

# Nome do Curso: Engenharia de Qualidade e Teste de Software

- Justificativa: O software faz atualmente parte da maioria das atividades cotidianas. Os bugs de software têm, portanto, um impacto econômico e social para organizações e para a sociedade em geral. Temos cases recentes com variados exemplos de consequências impactantes decorrentes de erros de software. Uma estratégia inadequada de qualidade de software aumenta o risco de erros e, por conseguinte, grandes consequências para os clientes e responsáveis pelo software. Percebemos então que a qualidade de software é um fator crítico para a sobrevivência e o sucesso de qualquer empresa que dele faz uso. E com o crescimento da indústria de software, várias normas e modelos de qualidade têm sido propostos ao longo dos últimos anos, as quais têm sido fortemente adotadas por organizações de forma global. O esforço é para que o processo de Garantia de Qualidade de Software (SQA) possa reduzir o retrabalho, obtendo maior produtividade e diminuindo o tempo de desenvolvimento. A Engenharia de Qualidade e Teste de Software é então uma área essencial para garantir a qualidade e confiabilidade das aplicações. Através de metodologias e práticas, essa matéria é responsável por identificar, prevenir e corrigir erros e falhas em programas de computador. O objetivo da gestão da Qualidade de Software é dar suporte a todos os requisitos; que seja amigável, seguro, útil, utilizável, estável, que satisfaça as necessidades e exigências dos usuários sem apresentar erros. Os profissionais responsáveis pela SQA medem, testam e melhoram o processo de desenvolvimento de software. Eles trabalham integrados com a equipe de desenvolvimento para implementar todos os aspectos do teste de software. No entanto, apesar de ser uma área estratégica para as empresas e com excelentes e promissoras oportunidades de trabalho, o mercado de trabalho apresenta uma grande carência de profissionais especializados e capacitados para atender às diversas expectativas. Isso pode ser observado na rede social LinkedIn. Ela divulgou recentemente um relatório apontando que o profissional Engenheiro de Qualidade de Software figura entre os cargos com maior demanda no Brasil nos últimos cinco anos. Esse cenário – investimento das organizações, carência de profissionais no mercado e oportunidades na carreira, reforça a necessidade de formar profissionais com habilidades e altamente qualificados e atualizados nas práticas e técnicas nessa área. Essa formação permitiria que eles pudessem atuar com maior eficiência na identificação e correção de falhas, além de serem capazes de implementar processos de qualidade e de testes em todo o ciclo de produção de software. O curso de Engenharia de Qualidade e Teste de Software insere-se nesse contexto. Ele tem como objetivo capacitar especialista com habilidades e visão da gestão estratégica sobre qualidade de software. Ele irá promover a compreensão dos alunos sobre os paradigmas de teste e em métodos, ferramentas e práticas relacionadas à qualidade. Isso ajudará as organizações a obter maior resiliência, disponibilidade e confiabilidade em suas aplicações corporativas e a reduzir o tempo e o trabalho no ciclo de desenvolvimento de software. Mostra também a evolução do processo de SQA e teste de software e suas tendências. Portanto, alinhado à missão da PUC Minas, a proposição deste curso tem o intuito de agregar valor aos profissionais de TI oferecendo conhecimentos especializados, elementos teóricos e as ferramentas

necessárias para atender a grande demanda do mercado de trabalho e ajudar a aumentar o valor agregado para as soluções de software. Matriz curricular atualizada para atender as necessidades do mercado, oferecendo uma formação focada na definição de arquiteturas de software em tecnologias atuais; Conteúdos apresentados por meio de casos reais, que colocam o aluno próximo de situações comuns no dia a dia de um usuário de dispositivo móvel; Tradição de ensino PUC Minas; Professores com muita experiência de mercado e com uma sólida formação acadêmica; Professores Mentores experientes focados em orientar e motivar para otimizar o aprendizado; Abordagens inovadoras de ensino-aprendizagem em que as aulas e atividades pedagógicas são centradas nas necessidades dos alunos. Elas seguem dinâmicas orientadas por princípios de metodologias ativas; Experiência de aprendizado é suportada por ferramentas interativas - acessível via Web ou dispositivos móveis - incluindo salas virtuais, bate-papos e fóruns de discussão para estimular o aluno a um maior engajamento com o seu curso; Aprendizagem flexível em que o aluno planeja o próprio ritmo para alcançar seus objetivos pessoais;

- Objetivos: O curso tem como objetivo principal proporcionar uma formação avançada e especializada na área de engenharia de qualidade e testes de software, visando a capacitação técnica, teórica e prática necessária para atuar em projetos de software garantindo sua qualidade. De forma estruturada, o aluno será capacitado a: Atuar com Engenheiros Qualidade e Teste de Software para garantir que o desenvolvimento siga os processos estabelecidos e seja orientado por uma abordagem centrada em objetivos de nível de serviço bem definidos; Compreender os conceitos básicos de garantia e gerenciamento de qualidade e entender a importância da cultura de qualidade e testes de software no contexto corporativo e no processo de desenvolvimento de software; Utilizar os principais modelos para analisar e avaliar a qualidade do processo de desenvolvimento de software; Descrever os princípios e as principais técnicas de verificação, validação e testes de software; Conhecer elementos básicos sobre tecnologias de apoio aos processos de controle de CD/CI, gerência de configuração, automação de testes, containerização e monitoramento e explicitar componentes básicos para uma solução que usa tais tecnologias; Aplicar técnicas e práticas modernas para a automação de testes juntamente com o gerenciamento de processos de desenvolvimento de software garantindo soluções robustas e de qualidade; Propor soluções e estar envolvido nas decisões para maximizar o valor do software fazendo, também, com que ele gere valor para os clientes. Definir e analisar métricas para mensurar a qualidade de software e para fazer estimativas do seu projeto; Comunicar informações de qualidade a várias partes interessadas.
- Público Alvo: Profissionais com formação superior: Em Ciência da Computação, Engenharia de Computação, Engenharia de Software, Sistemas de Informação, tecnólogos da área de Tecnologia da Informação ou áreas afins; Nas mais diversas áreas e que necessitam de competências emergentes para o Gestão da Qualidade e Teste de Software; Com experiência em Qualidade e Teste de Software e que queiram ampliar e aperfeiçoar seus conhecimentos;

## Disciplinas:

Disciplina 1: IA GENERATIVA PARA ENGENHARIA DE SOFTWARE

Ementa: Princípios de produtividade e agilidade. Fundamentos de IAs Generativas (GenAI). Plataformas de GenAI. Engenharia de Prompt. Desafios e planejamento para adoção de IA no processo de desenvolvimento. Conceitos de AI-as-a-Service, AI-as-a-commodity, AI Gateways. Desenvolvimento de soluções com GenAI, Definição de métricas e análise do ROI. Tendências.

## **Disciplina 2: QUALIDADE DE SOFTWARE**

Ementa: Fundamentos da qualidade de software. Qualidade do processo e qualidade do produto. Mindset de produto. Métricas de processo e do produto. Métricas e estimativas aplicadas a Teste de Software. Normas e modelos de maturidade de processo de desenvolvimento de software. Garantia da Qualidade de Software. Avaliação de qualidade e certificação. Valor do processo de qualidade de software para os objetivos estratégicos do negócio. Auditoria. Tendências. Service Level Objectives (SLO's). Monitoração contínua e Observabilidade. Estudo de casos.

## **Disciplina 3: CULTURA E PRÁTICAS DEVOPS**

Ementa: A cultura DevOps. Integração contínua e entrega contínua. Estratégias de deploy. Projeto de pipeline para build e deployment. Automação de testes. Infrastructure as Code (IaC). Ferramentas e infraestrutura do ambiente integrado DevOps: Containers, Docker, Kubernetes e OpenShift.

## **Disciplina 4: PRINCÍPIOS DE TESTES DE SOFTWARE**

Ementa: Fundamentos de teste. Testes no ciclo de vida de desenvolvimento de software. Níveis e tipos de testes. Técnicas de geração de dados de teste. Estratégias para teste de software. Abordagens de Test Driven Design (TDD), Acceptance test-driven development (ATDD) e Desenvolvimento Orientado por Comportamento (BDD) e suas aplicações. Mocks e Test Doubles. Test Maturity Model integration (TMMi). Métodos de inteligência artificial em testes de software. Testing as a Service (TaaS). Código de Ética.

## **Disciplina 5: AUTOMAÇÃO DE TESTES**

Ementa: Princípios de Automação de testes. Custo de automação e manutenção. Processo para Testes Automatizados (QAOps). Ferramentas para Automação de Testes. Implantação e configuração de ferramentas. Pipeline de teste não funcional automatizado. Visão geral sobre testes em pipeline e na produção. Automação do processo. Report do processo de teste. Mitigação de erros.

## **Disciplina 6: TESTES DE API**

Ementa: Tipos e protocolos funcionais da API. Princípios de testes de API. Tipos de testes. Principais Ferramentas. Melhores práticas de teste. Processo de teste. Definição de estratégia de teste. Automação de teste. Report do processo de teste. Tendências.

## **Disciplina 7: GESTÃO DA QUALIDADE DE SOFTWARE**

Ementa: Definição de uma estratégia de Qualidade de Software. Gerência dos processos qualidade e Teste de Software. Análise de Valor Agregado (EVA). Technology Readiness Level (TRL). Service Level Objectives (SLO's). Avaliação de metas. Monitoração contínua e Observabilidade. Melhorias de Processo de Teste (QAOps). Cultura DevEx - Developer Experience. Avaliação do ROI da garantia da qualidade. Gerenciamento de Incidentes.

## **Disciplina 8: USER EXPERIENCE (UX): TESTES DE USABILIDADE**

Ementa: Fundamentos de UX. Princípios de teste de UX, usabilidade e acessibilidade. Tipos de testes. Seleção de métodos e técnicas de teste. Habilidades importantes para testes de UX e usabilidade. Métricas de UX e Product Growth. Planejamento, preparação e execução dos testes. Automação de testes. Relatório de avaliação da usabilidade e Acessibilidade. ROI de UX.

### **Disciplina 9: CONTINUOUS INTEGRATION & DEPLOYMENT (CI/CD)**

Ementa: Conceitos de gerenciamento de código e controle de versão. Ferramentas de controle de código e versão. Visão geral e importância da integração e implantação contínua. Ferramentas de CI/CD: visão geral, tipos, líderes do mercado. Relação com os demais processos de engenharia de software. Implantação e configuração de ferramentas de CI/CD. Pipelines de CI/CD: segmentos e melhores práticas.

Automatização do fluxo de trabalho. Métricas recomendadas. Estudos de caso.

### **Disciplina 10: CULTURA E PRÁTICAS DEVSECOPS**

Ementa: Segurança e desenvolvimento ágil. Principais conceitos DevOps e DevSecOps. SDLC(Secure Development Lifecycle). Implementação de end-to-end security. Pipeline DevSecOps. Melhores práticas DevSecOps. Verificação de segurança: (IAST – Interactive Application Security Testing), SAST(Static Application Security Testing), DAST(Dynamic Application Security Testing), RASP(Run-time Application Security Protection). Monitoração de recursos e ambientes. Security Observability.

### **Disciplina 11: ARQUITETURA DE SOLUÇÕES EM NUVEM**

Ementa: Aspectos da Computação em Nuvem: conceitos, tipos, utilização e fornecedores. Componentes de infraestrutura em nuvem (regiões, zonas de disponibilidade). Desenho de soluções de IaaS de acordo com cenários de: elasticidade, balanceamento de carga, alta disponibilidade e DevOps. Arquiteturas de Cloud, Multicloud e Híbridas. Cenários multi-cloud e cloud híbrida: Interconexão entre nuvens públicas e nuvens privadas. Principais soluções de SaaS e PaaS: front-end, back-end, banco de dados e serverless. Segurança. Estratégias de migração de aplicações para provedores de computação em nuvem. Avaliação de viabilidade técnica e financeira (FinOps). Governança de Nuvem.

### **Disciplina 12: TESTES DE APLICAÇÕES MOBILE**

Ementa: Conceitos. Tipos de teste de aplicações mobile. Estratégias de teste para aplicações Mobile. Processos de Avaliação. Avaliação funcional, da usabilidade, de integração, de desempenho, de segurança, de estabilidade e outras. Verificação desempenho das tags, keywords e métricas definidas. Simulação de navegador remoto e visualização de aplicativo. Avaliação em dispositivos reais. Teste de aplicativos móveis. Automatização de testes. Principais Ferramentas.

### **Disciplina 13: MONITORAMENTO E OBSERVABILIDADE**

Ementa: Processo de tomada de decisão. Monitoramento x Observabilidade. Elementos, pilares e benefícios da observabilidade. Estratégias para medições e monitoramento contínuo. Conexão do monitoramento e observabilidade com as estratégias de SLO e Error Budgeting. Principais ferramentas de monitoramento. Abordagem de instrumentação e monitoramento SRE. Application Performance Management (APM). Definição de Dashboard. Monitoramento de aplicações: definição e geração de alertas e relatórios de performance. Utilização de logs, métricas e tracing.

Métricas e medição de maturidade para DevOps. OpenTelemetry.

#### **Disciplina 14: GOVERNANÇA DE DADOS**

Ementa: Contexto organizacional de dados. Conceitos de Governança de Dados (GD). Framework DMBok. Políticas, padrões e procedimentos aplicados aos dados: Data Stewardship, Data Owners, Dados Mestres, Dados Referência, Metadados, Data Catalog. Processo de implantação de GD. Modelos de maturidade de dados. GD aplicada em leis de Proteção (LGPD-GDPR). Compliance e Risk Assessment. GD 2.0: Ética nos dados, Agilidade em GD, Gerência de Mudanças.

#### **Disciplina 15: HUMANIDADES**

Ementa: O ser humano, o processo de humanização e o conceito de pessoa. Desafios contemporâneos e o lugar da religião e da espiritualidade. Autonomia e heteronomia na sociedade atual. Princípios éticos e ética profissional.

#### **Disciplina 16: DATA DISCOVERY E ANALYTICS**

Ementa: Fundamentos da descoberta de dados. Fundamentos e requisitos de aplicações de suporte a decisão. Princípios de projeto, arquitetura e construção de aplicações OLAP. Análise, visualização e comunicação de dados. Ferramentas de Data Discovery e Self-Service Analytics.