Engineer

Regresjonstesting

testing av om en endring bevarer eksisterende atferd

Hva det er

Regresjonstesting undersøker om programvaren fortsetter å oppføre seg riktig etter en endring. Et nytt eksportvalg kan oppfylle kravet til ny funksjonalitet samtidig som det endrer hvilke rader eksisterende brukere får. En test av den nye innstillingen kan vise at den virker. En regresjonstest går også gjennom det tidligere forløpet og ser om atferden andre avhenger av, fortsatt holder.

Håndboken til NASA beskriver regresjonstesting som en måte å sjekke at en endring ikke har medført nye feil. Den anbefaler blant annet å kjøre eksisterende tester knyttet til endret kode eller krav og å undersøke moduler som virker sammen. Fremgangsmåten gjelder også ved endring av avhengigheter, innstillinger, data eller kjøremiljø. Atferden kan endre seg selv om kildekoden er urørt.

Bekreftelse av en rettelse og regresjonstesting utfyller hverandre. Gjenskap problemet og sjekk at det nye resultatet er riktig. Prøv deretter nærliggende atferd som skal bevares. Behold gjerne det første eksempelet som en vakt mot at feilen kommer tilbake. Forventet resultat skal uttrykke hva produktet lover. Hvis en regel endres med vilje, oppdater forventningen og dokumenter forskjellen for eksisterende brukere.

Bruk begrepet i en beslutning

Bruk «Hva er regresjonstesting?» i en konkret beslutning. Vi kan knytte begrepet til systemet dere bygger eller kjøper. Ett team som tar en idé fra skisse til noe kundene dine kan logge inn på, i stedet for at den sendes videre mellom tre leverandører.

Se tjenestene våre Ta kontakt →

Undersøk antakelsene bak ordet

Bruk forklaringen og det tilknyttede arbeidet til å formulere spørsmål for prosjektet. Vi kan undersøke antakelsene sammen før et verktøy velges. Ledet en GIS-kartapp til 10 ganger mer omsetning — 10x inntektsvekst, ledet fra kartproduktet

Ledet en GIS-kartapp til 10 ganger mer omsetning Ta kontakt →

Hvorfor det er viktig

En endring kan få virkning langt utenfor linjene som er redigert. Bytter man en liste mot en mengde, kan rapporten miste duplikater og få en ny rekkefølge. En datofunksjon kan godta et nytt format og samtidig avvise tomme verdier fra et annet system. Automatiske prøver kan knytte virkningen til en fersk endring. Det er særlig verdifullt for integrasjoner som få i kodearbeidet kjører lokalt.

Start med nærmeste felles regel og et par faktiske, forskjellige kall hvis endringen er liten. Spør hvilke standardinnstillinger, grenseverdier, feilsituasjoner eller lagrede data som påvirkes. For endringer med store følger kan ytelse og tilgjengelighet trenge oppmerksomhet ved siden av de enkelte verdiene. Belegget må passe til virkningen av en oversett feil. En enhetstest kan undersøke en omregning nøyaktig. En prøve av tastaturbetjening kan kreve en ekte nettleser langs den virkelige brukerreisen.

Et godt navn forklarer hvorfor testtilfellet finnes. «Manglende dato forblir tom ved eksport av eldre saker» forteller mer enn «tilfelle tre består». Vis de vesentlige inndataene og resultatet. Da kan den neste som vedlikeholder koden, se om et krav er endret med vilje eller om noe har gått i stykker.

Slik ser det ut i praksis

Tenk på en CSV-eksport med et valg for å ta med arkiverte kontoer. Standardeksporten skal fortsatt utelate dem. Aktive kontoer er sortert etter referanse, og et tomt felt for siste kontakt skal forbli tomt. Valget legger til arkiverte kontoer uten å gjenta aktive rader. Sett to aktive kontoer i omvendt rekkefølge og én arkivert konto i testdataene. La kontaktdatoen mangle for én konto. Bruk den faktiske eksportøren i begge forsøk.

Sammenlign kontoreferansene, rekkefølgen og det tomme feltet ved standardeksport. Med valget på skal aktive og arkiverte kontoer forekomme én gang hver. Ta med et navn som inneholder komma. Hvis eksportøren bare setter sammen tekst, kan den gi riktig radantall med feil innhold i kolonnene. Les CSV-filen med samme type leser som en virkelig mottaker. Testdataene kan være små så lenge effekten blir synlig.

Undersøk om testene avviser en reell regresjon. La midlertidig standardeksporten ta med arkiverte kontoer, eller fjern anførselstegnene fra navnet med komma. Den relevante testen bør feile med forskjellige og lesbare grunner. Gjenopprett den gode utgaven og prøv igjen. En kontroll som bare ser etter en fil, kunne sluppet begge feilene gjennom. Kjør også et gyldig tilfelle uten arkiverte kontoer.

Vårt notat om understreking og delsett av skrifter i Safari viser en annen side ved bevisene. En CSS-endring så ut til å reparere visningen i Chromium, mens Safari fortsatt trakk streken gjennom de nedre delene av bokstavene. En regresjonstest av dette ville trenge observasjoner av nettleserne og tilfellene den sier at den dekker. Notatet beskriver diagnosen og det som ble observert. En test kan holde fast på den innsikten.

Hvor det går galt

I en eksport kan også hjemmel, anonymisering og tilgang sjekkes dersom endringen berører dem. En test der mottakeren har alle tillatelser, viser ikke hva som skjer for en ordinær konto. Velg derfor en representativ mottaker før du stoler på at filen er riktig.

Mange tester kan gi et misvisende inntrykk av dekning. Hundre enkle kontroller kan overse den eneste integrasjonen som leser historiske data. Velg tilfeller ut fra faktiske forløp, data og berørte avhengigheter. Når et problem har sluppet gjennom før, beskytt det berørte forløpet og forsikre deg om at testen avviser den gamle feilen.

Man kan også sammenligne for mange detaljer. Et fullstendig øyeblikksbilde kan forandre seg med tilfeldige identifikatorer eller klokkeslett. Sjekk rekkefølgen der den er avtalt. Sammenlign mengden og let etter duplikater hver for seg dersom bare utvalget er avtalt. Vurder endringer i forventet resultat like grundig som en endring i koden.

Til sist gjør ustabile tester det vanskelig å forstå feil. Vent på en observerbar tilstand med en frist, bruk kjente data og rydd opp etter hvert forsøk. Behold brede kontroller der virkelige samspill teller, og la raske, avgrensede tester følge den daglige utviklingen. De gir belegg på forskjellige nivåer.

Les mer

- NASAs håndbok om regresjonstesting av programvare
- Dokumentasjonen til pytest om parametere i testtilfeller
- Enhetstesting i Software Engineering at Google

Ordliste

Velg ut fra deres faktiske rammer

Beskriv arbeidsflyten, eksisterende grensesnitt og brukerne som trenger endringen. Vi kan definere en testbar leveranse og en overlevering som gjør teamet i stand til å vedlikeholde løsningen.

Se hva det koster Ta kontakt →

Gjør begrepene praktiske

Spør hvordan «Hva er regresjonstesting?» gjelder i organisasjonen deres. Et konkret eksempel fra dagens system er nok til å starte samtalen.

Se tjenestene våre

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

Ta med deg denne siden

Visittkort

Laget med kjærlighet av D.

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

Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.

Skrevet ut fra https://engineer.company/nb/glossary/regression-testing/