1. Acesse a página de cada associação profissional indicada, observando os temas principais abordados em seus sites

a. IEEE – Institute of Electrical and Electronics Engineers

Site: <https://www.ieee.org>

Temas principais:

* Publicações científicas e técnicas (IEEE Xplore)
* Desenvolvimento de padrões internacionais (ex. IEEE 829 – Teste de Software)
* Conferências internacionais em engenharia elétrica, computação, IA, redes e sistemas embarcados
* Educação continuada, workshops e certificações
* Comunidade técnica global com foco em inovação tecnológica

b. ACM – Association for Computing Machinery

Site: <https://www.acm.org>

Temas principais:

* Biblioteca ACM Digital Library (pesquisa e acesso a periódicos e conferências)
* Avanço da ciência e arte da computação
* Especializações como software engineering, HCI, sistemas distribuídos
* Eventos e prêmios como o Turing Award
* Ações educacionais e éticas na computação

C. SBC – Sociedade Brasileira de Computação

Site: <https://www.sbc.org.br>

Temas principais:

* Fomento à pesquisa em computação no Brasil
* Organização de eventos como SBES, CBSOFT, CSBC
* Publicação da SBC OpenLib (SOL) com artigos nacionais
* Apoio à formação de profissionais por meio do POSCOMP
* Discussões sobre políticas públicas e inclusão digital

2. Artigos das bibliotecas digitais sobre Qualidade de Software

Você pode procurar nas seguintes bibliotecas por artigos que tratem de qualidade de software:

1. IEEE Xplore Digital Library

Acesso: <https://ieeexplore.ieee.org>

Exemplo de artigo: *“Software Quality Attributes and Metrics: A Comparative Study”.*Descreve diferentes atributos de qualidade (como confiabilidade, usabilidade, eficiência) com base em padrões ISO e modelos como o ISO/IEC 25010.

1. ACM Digital Library

Acesso: <https://dl.acm.org>

Exemplo de artigo: *“A Systematic Literature Review on Software Quality Models”.* Analisa e compara modelos clássicos como McCall, Boehm, Dromey e ISO/IEC.

1. SBC OpenLib (SOL)

Acesso: <https://sol.sbc.org.br>

Exemplo de artigo: *“Uma Proposta de Extensão do Modelo ISO/IEC 25010 para Qualidade de Aplicativos Móveis”.* Discute como adaptar os critérios de qualidade para a realidade de dispositivos móveis.

3. Pesquise sobre os autores indicados e aponte suas principais obras e contribuições para a Qualidade de Software

A. Barry Boehm

* Obra principal: *Software Engineering Economics* (1981)
* Contribuições:
  + Criador do modelo de ciclo de vida Spiral Model
  + Introduziu o conceito de Custo da Qualidade de Software
  + Fundador do modelo COCOMO (COnstructive COst MOdel)

B. Watts Humphrey

* Obra: *Managing the Software Process*
* Contribuições:
  + Criador do PSP (Personal Software Process) e TSP (Team Software Process)
  + Atuou no SEI e contribuiu para o desenvolvimento do modelo CMM/CMMI

C. David Parnas

* Obra: *On the Criteria to Be Used in Decomposing Systems into Modules* (1972)
* Contribuições:
  + Introduziu o conceito de ocultamento de informação (information hiding) na engenharia de software
  + Pioneiro em modularização e estruturação de software

D. E. W. Dijkstra

* Obra: *A Discipline of Programming*
* Contribuições:
  + Fundador dos fundamentos teóricos da programação estruturada
  + Forte defensor da verificação formal de software

E. C. A. R. Hoare

* Obra: *Communicating Sequential Processes*
* Contribuições:
  + Criador do algoritmo de ordenação QuickSort
  + Introduziu a lógica de Hoare para verificação formal de programas

F. Michael Feathers

* Obra: *Working Effectively with Legacy Code*
* Contribuições:
  + Foco em refatoração e melhoria de código legado
  + Popularização de testes automatizados em sistemas antigos

G. Martin Fowler

* Obra: *Refactoring*, *Patterns of Enterprise Application Architecture*
* Contribuições:
  + Criador de vários padrões de projeto aplicados à engenharia de software
  + Especialista em arquiteturas limpas e ágeis

H. Robert C. Martin (Uncle Bob)

* Obra: *Clean Code*, *Clean Architecture*
* Contribuições:
  + Princípios SOLID
  + Advocacia de práticas ágeis e testes automáticos

I. Kent Beck

* Obra: *Test-Driven Development: By Example*
* Contribuições:
  + Criador do Extreme Programming (XP)
  + Popularizador do TDD – Test Driven Development