
Resumo: Software Quality

O texto aborda caracter sticas que fazem um software de qualidade, n o s o aquelas as quais os usu rios percebem, as externas, mas tamb m aquelas que s o os programadores percebem, as internas. Elas diferenciam um programa amador que s o ser  feito uma vez e usado para um s o caso de um software profissional que ser  editado diversas vezes ao longo dos anos por diferentes pessoas de diferentes  reas, cada vez mais com novas adapta  es e extens es para outros fins. Para isso, precisamos levar em conta o conte do interno para termos um c digo excelente e a experi ncia do usu rio para termos ideia do que disponibilizar para o consumo, afinal usu rios s o imprevis veis e n o s o programadores. Essencialmente, o software precisa ser elegantemente arquitetado e ser facilmente utilizado por qualquer pessoa de qualquer  rea, principalmente de  reas totalmente diferentes da  rea de computa  o.

Corretude:

A qualidade de corretude   a habilidade de um software realizar corretamente aquilo que foi feito para fazer, como realizar qualquer algoritmo sem erro, mas n o necessariamente de uma maneira elegante, ou seja, corretude   o m nimo necess rio para qualquer aplica  o.

Robustez:

A qualidade de robustez   a habilidade de um software reagir inteligentemente em situa  es n o programadas ou induzidas. Por exemplo, ao recebermos um input de um usu rio, precisamos ter certeza de que esse input   v lido, sen o podemos ter problemas no c digo.

Extensibilidade:

A qualidade de extensibilidade   a habilidade de adaptar o software para a realiza  o de tarefas diferentes daquelas originalmente especificadas, como realizar um algoritmo de ordena  o de uma maneira diferente, mas semelhante, como usar um outro elemento no vetor como elemento piv .

Reusabilidade:

A qualidade de reusabilidade   a habilidade de reutilizar aquilo que foi feito no software para outros meios, como a reutiliza  o de classes por meio de classes herdadas e, com isso, a reutiliza  o de propriedades e m todos, al m de reescrita caso necess rio.

Compatibilidade:

A qualidade de compatibilidade   a habilidade de combinar certos aspectos de um software com aspectos de outros, como n o assumir que um formato de data e hora ou de endere o espec fico de uma regi o do mundo ser o sempre usados, pois o software deve ser compat vel para todos os formatos que ser o usados por outras pessoas, assim como formatos de arquivos mais usados em diferentes regi es do mundo.

Resumo: Software Quality

Efici ncia:

A qualidade de efici ncia   a habilidade de um software realizar aquilo que foi feito para fazer de maneira elegante, eficiente, ou seja, usar e ocupar o m nimo de mem ria, ser r pido, ser leg vel, etc.

Portabilidade:

A qualidade de portabilidade   a habilidade de um software funcionar em m quinas e ambientes diferentes, como software criados em linguagens de programac o multiplataforma como Java e Flutter.

Facilidade de Uso:

A qualidade de facilidade de uso   a habilidade de diferentes pessoas de diferentes  reas, muitas vezes pessoas que nem sabem programar, conseguirem usar o software com facilidade e conseguirem resolver os problemas que precisam, como uma interface amig vel, direta e organizada.