

Fejlhåndtering via FMK-servicekald - Udvidet validering

Implementeringsnote

Implementeringsnotatet er tiltænkt, at være supplement til kodedokumentation der er beskrevet på [Wiki for FMK-Fejlhåndtering](#). Notatet er formidlingsmæssigt opdelt i to:

- Kapitel 1-2, er henvendt til projektledere, it-arkitekter, forretningsansvarlig og udviklere, der skal forstå hvad udvidet validering forretningsmæssigt understøtter og hvordan løsningen konceptuelt er opbygget.
- Kapitel 3, skal give konkret hjælp til udviklingsressourcer til at implementere servicekald mod FMK, som returnerer udvidet validering til anvendelsesystemet.

Dato
2020-01-30

Projektnavn
Fælles Medicinkort

Ansvarlig
Åse Grønberg Sørensen
Thomas Sonne Olesen

Telefon
+45 91334817
+45 20153376

E-mail
ags@sundhedsdata.dk
tso@lakeside.dk

1. Motivation og baggrund

Kapitlet beskriver kort motivationen og baggrunden bag muligheden for at foretage forespørgsler med FMK-servicekald med udvidet validering ud fra et forretningsmæssigt perspektiv.

Der har vist sig en række tilfælde, hvor FMK modtager data, der ser ud til at være fejlindtastninger, men som FMK ikke kan afvise, da de kan være rigtige. Dette kan f.eks. være et CPR-nummer, der ikke kendes, men faktisk er en nyfødt. I disse tilfælde er det ønskeligt at kunne advare brugeren, men også give mulighed for at gennemføre handlingen, hvis det er korrekt.

På den baggrund blev det i 2017 besluttet i FMK's goverancemodel, at give FMK mulighed for at returnere valideringsfejl, der kan tilsidesættes af brugeren. Dette betegnes som "udvidede valideringer".

Efterfølgende har det vist sig, at muligheden for centralt at kunne tilføje sådanne advarsler, kan være en effektiv måde at få implementeret hjælp til klinikere på; hjælp som det ellers grundet de almindelige udviklingstider, prioritering og certificering ville tage op til 18 måneder at få i produktion.

Endelig er der situationer, hvor FMK har adgang til data, som ikke er tilgængelig decentralt, og hvor FMK derfor kan lave en advarsel, som ikke umiddelbart er mulig at implementere i de decentrale systemer.

Derfor er der stor interesse for at få lavet en generisk model for håndtering af sådanne centrale udvidede valideringer, som man kan regne med, er understøttet af alle systemer.

Muligheden for brug af udvidede valideringer har f.eks. været overvejet i forbindelse med advarsler på forkert dosering af risikolægemidlet Methotrexat eller i forbindelse med udskrivning af recepter på udgåede lægemidler.

Fejlhåndtering i forbindelse med FMK-servicekald har derfor overordnet tre formål:

1. **Anvendelsesmæssigt – ved almindelig validering/fejlhåndtering:**
At tilbyde brugeren en fejl-dialog, hvis FMK-servicekaldet (opslag/opdateringer/oprettelser af data) ikke kan gennemføres.
2. **Anvendelsesmæssigt – ved udvidet validering:** At tilbyde brugeren en advarsel, hvis FMK-servicekaldet potentielt kan indebære fejl. Advarslen sikrer, at brugeren aktivt tager stilling til om handlingen bekræftes (advarslen overstyres) eller handlingen annulleres (advarsel anerkendes). Når handlingen bekræftes, antages det, at brugeren er klar over konsekvensen det medfører og kan foretage eventuelt korrigerende handling i forhold til dette.
3. **Arkitekturmæssigt:** At tilbyde en generisk valideringsmekanisme, der sikrer at fejlhåndtering med advarsler kan udvides/vedligeholdes centralt. Det kan have stor betydning, hvis der opstår udfordringer i forhold til patientsikkerhed, hvor der hurtigt kan tilføjes en advarsel centralt og ikke skulle forcere ændring af samtlige decentrale systemer. Derudover kan muligheden for at lave generelle valideringer centralt betyde at man beslutter at visse valideringer/advarsler implementeres centralt, for derved at forenkle kompleksiteten i de decentrale løsninger (f.eks. kunne man have valgt, at lave den i certificeringskravene krævede advarsel mod dobbeltordinationer, som en central udvidet validering).

Svar der returneres som følge af udvidet validering

Udvidet validering, er en del af [FMK fejlhåndtering](#) og gør brug af fejl-koderne fra [10000-10999](#). Fejlteksten vil indeholde en beskrivelse af fejlen på samme måde som almindelige fejl, men vil herudover også returnere en alternativ tekst, der udover information om advarslen også

beskriver konsekvensen ved at fortsætte. Dette kunne f.eks. være noget i stil med:

*Recepten på {drug} anviser en pakning, der for tiden ikke kan leveres.
Hvis der fortsættes, fastholdes den angivne pakning, som patienten
sandsynligvis ikke kan få på apoteket.
Ønsker du at fortsætte?*

Understøttelse

Fra og med FMK snitfladeversion 1.4.4 E2 skal anvendelse systemer kunne håndtere udvidet validering. FMK vil potentielt kunne returnere en udvidet validering for alle modificerende kald fra og med denne version.

2. Konceptuel løsningsbeskrivelse

Kapitlet beskriver, hvordan udvidet validering er konceptuelt opbygget, og skal hjælpe leverandørerne med at implementere håndteringen af udvidet validering mest hensigtsmæssigt ud fra et brugerperspektiv.

Det er ønsket fra FMK-programmets side, at sikre mulighed for løbende at kunne etablere nye udvidede valideringer, når disse er godkendt i FMK-governancemodellen. Derfor er det væsentligt, at anvendersystemet udvikler en generel funktionalitet, der håndterer valideringsfejl på en måde, så brugeren kan afgøre, hvad der skal ske.

For at kunne gøre dette generisk, skal alle udvidede valideringer følge et fast skema. Valideringsadvarslerne er opbygget efter følgende faste sætningskonstruktion:

Feltet "WarningQuestion" vises (i stedet for "FaultText")

Spørgsmålet "Ønsker du at fortsætte?" vises

Knapper med "Nej" og "Ja"

Hvis dialogen ikke giver mulighed for at svare "Ja" eller "Nej", men blot skal vise en fejlbesked, vil den have følgende opbygning:

Feltet "FaultText" vises

Knap med "Ok"

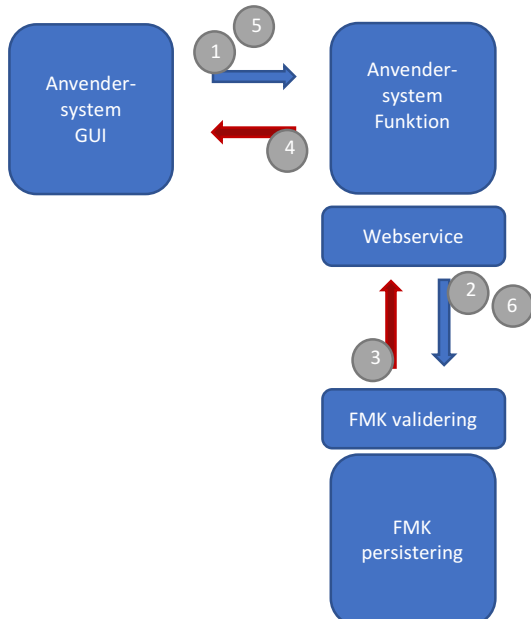
Ved serviceintegrationer er det normalt, at et servicekald enten fejler, og dermed ikke foretager opslag/oprettelse/opdatering af data, eller handlinger i servicekaldet gennemføres. Dette gælder også FMK-servicekald, hvor en valideringsfejl (fejlkode 1-9999) vil medføre, at oprettelse/opdatering ikke udføres.

I disse tilfælde vil anvendersystemet bede brugeren om at rette data til, og sende igen.

Men med udvidet validering (fejlkode 10000-10999) skal anvendersystemet vise den returnerede advarsel for brugeren. Hvis brugeren ønsker at fortsætte (på trods af advarslen), kan anvendersystemet kalde FMK igen med en angivelse af, at tidligere returnerede advarsel skal ignoreres af FMK og handlingen skal gennemføres. Et FMK-serviceresponse indeholder altid kun den første fundne valideringsfejl i FMK-servicekaldet. Således skal anvendersystemet genfremsende FMK-servicekald med svar på de konkrete valideringer, så længe FMK-servicerespons melder valideringsfejl tilbage, for at gennemføre forespørgslerne.

Det er også vigtigt at huske tidligere overstyringer, således at man ikke ved genfremsendelse risikerer at brugeren spørges om en allerede tidligere besvaret validering. Overstyringen bør ikke huske længere end til næste kontekstskift (f.eks. skift af patient eller skærbillede).

Nedenfor er konceptet for FMK-servicekald illustreret ved anvendelse af udvidet validering med en eller flere forespørgsler.



Figur 1: Illustrerer flow imellem anvendersystemets FMK-servicekald mod FMK's valideringsmekanisme

Trin 1: Brugeren beder anvendersystemet om at udføre en handling via GUI.

Trin 2: Anvendersystemet omsætter handlingen til et eller flere FMK-servicekald, der sendes i et **Request**.

Trin 3: FMK returnerer et **Response**, som indeholder en af følgende varianter:

3a: at FMK servicekald ikke medfører valideringsfejl.

3b: at FMK servicekald medfører valideringsfejl på den først identificerede handling med fejl på fejlkode fra 1-9999.

3c: at FMK servicekald medfører den første identificerede handling med

advarsel på fejlkode fra 10000-10999.

Trin 4: Anvendersystemet viser brugeren advarsel/fejl via dialog med fejltekst (WarningQuestion), som findes i det returnerede **Response**.

Trin 5: I tilfælde af 3c forholder brugeren sig aktivt til advarslen, ved at besvare spørgsmålet "Ønsker du at fortsætte?", hvor handlingen bekræftes med "Ja" (advarsel overstyres) eller handlingen annulleres med "Nej" (advarsel anerkendes).

Trin 6: Hvis der skal fortsættes, sender anvendersystemet et opdateret **Request** med "over-rule" af den pågældende handling.

Herefter gentages trin 3, hvor eventuelle nye advarsler/fejl returneres i et nyt **Response** - såfremt der er flere handlinger, der resulterer i valideringsfejl.

Forhåndsaccept af udvidet validering

I nogle tilfælde er det ønskeligt at overstyre kendt advarsel på forhånd. Eksempelvis vil fødselsafdelinger blive overbebyrdet med advarsler for

nyfødte, da nyfødtes CPR-numre ofte ikke er kendt i FMK. Lige præcis fødselsafdelinger – som konsekvent har med nyfødte at gøre og som givetvis har styr på, at de anvender korrekte CPR-numre, der endnu ikke findes i det centrale CPR-register – kan godt vælge konsekvent at ignorere valideringsreglen vedr. nyfødt – fejlkode 10000 "*Personen {CPR} er ukendt. Gennemføres handlingen, vil personen bliver opfattet som nyfødt.*". På den måde undgår brugeren at blive konfronteret med denne valideringsregel. Valideringsreglen vedr. nyfødte skal primært hjælpe andre parter end fødselsafdelingerne.

I andre tilfælde, hvor en udvidet validering skønnes ofte at forekomme, kan man i anvendersystemet eventuelt lade brugeren tage stilling til spørgsmålet inden det sendes til FMK, f.eks. med en check-box. F.eks. kan man i anvendersystemet have et akut flag på dosisdispenserede ordinationer, der informerer brugeren om, at apoteket skal kontaktes uden om FMK, for at ændringen bliver effektueret øjeblikkelig. I så fald er det ok, at overstyre advarslen om at ændringen ikke passer med næste dosispakning. Hvis brugeren markerer, at det skal ske akut, kan man i requested slå den pågældende validering fra, således at kaldet bliver gennemført uden brugeren præsenteres for advarslen.

Generisk håndtering

Afhængig af anvendersystemets opbygning, skal man understøtte forskellige typer håndtering af udvidede valideringer.

- **Model 1: Udvidet validering af "enkelt handling"**
Anvendersystemet benytter FMK-servicekald, der resulterer i udvidet validering med en enkeltstående handling af gangen, som brugeren forholder sig til enkeltvis.
- **Model 2: Udvidet validering af "flere handlinger" (bulkopdatering)**
Anvendersystemet benytter FMK-servicekald, der resulterer i udvidet validering med to eller flere handlinger. Dette er det mest komplekse tilfælde og kan indebære store krav til den generiske håndtering, og anbefales derfor ikke, med mindre der samtidig implementeres en omfattende håndtering af fejl fra FMK, se evt. mere på:

<https://wiki.fmk.netic.dk/doku.php?id=fmk:generel:transaktioner>

Hvis et system kun anvender model 1, er håndtering relativt simpelt. Hvis brugeren ønsker at gennemføre handlingen, slås valideringen fra og kaldet gennemføres – alternativt undlader man blot at sende kaldet igen, hvorved handlingen ikke gennemføres. Herefter kan brugeren vælge at rette til og evt. sende et nyt kald.

Når anvendersystemet anvender model 2, fremsendes flere handlinger i et Request (Figur 1 - trin 2).

Nedenfor er nævnt en række hensyn, som anvendersystemet skal forholde sig til i den forbindelse.

- Et modtaget Response vil kun aflevere svar på én advarsel i rækken af potentielt flere advarsler (Figur 1 - trin 3). Derfor skal det opdaterede Request fra anvendersystemet indeholde samtlige handlinger på-ny, inklusiv "over-rule" af tidligere accepterede valideringsfejl (Figur 1 - trin 6). Eksempelvis i forbindelse med opdatering af flere lægemiddelordinationer, hvor det forudsættes, at anvendersystemet indsætter "over-rule" på et konkret underelement/lægemiddel, som responset har refereret til. En undtagelse er valideringen på ikke-eksisterende cpr-numre (fejlkode 10000), hvor valideringsfejlen vil gælde alle underliggende lægemiddelordinationsoprettelser.
- Såfremt at brugeren vælger at svare "Nej" til en udvidet validering, betyder det, at ingen handlinger i Request'et udføres. Dette svarer til situationen, hvor en enkelt handling i et Request resulterer i en fejl.
Det anbefales ikke at man forsøger at udføre de øvrige handlinger, da der kan være kliniske afhængigheder mellem handlingerne, så når en bortfalder, bør klinikerens tage stilling til den samlede opdatering.
Hvis f.eks. en oprettelse af en recept resulterer i en valideringsfejl, som ikke skal overstyres, er det muligt, at præparater er en del af en kombinationsbehandling, og derfor skal en af de øvrige recepter således heller ikke oprettes.
- Såfremt at brugeren vælger at svare "Ja" til en udvidet validering, og en af de efterfølgende handlinger giver en hård fejl (fejlkode fra 1-9999), vil det være en hjælp for brugeren, hvis anvendersystemet husker dette, så at brugeren ikke spørges igen, når requestet fremsendes på ny, efter den hårde fejl er håndteret af brugeren.

Hjælp til implementering af udvidet validering uddybes i det næste kapitel.

3. Hjælp til implementering

Kommentar

Beslutninger, der påvirker dokumentation/implementering:

- Faulttext og WarningQuestion vil ikke overstige 1000 tegn.
- ID'er refereret til i faulttext gøres læsbar for brugerne.
Eksempelvis bliver medicinID opløst til konkret navn på medikament.

Præsentation af advarsel på forespørgsel – model 1

I figur 1 – trin 4 og 5, forventes anvendersystemet at kunne præsentere advarsel eller fejl på det fremsendte FMK-servicekald, som brugeren skal forholde sig til i en advarselsdialog. I de konkrete kendte tilfælde, kan anvendersystemet lave en mere brugervenlig løsning, der f.eks. sender brugeren hen til de data, der skal rettes, for at undgå valideringsfejl. Men da man ønsker en mulighed for at centralt at kunne tilføje nye valideringer, skal anvendersystemerne understøtte en generisk tolkning af det udvidede valideringsresponse.

Afsnittet vil tage udgangspunkt i en sådan 'case', hvor brugeren bliver præsenteret for en advarselsdialog, og hvor der er tale om brug af **model 1: Udvidet validering af "enkelt forespørgsel"**.

Pseudokode for FMK-servicerequest – Præsentation af advarsel på forespørgsel

```

sendRequest(request) {
    response or fault = sendToServer(request)
    if (fault) {
        if (isExtendedValidationError(fault)) {
            promptUser( fault.WarningQuestion +
                        "Ønsker du at fortsætte?")
            if (yes) {
                addSkipValidationToRequest(request,
                                           fault.faultCode,
                                           fault.faultDetails.ElementPath)
                sendRequest(request)
            } else {
                abort()
            }
        } else {
            doNormalErrorHandling(fault);
        }
    } else {
        // Return successful response
        return response;
    }
}

```


Pseudokoden ovenfor håndterer udvidet validering ved at præsentere en dialogbox og viser brugeren advarselsteksten 'fault.WarningQuestion' og tillader valget mellem:

"Ja" - 'addSkipValidationToRequest' forespørgslen bekræftes og advarsel overstyres, eller

"Nej" - 'abort' forespørgslen annulleres og advarsel anerkendes.

Hvis der returneres en fejl, som ikke er en udvidet validering, kaldes 'doNormalErrorHandling'.

Forespørgsel bekræftes (advarsel overstyres) – model 1

I figur 1 - trin 6, forventes det, at anvendersystemet skal kunne ignorere den enkelte valideringsfejl på en forespørgsel i FMK-servicekaldet. Næste skridt er derfor at definere hvad 'addSkipValidationToRequest' og tilhørende 'locateElement' indebærer.

Pseudokode for FMK-servicerequest – Forespørgsel bekræftes (advarsel overstyres)

```
addSkipValidationToRequest(request, faultCode, elementPath) {  
    element = locateElement(request, elementPath);  
    metadata = element.addChild(ModificationMetadata)  
    metadata.text = "skip validation for (" + faultCode + ")"  
}  
  
locateElement(request, elementPath) {  
    element = request.body  
    for each subPath of elementPath split on "." {  
        childName = subPath up to "["  
        n = subPath between "[" and "]"  
        element = find the n'th child of element named childName  
    }  
    return element  
}
```

I 'addSkipValidationToRequest' fremfindes det relevante element via funktionen 'locateElement' baseret på informationerne i 'elementPath', som returneres fra FMK.

Derefter tilføjes det metadataelement, der overstyrer valideringen, når requested gendesendes.

Så længe brugeren vælger at bekræfte at anvende 'addSkipValidationToRequest', vil forespørgslen forsætte. Men så snart at brugeren afviser, vil samtlige forespørgsler blive "glemt". Derfor skal applikationen eventuelt også registrere disse "over-rule", hvis brugeren efter at have rettet problemerne, gensender dele af requestet.

Bedre hjælp til brugeren

Tidligere har FMK returneret Id i fejlmeddelelser, hvilket brugeren ofte ikke har kunnet bruge til noget. Fremadrettet, vil FMK i stedet indsætte en mere sigende reference. F.eks. for lægemiddelordinationer og recepter, hvor lægemiddelnavnet indsættes i stedet. I det returnerede response, vil fortsat være en reference til Id'er, som kan anvendes af afsendersystemet i forbindelse med håndtering af fejl eller advarsler (udvidede valideringer).

Nedenfor er vist et eksempel, hvor der er indsat DrugName i teksten, men hvor lægemiddelordinationens Id findes i Key/Value listen.

```
...
    <soapenv:Fault>
      <faultcode>Server</faultcode>
      <detail>
        <medcom:FaultCode>10001</medcom:FaultCode>
        <medicinecard20120601:FaultText>Ej muligt at seponere lægemiddelordinationen på Panodil, da
der findes en åben recept</medicinecard20120601:FaultText>
        <medicinecard20120601:FaultDetails>
          <medicinecard20120601:KeyValueSet>
            <medicinecard20120601:Key>DrugMedicationIdentifier</medicinecard20120601:Key>
            <medicinecard20120601:Value>31237621536712</medicinecard20120601:Value>
          </medicinecard20120601:KeyValueSet>
          <medicinecard20120601:KeyValueSet>
            <medicinecard20120601:Key>DrugName</medicinecard20120601:Key>
            <medicinecard20120601:DrugValue>Panodil</medicinecard20120601:DrugValue>
          </medicinecard20120601:KeyValueSet>
          <medicinecard20120601:KeyValueSet>
            <medicinecard20120601:Key>ElementPath</medicinecard20120601:Key>
            <medicinecard20120601:Value>WithdrawDrugMedicationRequest.DrugMedication[1]</medicinecard20120601:Value>
          </medicinecard20120601:KeyValueSet>
        </medicinecard20120601:FaultDetails>
      </detail>
      <faultstring>Ej muligt at seponere lægemiddelordination Panodil da der findes en åben
recept</faultstring>
    </soapenv:Fault>
  ...
```