras quebras de segurança. 6. Testes de segurança de <i>software</i> . 7. Cópias de segurança. 8. Partilhas e permissões de acesso. 9. Conteúdos indevidos. 10. Funcionalidades de um editor de texto ou apresentações multimédia. 11. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar políticas e regras de segurança associadas ao MFA por serviço/aplicação e dados sensíveis. 3. Aplicar regras de desenho de código seguro no desenvolvimento da programação. 4. Definir regras e níveis de permissões de acesso. 5. Aplicar princípios de segurança na programação da autenticação das aplicações. 6. Executar práticas de segurança nos <i>scripts</i> para proteger contra vulnerabilidades e acesso não autorizado. 7. Configurar a realização de <i>backups</i> de segurança. 8. Criar um guião ou manual de utilizador de procedimentos de segurança no desenvolvimento de <i>software</i> . 9. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor. 7. Sentido analítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
*Programar parâmetros de segurança no desenvolvimento de *software*:*

1. Cumprindo regras e princípios de programação de código seguro.
2. Executando procedimentos na realização de testes de segurança.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.

4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos tecnológicos com acesso à internet.
3. Editor de texto ou apresentações multimédia.
4. Regulamento Geral de Proteção de Dados.

OBSERVAÇÕES

UC 0019/0000 Desenvolver projetos de programação

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento de um projeto de programação. 2. Estruturar dados, desenhar fluxogramas e protótipos. 3. Executar o programa. 4. Executar testes de funcionamento.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Ciclo de vida do <i>software</i>. 2. Pensamento computacional - princípios. 3. Algoritmia - princípios. 4. Desenvolvimento de <i>software</i> - metodologias 5. <i>Wireframing</i>. 6. Ambiente de desenvolvimento. 7. Fases de desenvolvimento de um programa - identificação do problema; elaboração do algoritmo; codificação; <i>debug</i>; testes de funcionamento; otimização algorítmica. 8. Elaboração de fluxogramas 9. Sintaxe e semântica. 10. Servidor <i>web</i>. 11. Funcionalidades de um editor de texto. 12. Normas e regras aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Utilizar orientações metodológicas para planear as etapas de desenvolvimento de um projeto de <i>programação</i>. 4. Aplicar técnicas para elaborar a modelação de dados. 5. Aplicar técnicas de elaboração de fluxogramas. 6. Instalar e configurar o ambiente de programação. 7. Utilizar a técnica de <i>wireframing</i> para criação de um protótipo dos <i>layouts</i> dos <i>screens</i>/ecrãs, organizar o tipo de conteúdos e ligações. 8. Utilizar elementos e sintaxe de uma linguagem de programação para implementar o programa. 9. Testar e depurar o programa. 10. Corrigir os erros identificados durante os testes ao programa. 11. Criar documentação técnica de suporte ao programa. 12. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor. 7. Sentido analítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Desenvolver projetos de programação:

1. Cumprindo as fases metodológicas no desenvolvimento do projeto de programação.
2. Cumprindo regras no uso dos elementos e sintaxe da linguagem de programação.
3. Detetando e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações de desenho de algoritmos, diagramas e prototipagem.
4. Ambientes integrados de desenvolvimento de programação.
5. Editor de texto.

OBSERVAÇÕES

UC 0020/0000 Produzir relatórios e documentos técnicos
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Elaborar relatórios sobre planos de trabalho e atividades realizadas. 2. Criar manuais de utilizador. 3. Elaborar manuais de procedimentos e normas de segurança.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Modelos de folhas de registo, fichas técnicas, relatórios técnicos e manuais do utilizador. 2. Técnicas para a elaboração de relatórios técnicos: 3. Registos de suporte aos relatórios. 4. Técnicas de elaboração de manuais de procedimentos. 5. Sistema de <i>tickets</i> . 6. Segurança da informação: normas de gestão e segurança da informação. 7. Funcionalidades de um editor de texto ou folha de cálculo. 8. Normas e regulamentos aplicáveis.	1. Interpretar as informações a contemplar num relatório técnico. 2. Distinguir os vários modelos de folhas de registo, fichas técnicas, relatórios técnicos e manuais do utilizador. 3. Reconhecer a importância dos relatórios técnicos para a monitorização de projetos. 4. Criar modelos de relatórios. 5. Organizar a informação técnica a incluir nos relatórios. 6. Registar informação sobre as intervenções feitas. 7. Reconhecer o funcionamento do sistema de <i>tickets</i> . 8. Utilizar um sistema de <i>tickets</i> . 9. Planear as fases de redação, revisão e atualização dos manuais. 10. Interpretar as normas de segurança e gestão de informação. 11. Utilizar as funcionalidades de um editor de texto ou folha de cálculo para relatórios e documentos técnicos. 12. Aplicar normas e padrões técnicos na criação e formatação de documentos. 13. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções e atribuições. 3. Empenho e persistência na resolução de problemas. 4. Rigor. 5. Sentido crítico. 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Produzir relatórios e documentos técnicos:

1. Organizando e atualizando os registos dos planos de trabalhos no decorrer das atividades.
2. Cumprindo os padrões definidos na elaboração da documentação técnica.
3. Mantendo atualizados manuais de utilizador e de segurança.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Lojas de informática.
3. Serviços de apoio técnico.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos tecnológicos com acesso à Internet.
3. *Software* para registo de serviços
4. Editor de texto
5. Folha de cálculo.

OBSERVAÇÕES

UC 0021/UC00616 Implementar as normas de segurança e saúde no trabalho no setor de Informática
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Analisar os princípios gerais sobre segurança e saúde no trabalho. 2. Aplicar medidas e procedimentos de segurança e saúde no trabalho.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Princípios de segurança e saúde no trabalho. 2. Normas e disposições relativas à segurança e saúde no setor da informática – legislação. 3. Plano de segurança do estabelecimento. 4. Plano de prevenção de acidentes. 5. Plano de prevenção de incêndios. 6. Plano de evacuação. 7. Plano contra roubos. 8. Manuais de segurança. 9. Meios e regras de segurança na informática – 10. Equipamentos de proteção individual (EPI), métodos de supressão da negligência e falta de atenção, proteção de equipamentos de trabalho e ergonomia. 11. Regras de segurança no manuseamento de equipamento e na movimentação de materiais - normas do vestuário, prevenção de choques elétricos, movimentação de cargas pesadas. 12. Causas de acidentes no trabalho - acidentes de movimentação, choques e quedas, acidentes provocados por ferramentas e máquinas em movimento, choques	1. Identificar as normas relativas à segurança e saúde no trabalho. 2. Interpretar o plano de segurança do estabelecimento. 3. Reconhecer os manuais de segurança. 4. Aplicar medidas de prevenção do risco. 5. Aplicar os procedimentos em caso de acidente de trabalho. 6. Aplicar os procedimentos de emergência. 7. Aplicar medidas de prevenção de roubo. 8. Distinguir os diferentes tipos de incêndio e respetivos sistemas de deteção e de extinção. 9. Aplicar medidas de prevenção de incêndios. 10. Utilizar o extintor. 11. Utilizar equipamentos de proteção individual. 12. Reportar a situação de emergência.	1. Responsabilidade pelas suas ações e pelas de terceiros. 2. Autonomia no âmbito das suas funções. 3. Autocontrolo. 4. Autocontrolo. 5. Sentido de organização. 6. Cooperação com a equipa. 7. Respeito pelas normas de segurança.

elétricos, acidentes provocados por agentes químicos e queimaduras.		
13. Caixa de primeiros socorros.		
14. Situações de emergência - perda de sentidos, feridas aberta e fechada, choque elétrico, eletrocussões, ataque cardíaco, entorses ou distensões, envenenamento, queimaduras.		
15. Causas de incêndio - sistema de aquecimento e cozedura, chaminé e tubos de fumo, materiais inflamáveis, aparelhos elétricos, trabalhadores e outras pessoas fumadoras.		
16. Tipos de incêndio.		
17. Sistemas de deteção.		
18. Tipos de extintores.		
19. Incêndio - plano de ataque, manipulação de extintores, acionamento do sistema automático.		
20. Técnicas de extinção de incêndio de gás.		

CRITÉRIOS DE DESEMPENHO

Implementar as normas de segurança e saúde no trabalho no setor de Informática:

1. Considerando os tipos de risco existentes no posto de trabalho e respetivas medidas de segurança e preventivas.
2. Cumprindo as medidas de atuação em situação de emergência.
3. Respeitando o protocolo interno definido.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Organizações do setor da informática.
2. Espaço do cliente.

RECURSOS

1. Dispositivos tecnológicos com acesso à Internet.
2. Legislação sobre segurança e saúde no trabalho.
3. Normativos específicos de segurança e saúde no trabalho.
4. Documentação sobre segurança e saúde no trabalho (relatórios, folhetos, brochuras, outros).
5. Equipamentos de proteção individual (EPI).
6. Planos de prevenção de acidentes, de incêndios, de evacuação e de roubo.

7. Planos de emergência.
8. Equipamento de sinalização.

UNIDADES DE COMPETÊNCIA OPCIONAIS

UC0001/UC00033	Comunicar e interagir em contexto profissional
-----------------------	---

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Preparar a mensagem a comunicar em contexto profissional. 2. Informar e esclarecer diferentes interlocutores em contexto presencial e não presencial.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Princípios da comunicação e do relacionamento interpessoal – processo, funções e elementos intervenientes. 2. Fatores facilitadores e inibidores da comunicação. 3. Comunicação verbal (oral e escrita) e comunicação não-verbal – cinésica (movimentos corporais, gestos, expressão facial e postura), paralinguística (tom, projeção da voz, pausas no discurso, sorriso, outros) e proxémica (distância espacial face a alguém). 4. Canais de comunicação presencial e não presencial. 5. Comunicação telefónica - técnicas de atenção telefónica, expressão verbal e sorriso “telefónico”. 6. Comunicação através da Internet (navegadores, email, redes sociais, mensagens) – técnicas. 7. Comunicação escrita – normas. 8. Processo de escrita – planificação, textualização e revisão. 9. Características dos estilos de comunicação – agressivo, passivo, manipulador, assertivo. 10. Comunicação assertiva – vantagens, componentes	1. Organizar a informação a comunicar. 2. Adaptar a comunicação oral e escrita ao interlocutor e ao contexto. 3. Interpretar informação de diferentes interlocutores em contexto presencial e não presencial. 4. Identificar as expectativas do interlocutor. 5. Utilizar técnicas de comunicação verbal e não verbal assertiva. 6. Formular questões, pedir esclarecimentos ou colocar dúvidas para interpretar e/ou explicitar a mensagem. 7. Partilhar informação com diferentes interlocutores. 8. Reportar informação profissional. 9. Aplicar técnicas de interação orais e escritas. 10. Aplicar técnicas de tratamento e resolução de conflitos. 11. Autoavaliar o seu desempenho no âmbito do processo de comunicação.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Cuidado com a imagem e postura profissional. 4. Assertividade. 5. Escuta ativa. 6. Empatia. 7. Controlo emocional. 8. Autoconfiança. 9. Respeito pela diferença. 10. Autoconhecimento. 11. Sentido crítico. 12. Cooperação com a equipa. 13. Sentido de organização.

verbais e não-verbais, técnicas. 11. Escuta ativa, empatia e controlo emocional. 12. Processamento interno da informação – fonético, literal (significado) e reflexivo (empático). 13. Perguntas no processo de comunicação – abertas, fechadas, retorno, reformulação. 14. Mensagem - construção, adaptação, envio, receção e interpretação. 15. Imagem e comunicação – autoimagem e autoconceito, primeiras impressões, expectativas e motivação. 16. Técnicas de programação neurolinguística (PNL) na comunicação. 17. Relações interpessoais no trabalho. 18. Conflito nas relações interpessoais – tipos e técnicas de resolução de conflitos. 19. Avaliação do processo de comunicação – feedback, resposta e reação.		
--	--	--

CRITÉRIOS DE DESEMPENHO

Comunicar e interagir em contexto profissional:

1. Adaptando a linguagem e a comunicação ao tipo de canal utilizado, ao público-alvo e ao contexto.
2. Demonstrando assertividade e uma imagem positiva de si e da sua organização.
3. Demonstrando uma comunicação verbal e não verbal empática e ajustada ao interlocutor.
4. Produzindo um texto escrito de forma clara e articulada, de acordo com a norma, aplicando técnicas de redação de documentos profissionais.
5. Avaliando o resultado do seu desempenho e contributo para a melhoria do processo de comunicação.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Aplicável a diferentes contextos.

RECURSOS

1. Dispositivos tecnológicos com acesso à Internet.
2. Recursos multimédia e audiovisuais.

3. Ferramentas de interação e de comunicação.
4. Boas práticas na comunicação.

OBSERVAÇÕES

5. Sugere-se que esta UC seja desenvolvida com recurso a métodos expositivos, nomeadamente, demonstração de casos práticos para análise
6. Esta UC poderá ser realizada em regime não presencial devido à sua vincada vertente teórica.

UC0002/UC00034 Colaborar e trabalhar em equipa
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Analisar a identidade pessoal e partilhada e respetivos comportamentos associados. 2. Colaborar na aplicação de dinâmicas facilitadoras do trabalho em equipa. 3. Colaborar na definição de estratégias de resolução de problemas e de tomada de decisão.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Identidade pessoal, social e profissional. 2. Fenómenos da dinâmica de grupo - influência social e papel social, normas sociais, atitudes e comportamentos facilitadores e dificultadores, padrão de grupo e motivação individual. 3. Trabalho em equipa - fatores pessoais, relacionais e organizacionais. 4. Equipa de trabalho - princípios de organização de grupo vs. equipa de trabalho, estilos comportamentais, estrutura e fases de desenvolvimento da equipa, perceção de desempenho individual, formas e técnicas de organização, cooperação e colaboração. 5. Comunicação assertiva - verbal e não-verbal, fatores facilitadores e inibidores. 6. Canais de comunicação presencial e não presencial. 7. Importância da comunicação no trabalho entre equipas - fluxos de comunicação, comunicação vertical e horizontal, <i>feedback</i> do desempenho. 8. Técnicas de negociação, de resolução de problemas e de tomada de decisão.	1. Identificar e analisar os estilos comportamentais individuais. 2. Identificar as competências individuais. 3. Identificar os papéis dos membros da equipa - competências e responsabilidades. 4. Reconhecer a fase de desenvolvimento de competências na qual a equipa se encontra. 5. Identificar os valores e as principais competências necessários para a equipa atingir o(s) objetivo(s) traçado(s). 6. Colaborar na definição dos mecanismos de coesão e controlo na equipa. 7. Colaborar na definição de tarefas e prazos para alcançar os objetivos traçados. 8. Participar na execução de tarefas predefinidas para a equipa. 9. Aplicar técnicas de comunicação em diferentes contextos. 10. Utilizar ferramentas de comunicação. 11. Partilhar informação presencialmente e/ou <i>online</i> .	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Autoconhecimento. 4. Automotivação. 5. Assertividade. 6. Empatia. 7. Escuta ativa. 8. Cooperação com a equipa. 9. Empenho e persistência na resolução de problemas. 10. Sentido crítico. 11. Sentido criativo. 12. Flexibilidade e adaptabilidade. 13. Disponibilidade para aprender. 14. Respeito e valorização das diferenças individuais. 15. Respeito pela sensibilidade e bem-estar dos outros. 16. Respeito pelas regras e normas definidas.

9. Gestão de tempo – técnicas, planeamento, autoavaliação e otimização das tecnologias. 10. Trabalho <i>online</i> ou teletrabalho - condições facilitadoras, equipas 4D e atitude partilhada. 11. Saúde no trabalho - síndrome de <i>burnout</i>. 12. Organização das equipas na área profissional.	12. Formular ideias e sugestões em diferentes contextos comunicacionais. 13. Trocar conhecimentos e experiências. 14. Identificar os princípios subjacentes à tomada de decisão. 15. Analisar problemas e tomar decisões. 16. Desenvolver rotinas em equipa em momentos formais, informais, presenciais e online. 17. Reconhecer sinais de <i>burnout</i> próprio e/ou dos colegas.	
--	---	--

CRITÉRIOS DE DESEMPENHO

Colaborar e trabalhar em equipa:

1. Mobilizando os recursos pessoais para a obtenção dos melhores resultados da equipa.
2. Aplicando técnicas de comunicação e negociação adequadas aos interlocutores e ao contexto.
3. Analisando problemas e propondo soluções.
4. Gerando oportunidades de desenvolvimento e aprendizagem colaborativa.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Aplicável a diferentes contextos.

RECURSOS

1. Dispositivos tecnológicos com acesso à Internet.
2. Ferramentas de interação, de comunicação e produtividade.
3. Recursos multimédia e audiovisuais.
4. Boas práticas na comunicação.

UC0003/UC00599

Interagir em inglês nas atividades do setor da informática

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Interpretar e selecionar informação especializada, verbal e não verbal, em suportes variados nas atividades do setor da informática 2. Transmitir enunciados orais coerentes no âmbito das atividades no setor da informática. 3. Redigir textos articulados e coesos relacionados com as atividades no setor da informática.		
CONHECIMENTOS	APTIDÕES	ATITUDES
1. Léxico (vocabulário) relacionado com a Informática 2. Funções da linguagem. 3. Estruturas do funcionamento da língua – sons, entoações e ritmos da língua, símbolos fonéticos; nomes, pronomes, adjetivos, advérbios, determinantes e artigos, elementos de ligação frásica, verbos. 4. Sintaxe. 5. Fluência de leitura. 6. Regras de produção de documentos escritos. 7. Regras de cortesia e convenções linguísticas.	1. Identificar o sentido de mensagens em contexto profissional e reconhecer léxico específico da área profissional num discurso oral. 2. Descodificar perguntas e informações. 3. Distinguir informação essencial da informação acessória em textos e suportes diversificados. 4. Responder a perguntas diretas. 5. Iniciar, manter e terminar conversas de âmbito profissional. 6. Descrever, narrar e expressar pontos de vista num discurso oral. 7. Redigir notas, mensagens, relatórios e preencher formulários. 8. Escrever ou responder a uma carta, e-mail e outro tipo de mensagens. 9. Utilizar vocabulário específico da área profissional. 10. Adequar o código oral e escrito à sua finalidade.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Assertividade. 4. Empatia 5. Empenho e persistência na resolução de problemas. 6. Escuta ativa. 7. Respeito pelas diferenças individuais. 8. Sentido crítico. 9. Respeito pelas regras e normas definidas.

	11. Identificar sequência e causalidade. 12. Contextualizar o texto no tempo e no espaço. 13. Respeitar as regras da morfologia e da sintaxe na produção oral e escrita. 14. Usar linguagens não verbais. 15. Mobilizar recursos linguísticos relacionando informação de áreas e fontes diversificadas. 16. Utilizar procedimentos de pesquisa e recolha de informação.	
--	--	--

CRITÉRIOS DE DESEMPENHO

Interagir em inglês nas atividades do setor da informática:

1. Identificando o contexto, a ideia principal, distinguindo informações simples e de maior complexidade do discurso oral e do texto escrito.
2. Comunicando oralmente de forma precisa e eficaz, com ritmo e entoação apropriados e adaptando o discurso ao registo do interlocutor.
3. Utilizando vocabulário, estruturas frásicas diversas e formas de tratamento adequados à situação comunicativa oral e escrita e ao público-alvo.
4. Produzindo um texto escrito de forma clara e articulada, de acordo com a sua finalidade e público-alvo.
5. Aplicando técnicas de redação de documentos profissionais e usando as regras de ortografia, de pontuação e de acentuação.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Nas atividades profissionais no setor da informática.

RECURSOS

1. Dispositivos tecnológicos com acesso à Internet.
2. Conteúdos multimédia.
3. Ferramentas de tradução, dicionários, entre outros.

OBSERVAÇÕES

Esta UC permite a comunicação em língua inglesa ao nível do utilizador independente (QEER, Escala Global, Nível B: Utilizador Independente; Conselho da Europa, 2001).

UC00627 Instalar e configurar servidores Web

PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Efetuar a instalação e configuração de um servidor Intranet e Internet em ambiente <i>web</i> . 2. Efetuar a manutenção e atualização do servidor intranet e Internet em ambiente <i>web</i> .		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Servidores <i>web</i> – conceitos. 2. Domínio <i>web</i> . 3. Alojamento <i>web</i> . 4. Estrutura de servidor <i>web</i> . 5. Serviços FTP, CHAT, entre outros serviços. 6. Gestão de servidores locais e remotos. 7. Gestão de utilizadores. 8. Normas de segurança da gestão de informação. 9. Segurança dos utilizadores. 10. Regulamento geral de proteção de dados. 11. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Utilizar metodologias de planeamento e desenho da estrutura de um servidor <i>web</i> . 3. Configurar as funcionalidades e serviços FTP, CHAT, entre outros. 4. Realizar a manutenção do servidor e serviços. 5. Configurar ferramentas de gestão remota do servidor. 6. Definir privilégios dos perfis dos utilizadores. 7. Interpretar normas e procedimentos de gestão e segurança da informação. 8. Aplicar procedimentos de segurança na configuração e manutenção de um servidor intranet e Internet em ambiente <i>web</i> . 9. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico. 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Instalar e configurar servidores Web:

1. Seguindo as orientações técnicas na instalação e na seleção das opções para os serviços.
3. Cumprindo os procedimentos de manuseamento e segurança.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da Administração Pública.

RECURSOS

1. Manuais, guíões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Domínio na *web*.
4. Alojamento *web*.
5. Software.

OBSERVAÇÕES

UC00664 Criar bases de dados *noSQL*
PONTOS DE CRÉDITO: 2,25

Realizações		
1. Modelar o armazenamento de dados. 2. Manipular dados. 3. Controlar o acesso aos dados.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Bases de dados não relacionais - conceitos 2. Bases de dados não relacionais: características - flexibilidade; performance; escalabilidade; transações. 3. Tipos de bases de dados não relacionais - chave-valor, documento, gráfico e coluna. 4. Modelo/forma de armazenamento de dados - documentos; coleções; referências. 5. Manipulação de dados. 6. Consultas. 7. Autenticação/privilegios de acesso. 8. Base de dados encriptada (defesa em camadas). 9. Políticas e procedimentos de segurança. 10. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar conceitos e características das bases de dados não relacionais. 3. Interpretar modelos para estruturar os dados. 4. Aplicar as técnicas de manipulação de dados em bases de dados NoSQL. 5. Implementar bases de dados orientadas a documentos. 6. Criar, inserir e importar documentos. 7. Inserir, editar e remover dados. 8. Utilizar consultas para filtrar e listar dados. 9. Criar utilizadores e definir permissões de controlo de acesso. 10. Aplicar normas e regulamentos aplicáveis.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico. 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
*Criar bases de dados *noSQL*:*

1. Modelando a estrutura de dados.
2. Manuseando modelos de dados para aceder e gerir os dados.
3. Aplicando as políticas e procedimentos de segurança estabelecidos.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.

3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos tecnológicos com acesso à Internet.
3. Aplicações de base de dados *low code*.
4. Emulador.

OBSERVAÇÕES

UC 0006/0000 | Desenvolver aplicações com plataformas *low code*
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento da aplicação. 2. Desenhar a estrutura da aplicação. 3. Criar e utilizar os componentes e objetos para a aplicação. 4. Programar as ações dos componentes e objetos da aplicação.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Desenvolvimento de aplicações <i>low code</i> – conceitos; princípios; metodologias. 2. <i>Wireframing</i> . 3. Ambiente de programação <i>low code</i> – funcionalidades; modelos visuais. 4. Componentes das <i>aplicações</i> : <i>interfaces</i> ; design e acessibilidade; multimédia; autenticação e acesso; conteúdos; navegação; ligações; sensores. 5. Elementos visuais de programação <i>drag and drop</i> . 6. Módulos pré-configurados. 7. Plugins pré-configurados. 8. Acessibilidade multiplataforma. 9. Funcionalidades para dispositivos móveis. 10. Normas e regras aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Utilizar metodologias de planeamento de desenvolvimento de aplicações. 3. Utilizar a metodologia de <i>wireframing</i> para criar um protótipo de <i>layouts</i> das <i>interfaces</i> e organizar o tipo de conteúdos e ligações. 4. Manipular o ambiente de programação visual. 5. Adicionar, editar e remover interfaces. 6. Inserir e configurar os componentes e objetos nas interfaces. 7. Utilizar aplicações de edição de imagem e ferramentas de criação multimédia para desenvolver e integrar elementos visuais em interfaces. 8. Utilizar os componentes integrados para programar ações dos componentes e objetos inseridos nas <i>interfaces</i> . 9. Utilizar as funcionalidades da plataforma para realizar as ligações entre <i>interfaces</i> . 10. Integrar módulos pré-configurados. 11. Integrar plugins pré-configurados.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico. 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

	12. Executar a aplicação em várias plataformas e dispositivos. 13. Testar a aplicação. 14. Corrigir erros da aplicação. 15. Aplicar normas e regulamentos aplicáveis.	
--	---	--

CRITÉRIOS DE DESEMPENHO

Desenvolver aplicações com plataformas low code:

1. Cumprindo as especificações metodológicas no planeamento do desenvolvimento da aplicação.
2. Assegurando uma estrutura organizada e alinhada com os objetivos do projeto.
3. Manuseando as aplicações e as funcionalidades disponíveis para criar e usar componentes e objetos na aplicação.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambiente de programação com blocos.
4. Aplicação de edição de imagem.

OBSERVAÇÕES

UC 0007/UC00669

Conceber programas em linguagem C/C++

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
R1. Aplicar metodologias de desenvolvimento de <i>software</i> . R2. Estruturar o programa. R3. Desenvolver o programa em C/C++. R4. Testar e depurar programa.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Ciclo de vida do <i>software</i> . 2. Pensamento computacional - princípios. 3. Algoritmos - princípios. 4. Metodologias de desenvolvimento de <i>software</i> . 5. Ambiente de desenvolvimento. 6. Linguagem estruturada - conceitos; características; estrutura de um programa. 7. Dados - variáveis, declarações e expressões; constantes; tipos de dados simples. 8. Operadores - aritméticos; atribuição; relacionais; lógicos. 9. Estruturas de controlo - sequência; seleção; repetição. 10. Subprogramas - estrutura (funções e procedimentos); variáveis locais e globais; passagem de variáveis por parâmetros. 11. Funcionalidades de um editor de texto. 12. Normas e regras aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Mobilizar os princípios de algoritmia. 4. Reconhecer as características, elementos e sintaxe da linguagem de programação. 5. Utilizar orientações metodológicas para planear as etapas de desenvolvimento do programa. 6. Instalar e configurar o ambiente de programação. 7. Definir os tipos de dados no desenvolvimento do programa. 8. Utilizar operadores no desenvolvimento do programa. 9. Utilizar funções, procedimentos e variáveis no desenvolvimento do programa. 10. Utilizar estruturas de controlo no desenvolvimento do programa. 11. Estruturar subprogramas. 12. Testar e depurar o programa. 13. Corrigir os erros identificados durante os testes ao programa. 14. Criar um guião ou manual de utilizador do programa.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico. 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Conceber programas em linguagem C/C++:

1. Cumprindo as fases metodológicas no desenvolvimento de *software*.

2. Cumprindo regras de programação na definição de dados, no uso de operadores, funções e estruturas de controlo.
3. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações de programação estruturada.
4. Ambientes integrados de desenvolvimento (IDE).
5. Editor de texto.

OBSERVAÇÕES

UC 0008/0000 Conceber programas complexos em linguagem C/C++
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Aplicar metodologias de desenvolvimento de <i>software</i> . 2. Implementar estruturas de dados no desenvolvimento dos programas. 3. Testar e depurar os programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios 2. Algoritmos. 3. Desenvolvimento de <i>software</i> - metodologias. 4. Ambiente de desenvolvimento 5. Estruturas de dados estáticas - <i>strings; arrays; arrays</i> multidimensionais. 6. Estruturas de dados compostas - conceitos; acesso e manipulação. 7. Estruturas de dados dinâmicas - apontadores; acesso e manipulação. 8. Classes e objetos. 9. Herança e Polimorfismo. 10. Exceções. 11. Ficheiros - acesso e manipulação. 12. Funcionalidades de um editor de texto. 13. Normas e regras aplicáveis.	1. Interpretar manuais, guiões e tutoriais de programação. 2. Interpretar os princípios do pensamento computacional. 3. Criar algoritmos. 4. Reconhecer as características, elementos e sintaxe da linguagem de programação. 5. Utilizar orientações metodológicas para planear as etapas de desenvolvimento do programa. 6. Utilizar elementos e sintaxe da linguagem C/C++ para desenvolver aplicações 7. Definir o acesso e manipulação de ficheiros. 8. Corrigir os erros identificados durante os testes ao programa. 9. Elaborar um guião técnico ou manual de utilizador do programa. 10. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido analítico. 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Conceber programas complexos em linguagem C/C++:

1. Cumprindo as fases metodológicas no desenvolvimento de *software*.
2. Cumprindo as regras de programação na utilização de estruturas de dados.
3. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.

4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações de programação estruturada.
4. Ambientes integrados de desenvolvimento (IDE).
5. Compiladores.
6. Editor de texto.

OBSERVAÇÕES

UC 0009/UC00618

 Criar aplicações em linguagem de programação *Python*

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear e estruturar os programas. 2. Criar scripts em <i>python</i> . 3. Testar e depurar programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios. 2. Algoritmos: princípios: 3. Metodologias de desenvolvimento de <i>software</i> . 4. Linguagem de programação <i>Python</i> - características; vantagens; diferenças entre versões; utilizações de <i>Python</i> . 5. Anaconda e ambientes de desenvolvimento integrado - <i>IDE's</i> (<i>Spyder</i> e <i>VS Code</i>). 6. <i>Jupyter Notebook</i> . 7. Tratamento de dados de várias fontes: <i>TEXT</i> O, <i>CSV</i> , <i>SQL</i> , <i>XLS</i> . 8. Programação em <i>Python</i> - variáveis; tipos de dados; tuplos; listas; dicionários; controlo do programa (<i>if</i> , <i>match</i> , <i>for</i> e <i>while</i>). 9. Manipulação de ficheiros e iteradores. 10. <i>Benchmark</i> . 11. <i>Profilers</i> de memória e <i>CPU</i> . 12. <i>Widgets</i> . 13. Geradores. 14. Programação orientada a objetos em <i>Python</i> - classes e objetos; métodos e atributos; construtores; herança; decoradores. 15. Bibliotecas - <i>Pandas</i> , <i>Turtle</i> e <i>NumPy</i> ou <i>Pugame</i> ; análise	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Criar algoritmos. 4. Reconhecer as características, elementos e sintaxe da linguagem de programação. 5. Utilizar orientações metodológicas para planear as etapas de criação de um programa. 6. Instalar o ambiente de desenvolvimento. 7. Utilizar as funcionalidades do ambiente de desenvolvimento para programar em <i>python</i> . 8. Utilizar os elementos e a sintaxe da linguagem <i>python</i> para programar <i>scripts</i> . 9. Utilizar bibliotecas em programas criados em linguagem <i>python</i> . 10. Corrigir os erros identificados durante os testes. 11. Criar um guião técnico, relatório ou manual de utilizador do programa. 12. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor 7. Sentido crítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

gráfica com <i>Mapplotlib</i>; importação de SQL, CSV. 16. Testes unitários. 17. Funcionalidades de um editor de texto. 18. Normas e regulamentos aplicáveis.		
---	--	--

CRITÉRIOS DE DESEMPENHO

Criar aplicações em Linguagem de Programação Python:

1. Cumprindo as fases metodológicas no desenvolvimento de *software*.
2. Cumprindo regras no uso dos elementos e sintaxe da programação.
3. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da Administração Pública.

RECURSOS

1. Manuais, guíões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambiente de desenvolvimento.
4. Editor de texto.

UC 0010/UC00619

Desenvolver aplicações em linguagem JAVA

PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planejar as etapas de desenvolvimento da aplicação. 2. Estruturar programas. 3. Testar e depurar programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios 2. Algoritmos. 3. Desenvolvimento de <i>software</i> - metodologias 4. Ambiente de desenvolvimento. 5. Programação orientada por objetos – paradigma; características (encapsulamento; partilha de comportamento; evolução); classes, objetos, métodos, atributos e mensagens; relações entre classes; relações entre objetos; polimorfismo; classes genéricas. 6. Exceções. 7. Interfaces. 8. Especificidade da linguagem de programação JAVA. 9. Funcionalidades de um editor de texto. 10. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Criar algoritmos. 4. Reconhecer as características, elementos e sintaxe da linguagem de programação. 5. Utilizar orientações metodológicas para planejar as etapas de desenvolvimento de uma aplicação. 6. Utilizar elementos e sintaxe de uma linguagem de programação orientada por objetos para desenvolver aplicações. 7. Desenhar as interfaces. 8. Corrigir os erros identificados durante os testes ao programa. 9. Elaborar um guião técnico ou manual de utilizador do programa. 10. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Sentido critico. 7. Sentido de organização. 8. Rigor. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Desenvolver aplicações em linguagem JAVA:

- Cumprindo as fases metodológicas no desenvolvimento de *software*.
- Aplicando os conceitos, os elementos e a sintaxe para desenvolver os programas.
- Manuseando as aplicações para desenvolver os programas.
- Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

- Empresas do setor da informática.

2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Aplicações de programação orientada por objetos.
4. Máquina virtual.
5. Compiladores.
6. Editor de texto.

OBSERVAÇÕES

UC 0011/UC00620 | Desenvolver aplicações em C#
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planejar as etapas de desenvolvimento do programa. 2. Estruturar os programas. 3. Utilizar a sintaxe da linguagem de programação. 4. Testar e depurar programas.		
CONHECIMENTOS	APTIDÕES	ATITUDES
1. Pensamento computacional: Princípios. 2. Algoritmos: Princípios. 3. Metodologias de desenvolvimento de <i>software</i> . 4. Microsoft .NET <i>Framework</i> . 5. Ambiente de desenvolvimento Microsoft Visual Studio .NET. 6. Variáveis <i>value-type</i> . 7. <i>Statements</i> e exceções. 8. Métodos e parâmetros. 9. <i>Arrays</i> . 10. Técnicas de programação orientada a objetos. 11. Variáveis <i>reference-type</i> . 12. Objetos. 13. Mecanismos de herança. 14. Agregação. 15. <i>Namespaces</i> . 16. Operadores e eventos. 17. Propriedades. 18. Indexadores. 19. Atributos. 20. Funcionalidades de um editor de texto. 21. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guíões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Mobilizar os princípios de algoritmia. 4. Reconhecer as características, elementos e sintaxe das aplicações produzidas no ambiente <i>Visual Studio</i> com recurso à linguagem Visual C#.NET. 5. Utilizar metodologias de desenvolvimento de <i>software</i> . 6. Utilizar elementos e sintaxe da linguagem de programação desenvolver programas. 7. Utilizar o visual C# como interface para base de dados. 8. Criar um guião técnico ou manual de utilizador do programa. 9. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor 7. Sentido critico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Desenvolver aplicações em C#:

1. Cumprindo as fases metodológicas no desenvolvimento de *software*.
2. Manuseando o ambiente de programação para desenvolver programas.
3. Aplicando os conceitos, os elementos e a sintaxe para desenvolver os programas.
4. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Editor de texto.
4. Ambiente de desenvolvimento de programação.
5. *Frameworks*.

OBSERVAÇÕES

UC 0012/UC00621 Criar aplicações em React
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Aplicar metodologias de desenvolvimento de <i>software</i> . 2. Estruturar os programas. 3. Testar e depurar programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios. 2. Algoritmos - princípios. 3. Metodologias de desenvolvimento de <i>software</i> . 4. Ambiente de desenvolvimento. 5. Linguagem de programação React – conceitos; DOM virtual na memória. 6. JSX (JavaScript XML). 7. Processo de <i>Build</i> . 8. Componentes. 9. Dados dinâmicos. 10. Integração com HTML e CSS. 11. Funcionalidades de um editor de texto. 12. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guíões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Criar algoritmos. 4. Reconhecer as características, elementos e sintaxe da linguagem de programação. 5. Utilizar orientações metodológicas para planear as etapas de desenvolvimento do programa. 6. Utilizar ficheiros JSP. 7. Utilizar tipos de dados complexos como adereços. 8. Aplicar técnicas para a criação de componentes. 9. Utilizar objetos e tipos de dados complexos. 10. Corrigir os erros identificados durante os testes ao programa. 11. Elaborar um guião técnico, relatório ou manual de utilizador do programa.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor 7. Sentido critico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Criar aplicações em React:

- Cumprindo as fases metodológicas no desenvolvimento de *software*.
- Cumprindo regras de programação na definição de dados, no uso de operadores, funções e estruturas de controlo.
- Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

- Empresas do setor da informática.

1. Empresas de consultoria de Informática/Tecnologias de Informação.
2. Lojas de informática.
3. Serviços de apoio técnico.
4. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambientes de desenvolvimento de programação.
4. Editor de texto.

OBSERVAÇÕES

UC 0013/0000 Criar aplicações em COBOL
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Aplicar metodologias de desenvolvimento de <i>software</i> . 2. Estruturar os programas. 3. Testar e depurar programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios. 2. Algoritmos - princípios. 3. Metodologias de desenvolvimento de <i>software</i> . 4. Ambiente de desenvolvimento. 5. Linguagem COBOL - linguagem COBOL no contexto das linguagens de programação; Vantagens e desvantagens do uso da linguagem COBOL no desenvolvimento de aplicações. 6. Estrutura da linguagem - notações e sintaxe da linguagem. 7. Estrutura de um programa COBOL - divisões, secções e parágrafos. 8. Divisão data division - tipos de dados; declaração dos dados; níveis; classes e categorias; cláusula picture; dados com valores constantes; decomposição dos dados por campos; estruturas de dados em cobol. 9. Ficheiros num programa COBOL: tipos de declaração dos ficheiros na environment e data division; registos dos ficheiros e as suas declarações; modelos de organização e acesso dos ficheiros em COBOL. 10. Divisão procedure division - estrutura da divisão; notações e a sintaxe das notações.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Elaborar algoritmos. 4. Reconhecer as características da linguagem de programação. 5. Utilizar orientações metodológicas para planear as etapas de criação do programa. 6. Utilizar regras para estruturar um programa em Cobol. 7. Utilizar elementos e sintaxe da linguagem COBOL. 8. Utilizar técnicas para descrever ficheiros num programa COBOL. 9. Corrigir os erros identificados durante os testes ao programa. 10. Elaborar um guião técnico ou manual de utilizador do programa. 11. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor 7. Sentido critico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

11. Estruturas de controlo - instrução perform; instrução if; instrução evaluate; análise das instruções em COBOL e as estruturas lógicas algorítmicas (comparação). 12. Instruções sobre os dados em memória - instrução move. 13. Expressões aritméticas - operadores aritméticos; regras de prioridades dos operadores. 14. Expressões condicionais - relações condicionais; operadores condicionais; relações compostas; operadores lógicos. 15. Expressões variadas - instrução compute; instrução add; instrução subtract; instrução multiply; instrução divide. 16. Funcionalidades de um editor de texto. 17. Normas e regulamentos aplicáveis.		
--	--	--

CRITÉRIOS DE DESEMPENHO

Criar aplicações em COBOL:

1. Cumprindo as fases metodológicas no desenvolvimento de *software*.
2. Cumprindo regras no uso dos elementos e sintaxe da programação.
3. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambiente de programação
4. Editor de texto.

OBSERVAÇÕES

UC 0014/UC00622 Criar aplicações em HTML5
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento de uma aplicação <i>web</i> . 2. Estruturar a aplicação. 3. Testar e depurar as aplicações.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional – princípios. 2. Algoritmos – princípios. 3. Metodologias de desenvolvimento de <i>software</i> . 4. Ambiente de desenvolvimento. 5. Elementos multimédia na página <i>web</i> . 6. Semântica do HTML5 7. Integração CSS3. 8. Funcionalidades de um editor de texto. 9. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Criar algoritmos. 4. Reconhecer as características, elementos e sintaxe da programação em HTML5. 5. Utilizar orientações metodológicas para planear as etapas de criação do programa. 6. Configurar o ambiente de programação. 7. Integrar elementos multimédia. 8. Utilizar elementos e sintaxe do HTML5. 9. Corrigir os erros identificados durante os testes. 10. Elaborar um guião técnico, relatório ou manual de utilizador do programa. 11. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética 5. Iniciativa. 6. Rigor. 7. Sentido crítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Criar aplicações em HTML5:

- Cumprindo as fases metodológicas no desenvolvimento de *software*.
- Manuseando o ambiente de programação para desenvolver programas.
- Cumprindo regras no uso dos elementos e sintaxe da programação.
- Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

- Empresas do setor da informática.
- Empresas de consultoria de Informática/Tecnologias de Informação.
- Lojas de informática.

4. Serviços de apoio técnico.
5. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambiente de desenvolvimento.
4. Editor de texto.

OBSERVAÇÕES

UC 0015/0000 Programar videojogos
PONTOS DE CRÉDITO: 4,50

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento do videojogo. 2. Instalar e configurar o ambiente de desenvolvimento do videojogo. 3. Executar o videojogo. 4. Testar e depurar a programação do videojogo.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Desenvolvimento de videojogos - evolução; conceitos. 2. Pensamento computacional - princípios. 3. Algoritmos - princípios. 4. Metodologias de desenvolvimento de videojogos. 5. Ambiente de desenvolvimento. 6. Elementos estruturais. 7. Design de jogos - mecânica; evolução; níveis; enredo; papéis. 8. Linguagem de programação - interface da ferramenta; objetos; scripts; variáveis e funções; personagens e objetos. 9. Funcionalidades do editor de texto.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Elaborar algoritmos. 4. Reconhecer as características, elementos e sintaxe da programação. 5. Utilizar orientações metodológicas para planear as etapas de criação do videojogo. 6. Instalar e configurar o ambiente de programação. 7. Aplicar técnicas de <i>design</i> de jogos para definição dos cenários e <i>sprites</i> . 8. Explorar a interface da ferramenta. 9. Criar e transformar objetos. 10. Criar scripts. 11. Utilizar variáveis e funções. 12. Animar as personagens e objetos. 13. Corrigir os erros identificados durante os testes ao videojogo. 14. a Elaborar um guião técnico, relatório ou manual de utilizador do videojogo.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética 5. Iniciativa. 6. Rigor. 7. Sentido crítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO
Programar videojogos:

1. Cumprindo as várias etapas de planeamento do jogo.
2. Cumprindo regras no uso dos elementos e sintaxe da programação.

3. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática e videojogos.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Dispositivos móveis.
4. Motores de jogos.
5. Ambiente de programação
6. Editor de texto.
7. Editor de imagem.
8. Editor de som.

OBSERVAÇÕES

UC 0016/0000 Programar artefactos tangíveis

PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Efetuar a montagem ou construção dos artefactos. 2. Programar as ações dos artefactos. 3. Testar e depurar a programação.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Pensamento computacional - princípios. 2. Algoritmos - princípios. 3. Robótica - conceitos. 4. Eletricidade e eletrônica - princípios. 5. Características e modo de funcionamentos dos artefactos - <i>robots</i>; placas eletrônicas; drones; sensores; atuadores; motores. 6. Ambiente de desenvolvimento. 7. Programação tangível - conceitos. 8. Linguagem de programação - sequências; eventos; ciclos; condições; funções; variáveis. 9. Normas de segurança e saúde no trabalho. 10. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Elaborar algoritmos. 4. Analisar o funcionamento dos mecanismos dos artefactos. 5. Montar ou construir artefactos. 6. Instalar o ambiente de programação. 7. Utilizar as funcionalidades do ambiente de programação para ligar e interagir com os artefactos. 8. Utilizar os elementos e a sintaxe da linguagem para programar as ações dos artefactos. 9. Corrigir os erros identificados durante os testes. 10. Aplicar procedimentos de segurança no manuseamento e na montagem dos artefactos. 11. Aplicar as normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética 5. Iniciativa. 6. Rigor. 7. Sentido crítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Programar artefactos tangíveis:

1. Cumprindo as orientações de montagem ou construção de artefactos.
2. Cumprindo regras no uso dos elementos e sintaxe da programação.
3. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.

4. Serviços de apoio técnico.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.
3. Ambientes de programação com blocos para artefactos tangíveis.
4. Artefactos (*Robots*, Placas eletrónicas, Drones, Sensores, Atuadores, Motores, Componentes diversos)
5. Ferramentas.

OBSERVAÇÕES

UC 0017/UC00623 Programar com sistemas de Inteligência Artificial
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Criar <i>chatbots</i> . 2. Programar modelos de visão computacional. 3. Testar e depurar programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Princípios do pensamento computacional. 2. Inteligência artificial – evolução; conceitos (<i>machine learning</i> , redes neurais, algoritmos genéticos, processamento de linguagem natural). 3. Ambiente de desenvolvimento. 4. Programação - conceitos e elementos. 5. Linguagem de programação – sequências; eventos; ciclos; condições; funções; variáveis. 6. <i>Machine learning</i> - supervisionado e não supervisionado; algoritmos 7. Processamento de Linguagem Natural - pré-processamento de texto (limpar e preparar dados); modelos. 8. Bibliotecas. 9. Modelos de visão computacional (reconhecimento de textos, objetos, imagens e sons). 10. Regulamento geral de proteção dos dados. 11. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Interpretar os princípios do pensamento computacional. 3. Criar algoritmos. 4. Interpretar os conceitos relacionados com inteligência artificial. 5. Analisar o funcionamento dos sistemas com inteligência artificial. 6. Instalar o ambiente de programação. 7. Reconhecer as características, elementos e sintaxe da linguagem de programação. 8. Utilizar elementos e a sintaxe da linguagem para desenvolver o programa. 9. Adicionar extensões e bibliotecas ao ambiente de programação. 10. Analisar dados e identificar padrões com algoritmos de <i>machine learning</i> . 11. Utilizar os sistemas de IA para treinar modelos com números, imagens, objetos e sons. 12. Programar o reconhecimento de textos, faces, objetos e sons. 13. Corrigir os erros identificados durante os testes. 14. Aplicar as normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor. 7. Sentido crítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Programar com sistemas de Inteligência Artificial:

1. Cumprindo regras no uso dos elementos e sintaxe da programação.
2. Utilizando bibliotecas para desenvolver *chatbots*.
3. Executando programas para o reconhecimento de objetos, textos, imagens e sons.
4. Executando a programação e corrigindo erros.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos tecnológicos com acesso à Internet.
3. Ambiente de desenvolvimento.
4. Bibliotecas.

OBSERVAÇÕES

UC 0018/UC00643
Programar com sistemas baseados na Internet das Coisas (IoT)
PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Planear as etapas de desenvolvimento do programa para dispositivos IoT. 2. Desenvolver código para aquisição e processamento de dados. 3. Estabelecer a comunicação entre os programas e os dispositivos. 4. Testar e depurar os programas.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Internet das coisas –evolução; conceitos. 2. Características e modo de funcionamentos dos dispositivos – microcontroladores; sensores; atuadores. 3. Pensamento computacional - princípios. 4. Algoritmos - princípios: 5. Plataformas e ambientes de desenvolvimento. 6. Programação – controle; lógica; matemática; texto; listas; tempo; cores; procedimentos; variáveis. 7. Protocolos de comunicação - MQTT, CoAP, HTTP/HTTPS; comunicação dos dispositivos entre si e com a nuvem. 8. Normas de segurança e saúde no trabalho. 9. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Reconhecer a aplicação da Internet das Coisas no dia a dia 3. Interpretar os princípios do pensamento computacional. 4. Mobilizar os princípios de algoritmia. 5. Reconhecer as características da linguagem de programação. 6. Analisar o funcionamento e a forma de comunicação dos mecanismos dos dispositivos. 7. Adotar as orientações técnicas para montar ou construir dispositivos. 8. Configurar o ambiente de desenvolvimento. 9. Utilizar as funcionalidades da aplicação de programação para ligar e interagir com os dispositivos. 10. Utilizar elementos e sintaxe da aplicação de programação para programar ações dos dispositivos. 11. Corrigir os erros identificados durante os testes ao programa. 12. Recolher dados.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Rigor. 7. Sentido crítico. 8. Sentido de organização. 9. Respeito pelas regras e normas definidas.

	13. Aplicar procedimentos de segurança no manuseamento e na montagem dos dispositivos. 14. Aplicar procedimentos de segurança e ética na programação e recolha de dados.	
--	--	--

CRITÉRIOS DE DESEMPENHO

Programar com sistemas baseados na Internet das Coisas (IoT):

1. Cumprindo as várias etapas de planeamento do projeto.
2. Cumprindo regras no uso dos elementos e sintaxe da programação.
3. Executando e testando a ligação entre a programação e os dispositivos.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.
2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da administração pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos tecnológicos com acesso à internet.
3. Plataformas e ambientes de desenvolvimento.
4. Artefactos tangíveis.

OBSERVAÇÕES

UC0019/UC00617 Utilizar serviços Git e GitHub

PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Instalar o <i>Git</i> . 2. Criar e configurar uma conta na plataforma <i>GitHub</i> . 3. Criar e gerir repositórios.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Git - definição e conceitos; repositório e <i>commit</i> ; comandos para o terminal; repositórios locais e remotos. 2. Cliente gráfico: repositórios locais e remotos. 3. GitHub -conta e página pessoal; repositórios; ficheiros README.md. 4. <i>Tags</i> . 5. <i>Branches</i> – conceitos; operações básicas. 6. Resolução de conflitos - localmente e na plataforma GitHub. 7. Normas e regulamentos aplicáveis.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Analisar os sistemas de controlo de versões. 3. Utilizar orientações técnicas para a instalação e configuração do Git e de um cliente gráfico. 4. Criar uma conta e página pessoal. 5. Criar, gerir e clonar repositórios. 6. Utilizar um cliente gráfico como alternativa à linha de comando. 7. Analisar e editar o histórico de <i>commits</i> para gerir repositórios. 8. Aplicar e gerir <i>tags</i> . 9. Utilizar <i>branches</i> e resolver conflitos nos repositórios. 10. Colaborar e contribuir em repositórios. 11. Aplicar normas e regulamentos.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Ética. 5. Iniciativa. 6. Sentido crítico. 7. Sentido de organização. 8. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Utilizar serviços *Git* e *GitHub*:

1. Seguindo as orientações técnicas na instalação da plataforma.
2. Manuseando a plataforma para criar e gerir os repositórios.
3. Cumprindo as normas e regulamentos aplicáveis.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Empresas do setor da informática.

2. Empresas de consultoria de Informática/Tecnologias de Informação.
3. Lojas de informática.
4. Serviços de apoio técnico.
5. Organismos da Administração Pública.

RECURSOS

1. Manuais, guiões e tutoriais técnicos.
2. Dispositivos eletrónicos com acesso à Internet.

OBSERVAÇÕES

UC 0020/0000 Editar imagens *bitmap*

PONTOS DE CRÉDITO: 2,25

REALIZAÇÕES		
1. Executar a edição de imagens. 2. Manipular e restaurar imagens. 3. Efetuar a composição de imagens.		
CONHECIMENTOS	APTIDÕES	ATTITUDES
1. Edição de imagem: conceitos e tipos. 2. Ambiente de trabalho da aplicação de edição de imagem. 3. Ferramentas e respetivas funcionalidades – de seleção, de corte, de retoque, de pintura, de texto e de desenho. 4. Edição de imagens – aplicações informáticas e ferramentas: de seleção, de manipulação de imagem, de pintura e texturas, camadas (<i>layers</i>), processo de edição: organização, edição e manipulação de texto. 5. Ficheiros – formatos, tipologias e finalidades: PSD, JPEG, PNG, TIFF e PDF. 6. Gestão de ficheiros digitais e arquivo - arquitetura de pastas, ficheiros: formatos e nomenclatura. 7. Normas de segurança e saúde no trabalho. 8. Normas de proteção ambiental 9. Direitos de autor.	1. Interpretar manuais, guiões e tutoriais técnicos. 2. Explorar ambiente de trabalho da aplicação de edição de imagem. 3. Utilizar ferramentas para selecionar, recortar e redimensionar imagens. 4. Utilizar funcionalidades e técnicas para manipular e restaurar imagens. 5. Elaborar composição com imagens e texto. 6. Selecionar os sistemas de cor. 7. Efetuar alterações de cor nas imagens 8. Aplicar filtros e efeitos para transformar imagens. 9. Utilizar camadas para fazer composição de imagem. 10. Criar elementos gráficos em formato de imagem para integrar no desenvolvimento de páginas <i>web</i> e aplicações. 11. Gravar imagens em diferentes formatos. 12. Atribuir a autoria na utilização de imagens.	1. Responsabilidade pelas suas ações. 2. Autonomia no âmbito das suas funções. 3. Empenho e persistência na resolução de problemas. 4. Iniciativa. 5. Sentido crítico. 6. Sentido de organização. 7. Respeito pelas regras e normas definidas.

CRITÉRIOS DE DESEMPENHO

Editar imagens bitmap:

1. Selecionando as ferramentas da aplicação informática para edição e composição de imagens.
2. Aplicando métodos e técnicas de manipulação e restauro.
3. Salvando e exportando o trabalho no formato de ficheiro de acordo com a finalidade pretendida.

CONTEXTO (EXEMPLOS DE USO DA COMPETÊNCIA)

1. Aplicável a diversos contextos.

RECURSOS

1. Manuais, guíões e tutoriais técnicos.
2. Dispositivos tecnológicos com acesso à Internet.
3. Dispositivos de captura de imagem.
4. Aplicações de edição de imagem.

OBSERVAÇÕES