

KONSEPT FOR FELLES NASJONAL ARBEIDSFLATE FOR BÅDE API- OG GUI- KONSUMENTER AV OFFENTLIGE TJENESTER

Versjon 0.1 – Bjørn Dybvik Langfors (bdl@digdir.no)

INNHold

Innhold	1
Introduksjon	2
Viktige begreper	2
Innbokselement og instans	2
Handlinger	2
GUI-handlinger	2
API-handling	3
Tjenesteressurs	3
Sesjonstoken	3
Scenarier som påvirker innboks	4
Overordnet diagram over konsept	5
Sekvensbeskrivelser	6
Tjenesteeier-initiert dialog	7
Opprettelse av innbokselement – sekvensdiagram	7
Opprettelse av innbokselement – tekstlig beskrivelse av trinn	8
Konsum gjennom GUI (portal) - sekvensdiagram	10
Konsum gjennom GUI (portal) – tekstlig beskrivelse av trinn	11
Konsum gjennom API – sekvensdiagram	12
Konsum gjennom API – tekstlig beskrivelse av trinn	13
Sluttbruker-initiert dialog	14
Gjennom GUI (portal) – sekvensdiagram	14
Gjennom GUI (portal) – tekstlig beskrivelse av trinn	15
Gjennom API - sekvensdiagram	16
Gjennom API – tekstlig beskrivelse av trinn	17

Dette er under arbeid, og alt som står i dette dokumentet må anses som foreløpig og gjenstand for endring. Formålet med dokumentet er primært å skissere et konsept, ikke detaljere en løsning (selv om teksten til tider er forholdsvis detaljert). Det må i fellesskap jobbes videre med å avklare ambisjonsnivå, sikre avgrensinger og samspill med øvrige produkter i Digdir.

I dette notatet beskrives et konsept for hvordan tjenester, som i denne konteksten er begrenset til dialogtjenester og digital post, kan nyttiggjøre seg av fellesfunksjonalitet i Altinn, herunder innboks, sluttbrukers arkiv, autorisasjon, varsling og hendelser uten at det innebærer behov for å benytte seg av Altinns utviklingsmiljøer eller applikasjonskjøretidsmiljø. Alle interaksjoner mellom tjenesteeiere (TE) og Altinn foregår via API-er, og det legges opp til stor fleksibilitet i hvorvidt Altinn innboks må involveres og det legges ingen begrensninger på hvordan forretningslogikken eller ulike brukerflater hos TE realiseres.

VIKTIGE BEGREPER

INNBOKSELEMENT OG INSTANS

Tilstand hos Altinn begrenser seg til metadata for en gitt *instans* av en tjeneste. En instans kan referere et skjema, en melding, eller en lengre saksgang – semantikken er opp til tjenesteeier å bestemme – men en instans manifesterer seg i Altinn i form av et *innbokselement*, som er en logisk entitet synlig i både API og GUI i Altinn Innboks, tilsvarende det en ser i innboksen i Altinn i dag.

Innbokselementet reflekterer tilstanden til en eller annen pågående eller avsluttet dialog eller melding fra en etat, og inneholder utelukkende metadata, f.eks. hvilken innboks den tilhører, adresse (URL), overskrift, dato, status samt en liste over aktuelle *handlinger* som kan utføres på innbokselementet. Altinn holder ikke på forretningsdata, og knytter semantikk kun til enkelte typer handlinger (slett og arkiver). Andre handlinger kan vilkårlig defineres av tjenesteeier, og all interaksjon med selve tjenesteinstansen kan foregå i tjenesteeiers brukerflater.

En viktig forskjell mot dagens «correspondence» i Altinn, er at innbokselementene er *mutérbare* og kan også være *dynamiske*. Et innbokselement kan (men må ikke) konfigureres til å foreta et synkront kall til et endepunkt hos TE idet brukeren åpner elementet for å vise en dynamisk oppdatert «preview» (etter TES forgodtbefinnende) av innholdet i elementet, sammen med en oppdatert liste over aktuelle handlinger.

Tjenesteeier kan også når som helst oppdatere metadata og tilgjengelige handlinger på elementet. Disse endringene kan også føre til at det genereres *hendelser*, som autoriserte parter kan agere på, f.eks. at det sendes et varsel eller at et SBS foretar seg noe.

HANDLINGER

En *handling* (som i «action») beskriver en interaksjon som brukere kan gjøre med eller relatert til en tjenesteinstans. Eksempler på handlinger er «Åpne», «Arkiver», «Slett», «Start signering», «Betal», «Bekreft», «Les mer» etc. Listen over aktuelle handlinger er en del av den strukturerte beskrivelsen av en tjenesteinstans, og kan når som helst endres av tjenesteeier gjennom API.

En handling er enten en «GUI»-handling eller en «API»-handling.

GUI-HANDLINGER

GUI-handlinger gjøres synlige for brukeren i form av knapper, lenker eller lignende. Tjenesteeieren oppgir selv om en gitt handling er å regne som en primær-, sekundær eller tertiær-handling, noe som påvirker hvordan dette presenteres til brukeren. En primærhandling vil typisk presenteres som en fremhevet knapp («call to

Kommentert [JMF1]: @Langfors, Bjørn Dybvik Kan vi her også legge inn et forbehold: Intensjonen for dokumentet er å skissere hvordan et felles målbilde for hvordan tjenester....Det må i fellesskap jobbes videre med å avklare ambisjonsnivå, sikre avgrensinger og samspill med øvrige produktgrupper / proudkter i Digdir.

Kommentert [LBD2R1]: Forsøkte meg et avsnitt.

Kommentert [ME3]: Fører dette til for tette/komplekse bindinger? Hvordan modellere dette for API? Bedre å kun vise metadata/actions, og heller navigere til innbokselement for å få preview etc? Actions kan vel også vises "inline" før åpnet?

Kommentert [LBD4R3]: Tanken var at dette var valgfritt avhengig av hvordan TE ønsket å implementere. Enten kan man ha en slik "live rendered preview"-funksjon tilgjengelig via eget API, som da on-demand kalles for hente siste versjon. Alternativt kan man bare pushe muteringer etter behov. Enkle tjenesten vil kanskje ikke trenge noen av delene, men de mer avanserte dialogene kan bruke begge mønstrene.

action»), og benyttes for det som er det logiske neste steget. En sekundærhandling (f.eks. «Avbryt») kan være en mer nedtonet knapp, eller tekstlenker, mens en tertiærhandling (f.eks. «Les mer om denne tjenesten») kan gjemmes bak en nedtrekksmeny eller lignende.

Alle GUI-handlinger har en URL. Disse URLene blir kalt i det brukeren aktiverer den aktuelle handlingen. Typisk innebærer dette at brukeren blir omdirigert til etatens egen brukerflate hvor den aktuelle handlingen da utføres, enten automatisk eller gjennom videre brukerinteraksjon. Andre handlinger kan markeres at de skal utføres gjennom bakkanal-kall. Brukeren blir da ikke omdirigert, men Altinn Innboks vil da foreta en forespørsel på vegne av brukeren til den oppgitte URL-en. Tjenesteeieren returnerer da det oppdaterte elementet, som umiddelbart blir vist brukeren igjen. Ved feil (enten av tekniske eller forretningslogiske årsaker) kan en feilmelding vises.

Det defineres to standard-handlinger; slett og arkivering. Disse fungerer som andre handlinger, men hvis merket som et bakkanal-kall vil disse innebære at elementet blir slettet / arkivert i Altinn Innboks. Altinn vil også knytte ekstra UI-logikk til disse handlingene (i form av å vise en «Er du sikker?» dialog i forkant).

API-HANDLING

En API-handling er bare synlige gjennom SBS-er som benytter Altinn Innboks API (GUI-handlinger er ikke synlige over API). Disse inneholder også en identifikator som indikerer hva slags type handling det er snakk om, hvilken URL som må kalles for å utføre handlingen, hvilken HTTP-operasjon som skal benyttes (typisk GET eller POST), og en lenke til en strukturert beskrivelse (JSON Schema eller lignende) av datamodellen som enten returneres eller forventes som input.

Altinn foretar ikke validering av noe data, og ser ikke hvilke data som flyter mellom SBS-et og tjenesteeierens API.

Alle handlinger – både GUI og API – har en identifikator som mappes til en *action* i *autorisasjonspolicyen* som er knyttet til en *tjenesteressurs*.

TJENESTERESSURS

Alle innbokselementer må referere en tjenesteressurs. En tjenesteressurs utgjør autorisasjonsbæreren, og kan sammenlignes med moderne bruk av lenketjenester i Altinn 2. Dette er en beskrivelse av en tjeneste som ligger i Altinn Resource Registry, en ny komponent i Altinn Autorisasjon. Hver tjenesteressurs har en autorisasjonspolicy uttrykt i XACML, som beskriver hvilke tilgangsregler som gjelder for alle innbokselementer som refererer den. XACML gir stor fleksibilitet i hvor grov- eller finkornet tilgangskontrollen skal være, og Altinn vil legge denne policyen til grunn for å bestemme hvem som kan se ett gitt innbokselement, og hvilke handlinger som skal være tilgjengelige. Eksempelvis vil GUI-handlingen «Signer» referere en *action* kalt «sign» i XACML-policyen, som krever tilganger den innloggende brukeren ikke besitter. Knappen vil derfor være grået ut og deaktivert. Tjenesteressursen er det tilgangsstyrere i virksomhetene forholder seg til, mht hvem som skal ha tilgang til å gjøre hva på vegne av en virksomhet (tilsvarende dagens tjenestedelegering).

På samme måte vil API-handlinger som ikke er tilgjengelige for SBS-et (med den identiteten SBS-et oppgir) ikke returneres. Det er imidlertid viktig at tjenesteeieren i sitt API foretar autorisasjonsoppslag mot Altinn for å forsikre seg om at et gitt kall er foretatt av en autorisert part.

Opplysninger om hvem som er den autoriserte parten overføres gjennom et *sesjonstoken*.

SESJONSTOKEN

Et sesjonstoken er enten et «opak» token (en tilfeldig tekststreng) eller en signert JWT som inneholder informasjon om den autentiserte brukeren/organisasjonen, hvilken aktør som er valgt, identifikator til innbokselementet og autorisasjonsressursen, dato og andre opplysninger. Ved bruk av et opak token må

tjenesteeieren foreta et oppslag mot Altinns API-er for å veksle det inn i informasjonen nevnt over. På denne måten kan sesjoner og autorisasjonsdata utover det som finnes i ID-porten/Maskinportet-tokens overføres mellom Altinn og den aktuelle etaten. Ved bruk av omdirigeringer i GUI-handlinger vil etaten også kunne lene seg på SSO fra ID-porten for å autentisere brukeren, og validere at informasjonen i sesjonstokenet stemmer overens.

SCENARIOER SOM PÅVIRKER INNBOKS

Det er typisk tre scenarioer som innebærer behov for interaksjon med innboks og innbokselementer.

1. **Sluttbruker-initiert dialog**, hvor sluttbruker (på vegne av seg selv eller annen part) finner og starter tjeneste ved hjelp av

- Offentlig tjenstekatalog
- Søkemotor
- Etatenes nettsider
- SBS (enten manuelt eller fordi SBS agerer på en hendelse)

Dette fører til at tjenesteeier oppretter et innbokselement i den aktuelle innboksen.

2. **Tjenesteeier-initiert dialog**, hvor tjenesteeier oppretter innbokselementet selv i den aktuelle innboksen. Dette er typisk:
 - «Prefill»-scenarioer, hvor tjenesteeieren trenger å innhente opplysninger og gir aktøren et delvis forhåndsutfylt skjema å begynne med
 - Proaktive/sammenhengende tjenester, hvor en etat igangsetter en dialog som følge av en hendelse (som kan ha oppstått i en annen tjeneste)

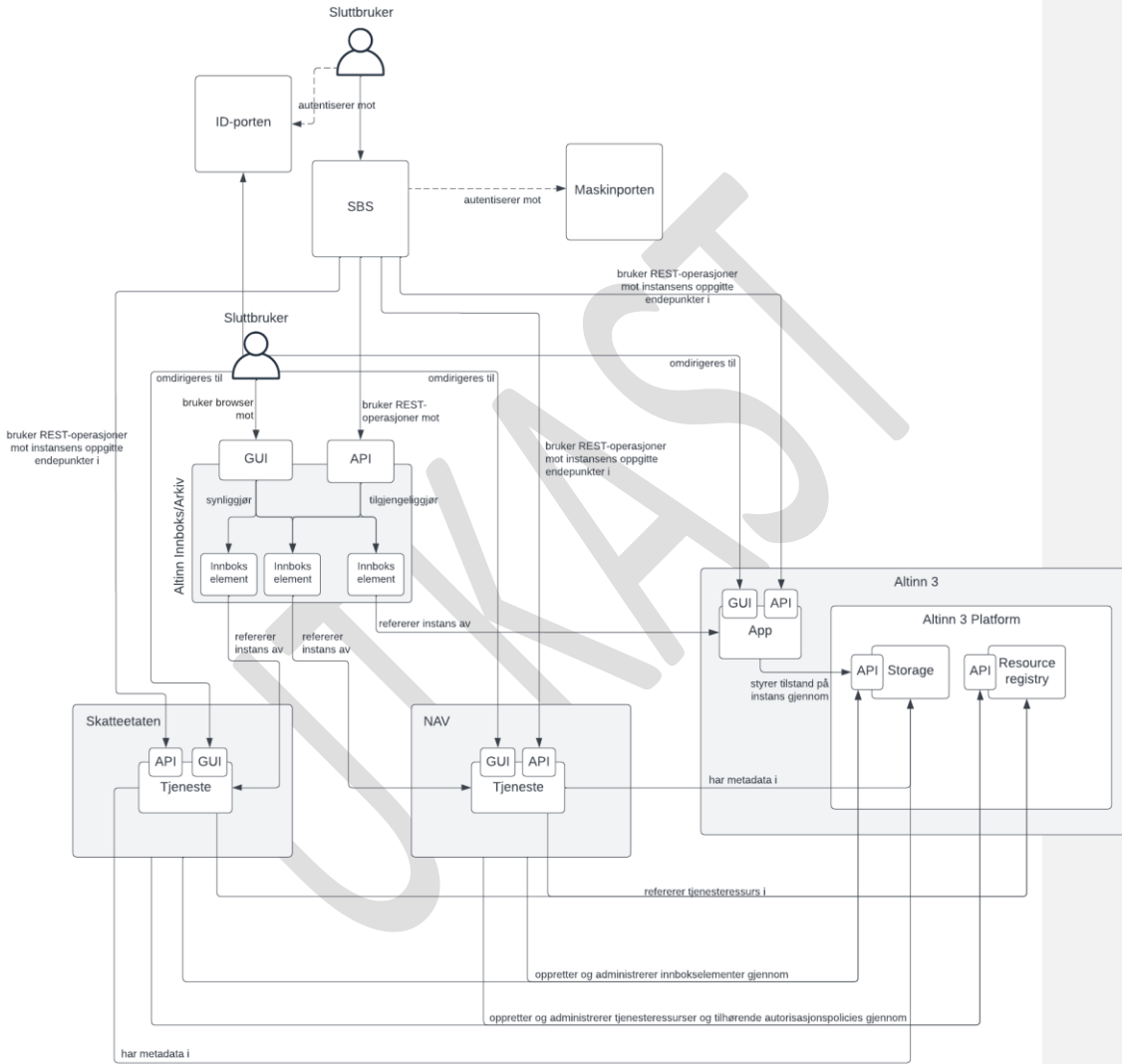
3. **Sending av digital post**

- DPV, DPI, DPO
- Typisk én-veis (foruten «rekommantert»-funksjonalitet hvor tjenesteeier trenger bekreftelse på at melding er mottatt og lest)
- Kan være utgangspunkt for sluttbruker-initiert dialog.
- Teknisk/funksjonelt subset av tjenesteeier-initiert dialog.

Det finnes andre scenarioer rundt oppslag/innsynstjenester og filoverføringer som trolig ikke vil behøve en representasjon i en innboks, og er følgelig out-of-scope for dette arbeidet.

OVERORDNET DIAGRAM OVER KONSEPT

[Lenke til større diagram](#)



SEKVENSBESKRIVELSER

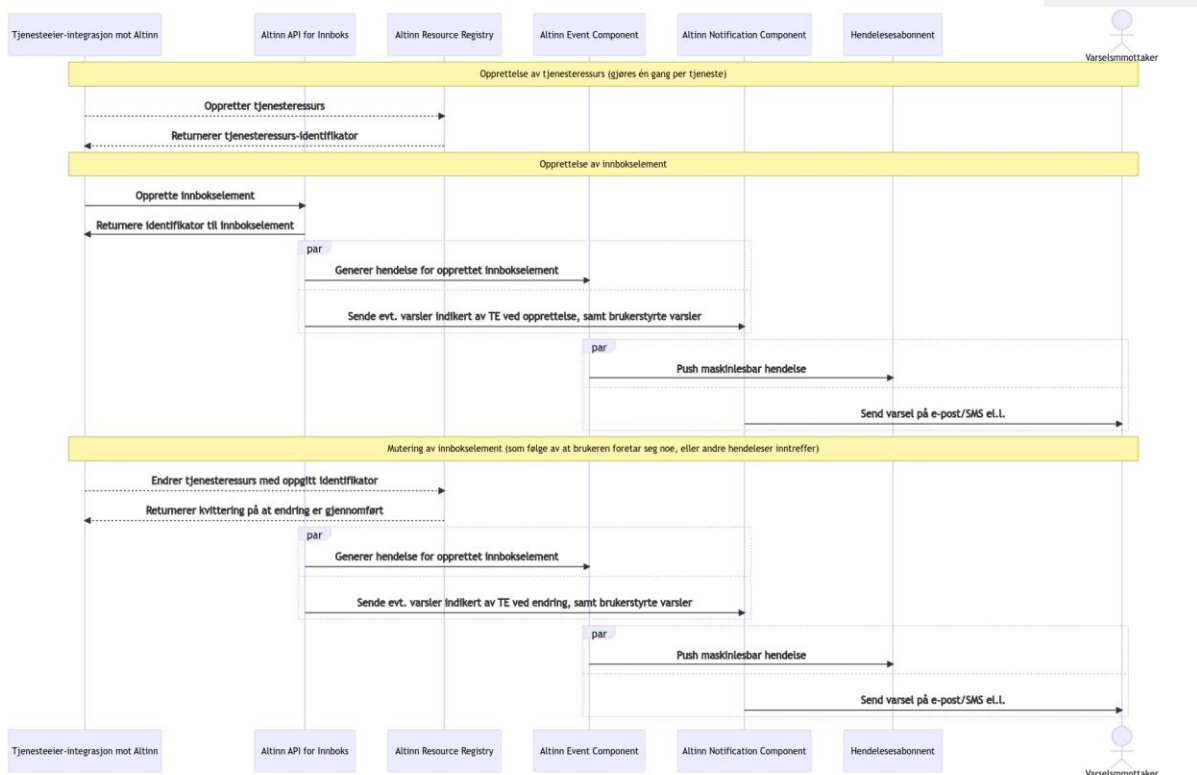
Under følger punktvis beskrivelser av fire ulike sekvenser. Det er to måter en dialog kan initieres på; enten av sluttbruker eller tjenesteeier. Hver av disse kan gjennomføres ved hjelp av GUI (Altinn Innboks + tjenesteeierens GUI-brukerflate), eller API (Altinn Innboks API + endepunkter som tjenesteeier eksponerer i sin API-brukerflate).

UTKAST

TJENESTEEIER-INITIERT DIALOG

OPPRETTELSE AV INNBOKSELEMENT – SEKVENSDIAGRAM

[Lenke til større bilde](#)



OPPRETTELSE AV INNBOKSELEMENT – TEKSTLIG BESKRIVELSE AV TRINN

1. Tjenesteieier oppretter tjenesteressurs i Altinn Resource Registry som refererer den aktuelle tjenesten
2. Tjenesteieier oppretter innbokselement som

- referer tjenesteressurs
- Inneholder tekstlig metadata (tittel, ingress etc) i flere språk
- Inneholder andre (valgfrie) metadata som f.eks.
 - Frist
 - "Fra"-felt
 - Dato for når elementet skal aktiveres
 - TE sin egen referanse/ID
- Inneholder (valgfri) URI for å hente oppdatering/strukturert versjon av innbokselementet, hvis hensiktsmessig for tjenesten. Brukes typisk i det brukeren ekspanderer innbokselementet for å vise
 - Dynamisk rikt innhold hentet i sanntid
 - Hvilke handlinger som kan utføres på elementet (se under)

Hvis ikke oppgitt, vises de metadata/handlinger jf. siste oppdatering som ble gjort av tjenesteieier (hvis noen, etter opprettelse)

- Beskriver tilgjengelige handlinger (for både API og GUI)
 - Strukturert liste over handlinger som kan gjøres på elementet, typisk "Åpne", "Arkiver", "Slett", "Bekreft", "Signer", "Betal", "Les mer", etc. GUI-handlinger kan flagges som primær, sekundær og tertiær-handlinger som påvirker hvordan dette vises til bruker (f.eks. kan primær og sekundærhandlinger vises som ulike typer knapper, mens tertiærhandlinger vises i en nedtrekksliste/tekstlenker).
 - Er gjenstand for autorisasjon definert av referert tjenesteressurs. F.eks. vil f.eks. "Signer" kunne kreve andre rettigheter avhengig av policy knyttet til tjenesteressursen.
 - Hver GUI-handling inneholder
 - En identifikator for handlingen. "Standardstandard"-handling vil kunne oppgis som allerede finnes oversatt i Altinn.
 - Hvis ikke standard-handling, tekst som beskriver handlingen i flere språk
 - En valgfri hjelpetekst som kan fremvises brukeren som gir mer informasjon om hva handlingen innebærer
 - Flaggsom indikerer om handlingen skal utføres i bakkanal
 - En URI som enten (1) brukeren vil omdirigert til når hen aktiverer det aktuelle GUI-elementet (f.eks. en knapp) eller (2) Altinn vil kalle hvis på vegne av brukeren hvis flagget som bakkanal-kall
 - Hvis bakkanal-kall, skal URI-en skal returnerer en standardmodell som indikerer om kallet var vellykket, eller om det skal vises en feilmelding.
 - Flaggsom indikerer om elementet skal slettes/arkiveres i Altinn hvis kall til URI lykkes (vil brukes til f.eks. "Er du sikker"-prompts)
 - Hver API-handling inneholder
 - En identifikator for handlingen
 - Hvilken http-metode som skal benyttes
 - En URI som handlingen skal utføres mot
 - JSON Schema el.l. som beskriver datamodellen som skal pushes/pulles

Kommentert [ME5]: Er det fornuftig at innboks skal pulle ved hver eneste ekspandering? Hvordan modellere "ekspandering" for API?

Kommentert [LBD6R5]: Tanken var at det bare skulle gjøres for de som har oppgitt et endepunkt for å hente oppdatering av elementet. Det kan i prinsippet være samme modellen som TE eller pusher inn ved oppdatering.

Kommentert [ME7]: Vil variere avhengig av stratus til elementet, og vil vel også typisk kreve at man har navigert til elementet (for både API og GUI)? Hvis ikke blir dette fort ekstremt komplisert.

Kommentert [LBD8R7]: Ja, listen med actions tenker jeg ikke er synlig i liste-visninger (verken api eller gui). Hva som er status på elementet utover "arkivert" og "slettet" er vel egentlig ikke noe vi trenger å legge oss så veldig borti? Ei heller hvilke actions som skal være tilgjengelige til enhver tid. Kanskje en action skal være "Send inn klage" på et arkivert element?

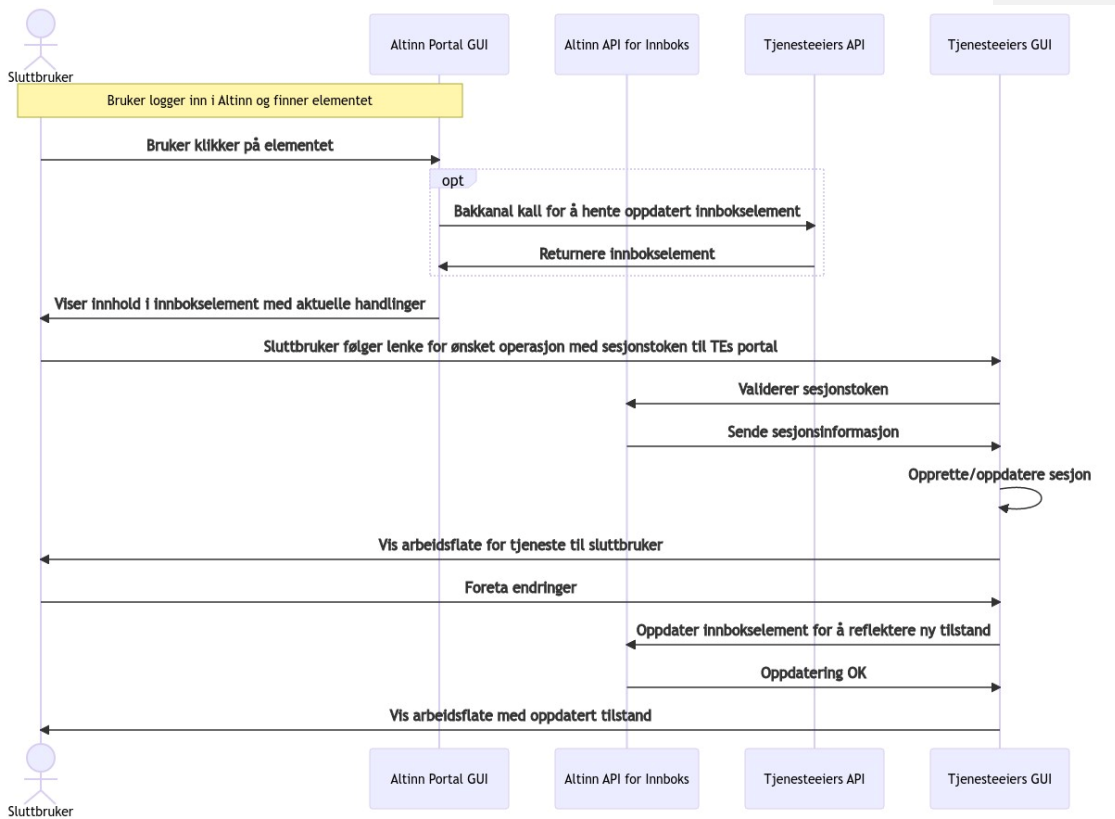
- Inneholder informasjon om varsling på SMS/e-post/push etc
- Inneholder valgfri liste over filvedlegg, som består av
 - En tittel på flere språk
 - En MIME-filtype som brukes for å indikere til brukeren type vedlegg (PDF etc)
 - En URI som dyplinker til vedlegget
 - Størrelse på vedlegget i bytes
- Kan muteres etter opprettelse
 - Tilgjengelige handlinger kan oppdateres når som helst av TE (som alternativ/supplement til at kall gjøres når bruker laster elementet)
 - Tittel og annen tekstlig metadata
 - Status ("under utfylling", "klar til signering", "venter på andre", "venter på svar fra etat" etc)

Hvis vellykket, returneres en unik identifikator for innboks-elementet.

3. Når elementet er opprettet/endret vil det
- Genereres hendelser som vil kunne konsumeres av parten
 - Disse kan igjen være koblet til brukerstyrt varsling på e-post/SMS andre push-kanaler

KONSUM GJENNOM GUI (PORTAL) - SEKVENSDIAGRAM

[Lenke til større bilde](#)



KONSUM GJENNOM GUI (PORTAL) – TEKSTLIG BESKRIVELSE AV TRINN

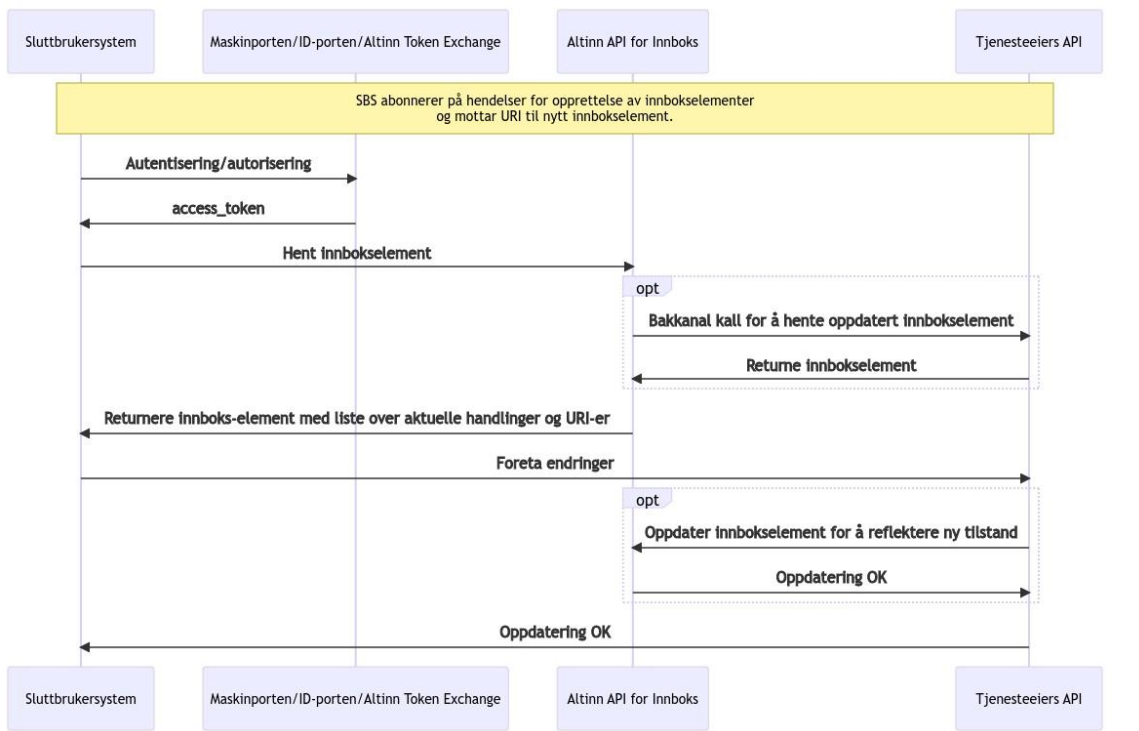
1. Bruker mottar varslings på en eller annen kanal, og logger inn i Altinn
2. Elementet ligger i innboksen og viser overskrift, status og andre metadata
3. Bruker klikker på elementet for å ekspandere det. Hvis det av TE ble oppgitt en URI for å oppdatere elementet, vil Altinn kalle denne i en bakkanal og vise innholdet dynamisk samt oppdatere evt endrede metadata. Ekspandert element viser rikt innhold som TE har definert, sammen med tilgjengelige handlinger. Hvis oppdatering feilet, vises enten feilmelding som TE oppga, eller en standardfeilmelding.
4. Bruker klikker på den definerte primærhandlingen.
 - a. Altinn vil da redirecte brukeren (nettleseren) til oppgitt URI. Det legges på en opaque sesjonstoken som parameter i URI-en (en ikke-forutsigbar tilfeldig tekststreng).
 - b. Når tjenesteier mottar forespørsel fra nettleser, gjøres et bakkanal-oppslag mot Altinn TE-API hvor sesjonstoken oppgis. Returnerer en modell som forteller
 - i. autentisert part (f/dnr, orgnr)
 - ii. valgt aktør
 - iii. Tidspunkt
 - iv. identifikator for elementet som ble klikket
 - v. TEs referanse til elementet
 - vi. identifikator for valgt handling

Alternativ til opak streng kan det brukes et token i form av en JWT som inneholder den samme informasjonen signert av Digdir. Dette unngår bruken av bakkanal, og reduserer behovet for tilstandshåndtering.

5. Ved hjelp av tokenet og SSO i ID-porten blir brukeren umiddelbart logget inn hos TE og tatt inn til tjenesteinstansen, hvor brukeren interagerer med tjenesten. Etter hvert som dialogen skrider frem, kan TE gjøre bakkanal-kall til Altinn for å oppdatere innbokselementet slik det fremstår for brukeren.
6. Hvis brukeren fullfører dialogen, kan TE gjøre et bakkanal-kall for å indikere til Altinn at innbokselementet skal arkiveres. Elementet blir da flyttet til sluttbrukers arkiv. Merk at det fremdeles kun ligger da (ikke lenger muterbare) metadata på elementet i Altinn.
7. Når brukeren senere ekspanderer elementet i arkivet, gjøres det samme kallet for å hente siste oppdaterte (altså arkiverte) element fra TE. Typisk vises da bare en kort tekst og et vedlegg til en PDF-versjon av en kvittering/gjenpart el.l.

KONSUM GJENNOM API – SEKVENSDIAGRAM

[Lenke til større bilde](#)



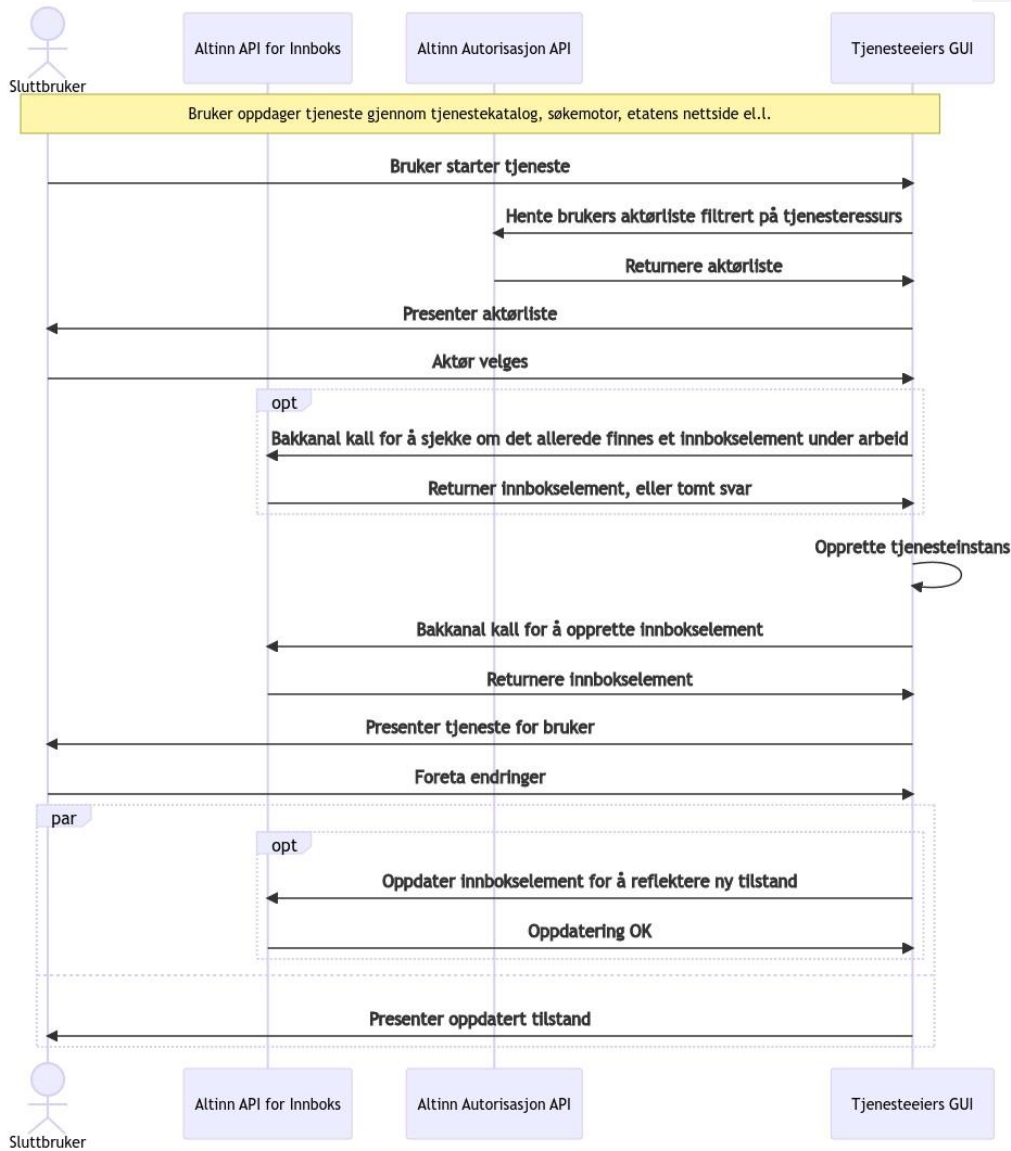
KONSUM GJENNOM API – TEKSTLIG BESKRIVELSE AV TRINN

1. SBS abonnerer på hendelser knyttet til opprettelse av innbokselementer av en eller flere typer, og mottar en notifikasjon om at et nytt innbokselement er opprettet. Notifikasjonen inneholder en URI til innbokselementet i Altinns API. Alternativt kan liste med innbokselementer hentes gjennom standard innboks-API-er
2. Avhengig av autorisasjonspolicy tilhørende tjenesterressursen, autoriserer SBS-et seg. Dette kan være gjennom brukerstyrt datadeling i ID-porten (OAuth2/OIDC) eller ved hjelp av Maskinporten (ren OAuth2). Tokenet kan også inneholde scopes som kreves av etaten, og tokenet bør da utstedes med både altinn og TE som i "aud"-claim. Hvis Maskinporten-token, kan policyen ha rettighetskrav som krever at det i tillegg autentiseres en virksomhetsbruker (systemidentitet i Altinn med tildelte rettigheter som gir tilgang til tjenesterressursen), som innbærer en tokenutveksling og utstedelse av et beriket Maskinporten-token. Dette resulterer i ett (eller to, hvis også beriket Altinn-token) access token som brukes i alle påfølgende kall.
3. Ved uthenting av elementet som ble referert av hendelsen, returneres en strukturert modell som langt på vei speiler modellen som TE opprinnelig sendte inn. Listen over handlinger definerer da hva SBS-et kan foreta seg, og hvilke endepunkter som må aksesseres for å utføre handlingene. Enkelte handlinger kan være synlige/gyldige kun for portal, eller kun for API. Handlinger kun synlige for API kan også referere JSON schemas el.l. som beskriver datamodellen som forventes på det aktuelle endepunktet. Tilsvarende håndtering av GUI-handlinger legges det ved et sesjonstoken som inneholder informasjon om tidspunkt, autentisert part, valgt avgiver, aktuelt element, valgt handling.
4. SBS-et interagerer med TEs API gjennom endepunktene som elementet beskriver, og/eller i tråd med swagger/annen dokumentasjon som etat har tilgjengeliggjort f.eks. via API-katalogen. Access token som utstedt av Altinn og/eller Maskinporten benyttes. Etter hvert som dialogen skrider frem, kan TE gjøre bakkanal-kall til Altinn for å oppdatere innbokselementet slik det fremstår for brukeren både i portal og API.

SLUTTBRUKER-INITIERT DIALOG

GJENNOM GUI (PORTAL) – SEKVENSDIAGRAM

[Lenke til større bilde](#)



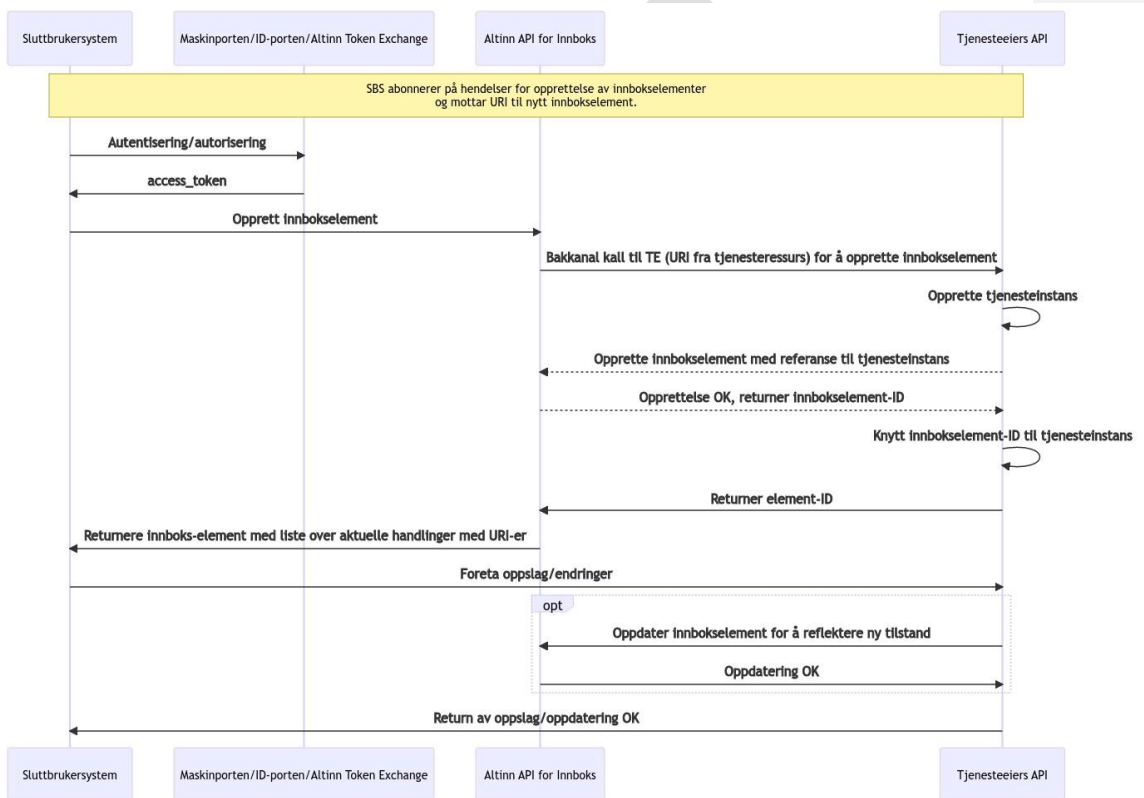
GJENNOM GUI (PORTAL) – TEKSTLIG BESKRIVELSE AV TRINN

1. Bruker oppdager tjeneste gjennom tjenestekatalog, søkemotor, etatenes nettsider el.l.
2. Bruker starter tjenesten og blir umiddelbart tatt inn i brukerflaten hos etaten, og velger aktør avhengig av tjenestens natur (via autorisasjonskall mot Altinn Autorisasjon for å bygge aktørliste)
3. TE gjør bakkanal-kall for å opprette innboksinstans i Altinn
 - a. Her kan en se på muligheten for å la TE sjekke om det allerede finnes instanser av tjenesten hos samme aktør som inneholder en eller annen status ("under arbeid" eller lignende) og gi bruker mulighet til å fortsette å jobbe med den instansen, eller opprette en ny.
4. Sluttbruker interagerer med tjenesten, og TE gjør kall til Altinn for å oppdatere innboksinstansen.
5. Hvis bruker avslutter økten før dialogen er ferdigstilt, kan han/hun (eller andre autoriserte) fortsette å jobbe med dialogen gjennom å aksessere innbokselementet i Altinn. Prosessen blir da som "Tjenesteeier-initiert dialog / Konsum gjennom GUI (portal)" steg 2.

GJENNOM API - SEKVENSDIAGRAM

- Her er det to nærliggende alternativer - skal tjenester kunne "instansieres"
 - ... gjennom et felles "instansierings"-API i Altinn som gjør bakkanal-kall til TE og returnerer et innbokselement med liste over handlinger/endpointer?
 - ... direkte mot TEs API-er som da kan gjøre bakkanal-kall til Altinn for å oppgi innbokselementet?
- Begge deler bør kanskje kunne støttes? Førstnevnte gir en mer homogen løsning sett fra SBS-ene sin side; selv om all kommunikasjon går direkte mot TE etter instansiering, er innbokselementet et som vil kunne reflektere gjeldende tilstand/status og aktuelle handlinger. Det andre løsnings gir en løsere kobling til Altinn, men gjør at SBS-et i mindre grad kan forholde seg til en felles brukerflate.
- Under skisseres en løsning med felles instansierings-API

[Lenke til større bilde](#)



GJENNOM API – TEKSTLIG BESKRIVELSE AV TRINN

1. ("Design-time") SBS oppdager API gjennom API-katalog eller annen dokumentasjon, og foretar merkantil og teknisk onboarding (setter opp Maskinporten-klient med rette scopes, oppretter virksomhetsbruker etc)
2. SBS autoriserer seg (tilsvarende "Tjenesteeier-initiert dialog / Konsum gjennom API", trinn 2.
3. SBS gjør et kall til et standard API i Altinn Innboks ("createinstance" el.l) som oppgir aktør og tjenesteressurs
4. Altinn foretar autorisasjon basert på policy knyttet til tjenesteressurs, og hvis godkjent gjør et bakkanal kall til et instansierings-ende punkt som er definert på tjenesteressurs. Kallet inneholder en standard modell som beskriver autentisert part, valgt aktør, og tjenesteressurs.
5. Tjenesteeier oppretter instans (el.l) i egne systemer, og gjør bakkanal-kall tilbake til Altinn for å opprette innbokselement som beskrevet i "Tjenesteeier-initert dialog", trinn 2.
6. Tjenesteeier mottar fra Altinn identifikator til innbokselement, som da kan knyttes til egen tjenesteinstans
7. Tjenesteeier returnerer identifikator til innbokselement til Altinn
8. Altinn laster det nyopprettede innbokselementet, og returner dette til SBS
9. SBS interagerer med TEs API-er som beskrevet i "Tjenesteeier-initiert dialog / Konsum gjennom API", trinn 4