ESCOLA DO FUTURO

PLANO DE ENSINO

1. IDENTIFICAÇÃO DO CURSO

NOME DO CURSO		QTP3 Segurança e Hospedagem *Como terceiro componente da Habilitação Profissional Técnica em Desenvolvimento Web e Cibersegurança*	
MODALIDADE		Concomitante Intercomplementar ao Ensino Médio Integral	
CATEGORIA		Presencial	
EIXO TECNOLÓGICO		Informação e Comunicação	
COMPONENTE CURRICULAR		Segurança e Hospedagem	
DATA INÍCIO	19/01/2026	**DATA FIM**	18/12/2026
CARGA HORÁRIA		120h (120 aulas de 60 minutos)	

4. PLANO DE ENSINO

EMENTA:
Fundamentos da Arquitetura Cliente-Servidor e Protocolos da Internet (TCP/IP, HTTP/1.1, HTTP/2, QUIC). Infraestrutura de Nomes de Domínio (DNS): Zonas, Registros (A, CNAME, MX, TXT) e configuração de servidores BIND9. Administração de Sistemas Operacionais para Servidores Web (Linux Server e Windows Server IIS). Instalação, configuração e otimização de Servidores Web (Apache HTTP Server, Nginx). Hospedagem de Aplicações Dinâmicas: Runtimes (PHP-FPM, Python/WSGI, Node.js) e Servidores de Banco de Dados (MySQL/MariaDB, PostgreSQL). Virtualização e Containerização aplicadas à hospedagem (Docker, Docker Compose). Segurança de Servidores (*Hardening*): Gestão de usuários, permissões de sistema de arquivos (ACLs), configuração de SSH seguro e gerenciamento de atualizações. Segurança de Perímetro e Aplicação: Firewalls de Host (UFW, Iptables), Web Application Firewalls (ModSecurity), prevenção contra DDoS e mitigação de ataques de força bruta (Fail2Ban). Criptografia e PKI: Implementação de SSL/TLS, gestão de certificados (Let's Encrypt/Certbot) e segurança de transporte (HSTS). Alta Disponibilidade e Performance: Balanceamento de Carga, Proxy Reverso, Caching (Varnish, Redis) e Redes de Distribuição de Conteúdo (CDN). Monitoramento de Infraestrutura (Zabbix/Prometheus), gestão de Logs e estratégias de Backup e Recuperação de Desastres (Disaster Recovery). DevOps para Hospedagem: Pipelines de Deploy (CI/CD) e Git. Aspectos legais e éticos na hospedagem de dados (LGPD).

UFG UNIVERSIDADE FEDERAL DE GOIÁS

ESCOLA DO FUTURO

OBJETIVOS:

- **Compreender** o ciclo de vida de uma requisição web, desde a resolução de nomes (DNS) até a entrega do conteúdo pelo servidor, diagnosticando gargalos e falhas;
- **Implementar e Gerenciar** servidores web robustos (Nginx/Apache), configurando Virtual Hosts, reescrita de URL e otimizações de performance para múltiplas aplicações simultâneas;
- **Projetar** infraestruturas de hospedagem seguras, aplicando técnicas de *Hardening* no Sistema Operacional para minimizar a superfície de ataque e prevenir invasões;
- **Configurar** a camada de segurança de transporte (HTTPS), gerenciando o ciclo de vida de certificados digitais SSL/TLS e garantindo a confidencialidade dos dados dos usuários;
- **Mitigar** ataques comuns contra servidores web (SQL Injection, XSS, DDoS, Brute Force) através da implementação de WAFs (Web Application Firewalls) e ferramentas de detecção de intrusão;
- **Automatizar** processos de *deploy* e manutenção de servidores utilizando contêineres (Docker) e conceitos de Infraestrutura como Código, facilitando a escalabilidade;
- **Monitorar** a saúde e o desempenho dos serviços de hospedagem, configurando alertas proativos para uso de CPU, memória, disco e tempo de resposta;
- **Desenvolver** planos de contingência, backup e recuperação de desastres para garantir a continuidade do negócio em casos de falha crítica ou perda de dados.

UFG UNIVERSIDADE FEDERAL DE GOIÁS

ESCOLA DO FUTURO

COMPETÊNCIA:

1. Administração de Infraestrutura Web:
- Provisionar servidores Linux/Windows, instalar pilhas de serviços (LAMP/LEMP) e gerenciar domínios e zonas de DNS com autonomia.

2. Segurança de Servidores e Aplicações:
- Aplicar *patches* de segurança, configurar firewalls locais e implementar regras de WAF para proteger aplicações hospedadas contra o OWASP Top 10.

3. Otimização e Alta Disponibilidade:
- Configurar estratégias de cache, balanceamento de carga e proxy reverso para suportar alto tráfego e garantir a disponibilidade do serviço.

4. Operações (DevOps) e Monitoramento:
- Utilizar ferramentas de monitoramento para análise de logs e métricas, e implementar rotinas automatizadas de backup e deploy de aplicações.

METODOLOGIA DA APRENDIZAGEM:
- **Aulas Expositivas Dialogadas:** Introdução aos conceitos de segurança com análise de casos reais de vazamento de dados e falhas de infraestrutura, estimulando o debate sobre o impacto ético e financeiro dessas falhas.
- **Laboratórios Práticos:** Uso intenso do laboratório para codificação. Os alunos construirão APIs vulneráveis propositalmente para, em seguida, aplicar as correções de segurança (ataque e defesa).
- **Simulações de Deploy:** Vivência do "chão de fábrica" de uma empresa de TI, onde os alunos deverão configurar servidores, lidar com erros de versão, variáveis de ambiente e bancos de dados em nuvem.
- **Resolução de Problemas (PBL):** Desafios semanais onde os estudantes precisam "salvar" uma aplicação que caiu ou foi invadida, promovendo o raciocínio rápido e trabalho em equipe.
- **Feedback Contínuo:** Acompanhamento individual durante as práticas para correção de rota imediata, focando na qualidade do código e nas boas práticas de segurança.

UFG UNIVERSIDADE FEDERAL DE GOIÁS

ESCOLA DO FUTURO

CONTEÚDOS PROGRAMÁTICOS:

1. **Fundamentos de Middleware e Validação**
   - Lógica e construção de Middlewares personalizados no Node.js/Express;
   - Validação e sanitização de dados de entrada (Input Validation);
   - Tratamento centralizado de erros e prevenção de vazamento de stack trace.
2. **Autenticação e Controle de Acesso**
   - Criptografia e Hashing de senhas (Bcrypt/Argon2);
   - Implementação de JWT (JSON Web Tokens) para sessões stateless;
   - Controle de acesso baseado em funções (RBAC - Role Based Access Control).
3. **Proteção contra Vulnerabilidades Web (OWASP)**
   - Prevenção a Injeção de SQL e NoSQL;
   - Proteção contra Cross-Site Scripting (XSS) e Cross-Site Request Forgery (CSRF);
   - Configuração de Headers de Segurança (Helmet.js, CORS);
   - Rate Limiting e proteção contra força bruta.
4. **Hospedagem e Infraestrutura em Nuvem**
   - Diferenças entre IaaS, PaaS e SaaS;
   - Configuração de variáveis de ambiente (.env) para produção;
   - Provedores de Nuvem (AWS, Azure, Google Cloud, Render, Vercel);
   - Configuração de Banco de Dados na Nuvem (Atlas, RDS).
5. **Deploy e Operação (DevSecOps)**
   - Conceitos de CI/CD (Integração e Entrega Contínuas);
   - Deploy automatizado via Git;
   - Logs de auditoria e monitoramento de saúde da aplicação;
   - Estratégias de Backup e Recuperação de Desastres.

UFG UNIVERSIDADE FEDERAL DE GOIÁS

ESCOLA DO FUTURO

RECURSOS DIDÁTICOS:

- Laboratório de informática com computadores (configuração para virtualização se possível);
- Acesso à internet liberado para repositórios (GitHub) e documentações;
- Softwares: VSCode, Node.js, Git, Postman/Insomnia, MongoDB Compass/MySQL Workbench;
- Contas de estudante/gratuitas em provedores de nuvem (AWS Educate, Render, Vercel);
- Projetor multimídia para *live coding*;
- Quadro branco para desenho de arquitetura de servidores e fluxos de dados.

CRITÉRIOS DE AVALIAÇÃO:

A avaliação será contínua e processual, focada na evolução técnica e comportamental do estudante:

1. **Avaliação Diagnóstic**a (Início dos bimestres)

    - **Qualitativa:** Rodas de conversa para medir o conhecimento prévio sobre como a internet funciona, o que são servidores e experiências passadas com "vírus" ou invasões.
    - **Quantitativa:** Quizzes rápidos (Kahoot ou Forms) para nivelamento de conceitos de redes e HTTP.

UFG UNIVERSIDADE FEDERAL DE GOIÁS

2. **Avaliação Formativa** (Durante o processo)

   - **Qualitativa:** Observação da autonomia durante os laboratórios. Capacidade de pesquisar erros (debug) e colaborar com colegas na resolução de problemas de configuração.
   - **Quantitativa:** Entrega de pequenos desafios semanais (ex: "Crie uma rota que só aceita emails válidos", "Faça o deploy dessa API Hello World").

3. **Avaliação Somativa** (Fechamento de ciclo)

4. **Qualitativa:** Defesa oral das escolhas arquiteturais no projeto final (Por que usou esse banco? Por que essa cloud?).

5. **Quantitativa:** Projeto Prático Final – Uma API segura, hospedada na nuvem, com documentação, que passe por um checklist de segurança (sem erros de console, autenticação funcionando, dados criptografados).

Janeiro

1º Bimestre: Fundamentos de Segurança e Middlewares

Aulas 1-6: Introdução à Segurança da Informação (Confidencialidade, Integridade, Disponibilidade). O papel do Back-End na segurança.
Aulas 7-14: Construção de Middlewares no Express. Interceptação de requisições. Logs de acesso.
Aulas 15-22: Validação de Dados. Bibliotecas de validação (Joi/Zod). Sanitização de inputs.
Aulas 23-30: Tratamento de erros seguro. Diferença de erros de produção vs. desenvolvimento. Avaliação prática do bimestre.

2º Bimestre: Autenticação e Autorização

Aulas 31-38: Conceitos de Criptografia. Hashing de senhas na prática. Salting.

Aulas 39-46: Implementação de Login. Geração e validação de tokens JWT.

Aulas 47-54: Controle de acesso. Proteção de rotas (Públicas vs. Privadas).

Aulas 55-60: Gerenciamento de sessão. Logout e expiração de tokens. Avaliação prática (Sistema de Login).

3º Bimestre: Defesa Contra Vulnerabilidades (Hardening)

Aulas 61-68: Estudo do OWASP Top 10. Laboratório de Injeção de SQL/NoSQL (Ataque e Correção).

Aulas 69-76: Proteção contra XSS e CSRF. Configuração de Headers HTTP seguros (Helmet).

Aulas 77-84: Proteção contra força bruta (Rate Limiting). Segurança em APIs REST.

Aulas 85-90: Auditoria de código. Ferramentas de análise estática. Avaliação prática (Pentest básico na própria aplicação).

4º Bimestre: Hospedagem, Nuvem e Deploy

Aulas 91-98: Introdução à Computação em Nuvem. Preparação do ambiente (Variáveis .env, Git Ignore).

Aulas 99-106: Deploy de Banco de Dados na Nuvem (DBaaS). Conexão segura.

Aulas 107-114: Deploy da Aplicação (PaaS). Processo de Build e Start. Logs de produção.

Aulas 115-120: Monitoramento básico (Uptime). Estratégias de Backup. Projeto Final Integrador (Aplicação Segura e Hospedada).

REFERÊNCIAS:

Bibliografia básica:

TORRES, Dorian. Construindo aplicações do zero: com NodeJS, Express e React. [Recurso eletrônico], 2023.
LIMA, Adriano. Segurança na computação em nuvem. Editora SENAC São Paulo, 2018. [Recurso eletrônico].
MEDEIROS, Luciano Frontino de. Banco de dados: princípios e prática. 1. ed. Curitiba: Intersaberes, 2013.

Bibliografia complementar:

SILVA, Michel Bernardo Fernandes da. Cibersegurança: uma visão panorâmica sobre a segurança da informação na internet. 1. ed. [S.l.]: Freitas Bastos, 2023.
SOUSA NETO, Manoel Veras de. Computação em nuvem. 1. ed. Rio de Janeiro: Brasport, 2015.
BEAVER, Kevin. Hacking para leigos. 3ª Ed. Alta Books, 2014. [Recurso eletrônico].