Engineer

Kodkvalitet

hur väl koden fungerar, går att förstå och ändra

Vad det är

Kodkvalitet handlar om hur väl källkoden gör sitt jobb och hur lätt andra kan förstå, kontrollera och ändra den. Tänk på en liten faktureringsfunktion. Den ena versionen räknar fram rätt belopp för exempelbeställningen eftersom den följer precis den ordning som utvecklaren tänkte sig. En annan anger valutan, samlar avrundningsregeln på ett ställe och hanterar krediteringar uttryckligen. Båda kan klara testet för en vanlig beställning. Den andra är lättare att utgå från för den som behöver undersöka en ifrågasatt faktura.

Korrekthet omfattar även ogiltiga indata, nya försök och delvis utförda åtgärder. Läsbarhet handlar om vilka beslut nästa person kan förstå vid läsningen. Ändringsbarhet beror på hur mycket ett nytt krav påverkar och om följderna går att kontrollera. En kort funktion kan räkna ut fel skatt. En krånglig funktion kan fungera i dag och göra det svårt att kontrollera en ny valuta i morgon.

Kodgranskning hjälper ett team att bedöma egenskaperna i en viss ändring. En spärr kan hejda ändringen när valda kvalitetskontroller misslyckas. Teknisk skuld handlar om framtida kostnader som följer av ett val. Kodkvalitet beskriver kodens tillstånd. Googles granskningsregel ger frågan en praktisk form genom att be granskaren bedöma om ändringen förbättrar kodbasen som helhet.

Använd begreppet i ett beslut

Använd ”Vad är kodkvalitet?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Ett team som tar en idé från en skiss till något dina kunder kan logga in i, i stället för att lämna den vidare mellan tre leverantörer.

Se våra tjänster Hör av er →

Granska antagandena bakom ordet

Använd förklaringen och det länkade arbetet för att formulera frågor till projektet. Vi kan undersöka antagandena tillsammans innan ett verktyg väljs. Ledde GIS-kartappen till 10 gånger högre intäkter — 10x intäktstillväxt, med kartprodukten i spetsen

Ledde GIS-kartappen till 10 gånger högre intäkter Hör av er →

Varför det spelar roll

Kvaliteten märks ofta när ett krav ändras. En tjänst räknar på offerter, skickar fakturor och betalar tillbaka pengar. Om samma rabattregel finns i tre varianter måste teamet hitta och uppdatera alla vid nästa kampanj. Skillnaden kanske märks först vid en återbetalning. En gemensam regel på ett känt ställe och tydliga indata från varje anrop gör det lättare att förstå ändringen och avgöra vilka flöden som behöver testas.

Leta efter belägg som svarar mot risken. För belopp kan man välja ett fall där avrundning per rad ger ett annat resultat än avrundning av summan. För en åtgärd som kan köras igen testar man om en andra begäran leder till ytterligare en debitering. Om beräkningen körs vid varje köp går det att mäta svarstiden under väntad belastning innan teamet lägger till en cache. Varje observation säger något under bestämda förhållanden.

Det finns också ett värde vid överlämning. Människor byter projekt och nästa granskare kanske saknar kunskap om modulens historia. Tydliga namn, ett körbart exempel och en kort förklaring till ett ovanligt villkor gör en bedömning möjlig utan att vänta på den som skrev koden. Det är viktigt både vid introduktion och vid felsökning utanför arbetstid.

Så ser det ut i praktiken

Anta att ett företag avrundar summan av skattebeloppen för varje skattesats. En faktura har två rader med ett oavrundat belopp om 0,005 valutaenheter på varje rad. Avrundas båda raderna till hundradelar, med avrundning uppåt precis på gränsen, blir summan 0,02. Summerar man först och avrundar en gång blir den 0,01. Det går att kontrollera med huvudräkning. Regeln för en verklig faktura måste komma från produktkraven och de bestämmelser som gäller.

Ange vilka enheter funktionen förväntar sig och i vilken ordning den räknar. Mellanresultaten måste bevara den precision som behövs. Gör grupperingen tydlig och avrunda vid det överenskomna steget. Ett test med talen ovan skiljer

mellan de två reglerna. Ett annat kan blanda skattesatser så att en ändring inte råkar ersätta grupperna med en enda totalsumma. Om återbetalningar har en annan regel behövs ett särskilt exempel.

Följ beloppet som kunden får se, beloppet som exporteras och anropet som görs vid en återbetalning. Äldre fakturor ska behålla de belopp som sparades när de utfärdades. Ett test av hela flödet kontrollerar att rätt enhet når beräkningen. Ett isolerat test visar vad beräkningen gör med kända indata.

I vår anteckning om att sälja den sista biljetten en enda gång används samma arbetssätt för lagersaldot. Den beskriver en likhet som måste gälla hela tiden, beskriver tester med samtidiga försök och föreslår att likheten övervakas i drift. Den ger granskaren ett konkret krav som behöver bevaras vid ändringar i flera flöden.

Var det går fel

En mätbar signal kan förväxlas med ett bevis. Täckning visar vilka rader eller grenar som testet körde. Det kan ändå förvänta sig fel avrundningsregel. En komplexitetssiffra kan peka ut kod värd att läsa. Att sprida en tydlig jämförelse över flera otydliga hjälpfunktioner kan ändå göra felsökningen svårare. Bestäm först vilket fel den nya mätningen ska fånga och undersök exempel på båda sidor om gränsen.

Upprensning kan få fel prioritet. Ett otydligt namn i en sällan använd rapport väger annorlunda än ett oklart villkor som används vid alla betalningar. Beskriv risken, skilj nödvändiga rättelser från valfria förbättringar och bestäm när frågan behöver följas upp. En gemensam regel och automatisk formatering kan ofta lösa en diskussion om kodens utseende.

En lokal förbättring kan till sist flytta problemet. En gemensam rabattfunktion passar när flera anrop följer samma regel. Den blir svår att förstå om den måste gissa vad som gäller för en förnyelse, en kampanj eller en återbetalning. Ta reda på vilka indata och villkor som faktiskt är gemensamma. Dela den stabila delen, gör skillnaderna tydliga och kontrollera resultatet.

Läs mer

- Googles bedömningsgrund för att godkänna kodändringar

- Vad Google granskar i en kodändring
- Enhetstestning i Software Engineering at Google

Ordlista

Välj utifrån era faktiska förutsättningar

Beskriv arbetsflödet, befintliga gränssnitt och användarna som behöver förändringen. Vi kan definiera en testbar leverans och en överlämning som gör att teamet kan underhålla lösningen.

Se vad det kostar Hör av er →

Gör begreppen praktiska

Fråga hur ”Vad är kodkvalitet?” gäller i er organisation. Ett konkret exempel från ert nuvarande system räcker för att börja samtalet.

Se våra tjänster

welcome@engineer.company +45 50 10 61 84 Köpenhamn, Danmark

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/glossary/code-quality/