Engineer

Kontrakttest

test af aftalte grænseflader mellem komponenter

Hvad det er

En kontrakttest kontrollerer, at to komponenter er enige om de anmodninger, svar eller beskeder, de udveksler. En side til pakkesporing og tjenesten bag den kan bestå hver sin test. Siden fejler alligevel, hvis den forventer feltet `status`, og tjenesten sender `state`. En test af grænsefladen udtrykker, hvad siden bruger, og efterprøver, at den anden side kan levere det.

Pact beskriver en metode, hvor forbrugeren kører mod en kontrolleret erstatning for udbyderen. Testen gemmer udvekslingerne i en kontrakt. Udbyderen efterprøver den i sin egen implementering under bestemte forudsætninger. For beskeder handler beviset om, hvad den ene side producerer og den anden læser. De to sider kan udvikle og udgive i forskelligt tempo, når beviserne handler om de versioner, som skal virke sammen.

Klienten kan også have en aftalt adfærd ved en ukendt værdi. Hvis en manglende status kan vises som »Ukendt status«, behøver en ny værdi ikke ødelægge hele siden. Det kræver en forretningsbeslutning og et testtilfælde, der dokumenterer, hvad klienten gør.

Kontrakttest kan også betyde, at en udbyder kontrolleres mod en offentlig grænsefladebeskrivelse som OpenAPI. Det passer, når den skrevne specifikation er selve aftalen. Den kan angive felter, statuskoder og tilladte værdier. Eksempler fra forbrugeren kan desuden vise, hvilke kombinationer et bestemt kald bruger. Metoderne kan bruges sammen. Et testforløb gennem den virkelige integration kan afdække forhold uden for den enkelte udveksling, som rute, adgang og opsætning.

Brug begrebet i en beslutning

Brug »Hvad er en kontrakttest?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Den del, brugerne aldrig ser: hvor data ligger, hvilke regler der gælder, og hvordan andre systemer må tale med den.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har designet og drevet 6 ETL/ELT-pipelines — 6 ETL-pipelines på tre databasemotorer

Har designet og drevet 6 ETL/ELT-pipelines Kontakt os →

Hvorfor det betyder noget

En grænseflade lever ofte længere end den kode, som først leverede den. Tjenestens hold kan omdøbe et felt og opdatere egne tests uden at ændre en klient i et andet projekt. En mobilapp kan fortsat køre en gammel version i flere måneder. Selv ændring af typen på et felt betyder noget, når feltet vælger en tekst, brugeren ser. Med en efterprøvbar kontrakt kan holdene se den afhængighed før frigivelse.

Uafhængige udgivelser øger værdien af viden om kompatibilitet. En ældre klient forstår måske kun få statustyper, mens udbyderen har brug for en ny mellemtilstand. Holdene kan forberede forbrugeren først, bevare begge former gennem overgangen og fjerne den gamle, når ingen understøttet version bruger den. En sikker rækkefølge afhænger af registrerede aftaler, den virkelige brug og udgivelsespolitikken. Et vellykket forsøg med de to nyeste udgaver dækker ikke automatisk de tidligere.

Testen bør beskrive det, der betyder noget for aftalen. Er enhver gyldig reference acceptabel, så kontrollér formen og undlad at låse en bestemt eksempelværdi. Vælger et felt en synlig status, så kontrollér typen og de værdier, klienten forstår. Et præcist eksempel beskytter aftalen uden at fryse udbyderens interne løsning.

Sådan ser det ud i praksis

En tænkt app sporer pakker ved hjælp af en reference. Når pakken findes og er på vej, forventer siden et vellykket svar, en reference og en tekstværdi, den kan omsætte til en status. Når pakken er ukendt, forventer den et andet svar og giver mulighed for at kontrollere referencen. Hver udveksling har en forudsætning, for eksempel at pakken er på vej, eller at referencen ikke findes.

Kør forbrugerens virkelige anmodningskode mod den kontrollerede erstatning. Kontrollér resultatet i appen. Afspil de samme udvekslinger på udbyderens virkelige svarvej. Opret forudsigelige testdata for den aftalte tilstand, og nulstil dem mellem forsøg. Erstatter testen en database eller transportørens system, så behold den anmodningsbehandling, programlogik og dannelse af svaret, som forbrugeren afhænger af.

Omdøb midlertidigt `status` til `state` uden at ændre klienten. Kontrollen skal angive det manglende felt og pege på den relevante udveksling. Et ekstra valgfrit felt, som forbrugeren ignorerer, kan være i orden, hvis kontrakten tillader det. Knyt kontrollerne til de præcise versioner. En blanding af nye resultater og gamle ukendte eksempler gør det svært at se, hvilken kombination der blev efterprøvet.

Vores note om at åbne en app i mobile Solana-tegnebøger viser integrationer, der forventer forskellige adresser og kodning af parametre. Noten beskriver forskellene og forsøg på en enhed. En test af de dannede links kan fastholde deres forventede form. Et forsøg på en rigtig enhed kan vise, om browseren sender linket videre til den installerede app, og om forløbet lykkes. Det er to forskellige observationer.

Hvor det går galt

En kontrakt kan blive for stiv, hvis hver byte i svaret skal være som i et gemt eksempel. En udbyder kan have frihed til at tilføje et ubrugt felt eller ændre formen på en besked. Kræv fast rækkefølge, enhed eller ordlyd, når klienten afhænger af det, eller den offentlige specifikation kræver det.

En test mister værdi, hvis den går uden om grænsen. Et håndskrevet JSON-objekt kan virke uden at køre koden, der normalt danner svaret. En erstatning kan levere et aftalt eksempel, uden at klienten kan læse det gennem den normale vej. Kør den rigtige kode for den side, der efterprøves, og dokumentér de afhængigheder, der er erstattet.

Et enkelt lykkeligt forløb siger endelig lidt om manglende data, ukendte referencer eller gamle versioner. Find de fejltilstande, klienten håndterer, og de versioner, udbyderen skal understøtte. Behold integrationstest af opsætning og ruter. Kontrakten giver så et mindre, hurtigt bevis, der kan anvendes ved hver relevant ændring.

Læs mere

- Pacts beskrivelse af kontrakttest
- Sådan efterprøver Pact forbrugerens og udbyderens kontrakt
- Dokumentation af udbydertilstande i Pact
- OpenAPI-specifikationen for HTTP-grænseflader

Ordliste

Vælg ud fra jeres faktiske vilkår

Beskriv arbejdsgangen, de eksisterende grænseflader og de brugere, der har brug for ændringen. Vi kan definere en testbar leverance og en overdragelse, der gør teamet i stand til at vedligeholde løsningen.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er en kontrakttest?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

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

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

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

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/contract-testing/