

# Implementasjonsguide HelseAPI

E-helse Arkitektur

Exported on Mar 05, 2019

## Table of Contents

<b>1</b>	<b>Introduksjon</b>	<b>4</b>
<b>2</b>	<b>Formål med dokumentet, målgruppe og avgrensning</b>	<b>5</b>
2.1	Formål	5
2.2	Målgruppe	5
2.3	Avgrensning	5
2.4	Definisjoner	5
<b>3</b>	<b>HelseAPI profiler</b>	<b>6</b>
<b>4</b>	<b>HelseAPI konformitetskrav</b>	<b>7</b>
<b>5</b>	<b>Innholdsfortegnelse</b>	Feil! Bokmerke er ikke definert.
<b>6</b>	<b>Felles definisjoner, tolkninger og krav</b>	<b>8</b>
6.1	Felles klinisk datasett - Dataelementer	8
6.2	Må støttes (mustSupport)	8
6.3	Referanser i HelseAPI profilene	8
6.4	Syntaks leseoperasjoner	9
6.5	Syntaks søkeoperasjoner	9
6.6	Syntaks for søk som er begrenset til en pasient	9
6.7	Søk på tvers av plattformer	10
6.8	Veiledning på begrensning av antall resultater ved søk	10
6.9	Håndtering av versjonskonflikter	10
6.10	Responshåndtering	11
<b>7</b>	<b>Profiler</b>	<b>12</b>
7.1	Profiler definert som en del av HelseAPI implementasjonsguiden	12
7.2	HelseAPI Address	12
7.2.1	Obligatoriske elementer og terminologi	12
7.2.2	Formelle detaljer om profilen og profilinnhold	13
7.3	HelseAPI Encounter	13
7.3.1	Obligatoriske elementer og terminologi	13
7.3.2	Formelle detaljer om profilen og profilinnhold	14
7.3.3	Obligatoriske og anbefalte RESTful-interaksjoner	14
7.4	HelseAPI Organization	15
7.4.1	Obligatoriske elementer og terminologi	15
7.4.2	Formelle detaljer om profilen og profilinnhold	15
7.4.3	Obligatoriske og anbefalte RESTful-interaksjoner	16
7.5	HelseAPI Patient	17
7.5.1	Obligatoriske elementer og terminologi	17
7.5.2	Formelle detaljer om profilen og profilinnhold	17
7.5.3	Obligatoriske og anbefalte RESTful-interaksjoner	18
7.6	HelseAPI Practitioner	19
7.6.1	Obligatoriske elementer og terminologi	19
7.6.2	Formelle detaljer om profilen og profilinnhold	19
7.6.3	Obligatoriske og anbefalte RESTful-interaksjoner	20
7.7	NO basis HumanName	21
7.7.1	Eksempler	21
7.7.2	Formelle detaljer om profilen og profilinnhold	21
<b>8</b>	<b>Utvidelser (extensions)</b>	<b>23</b>
8.1	Utvidelser (extensions) definert som en del av HelseAPI implemenentasjonsguiden	23
8.2	NO basis AddressOfficial	23
8.2.1	Formelle detaljer om utvidelsen og profilinnhold	23
8.3	NO basis MiddleName	23
8.3.1	Formelle detaljer om utvidelsen og profilinnhold	23
8.4	NO basis PropertyInformation	24
8.4.1	Formelle detaljer om utvidelsen og profilinnhold	24
8.5	NO basis UrbanDistrict	24
8.5.1	Formelle detaljer om utvidelsen og profilinnhold	24
8.6	patient-nationality	25

8.6.1	Formelle detaljer om utvidelsen og profilinnhold .....	25
<b>9</b>	<b>Terminologi</b> .....	Feil! Bokmerke er ikke definert.
<b>10</b>	<b>Søkeparametere og operasjoner</b> .....	Feil! Bokmerke er ikke definert.
<b>11</b>	<b>Konformitetskrav</b> .....	<b>26</b>
11.1	Konformitetskrav for HelseAPI tjenere.....	26
11.1.1	Oppførsel .....	26
11.1.2	Sikkerhet .....	26
11.1.3	Profilinteraksjon .....	27
11.1.4	Patient.....	27
11.1.5	Practitioner.....	27
11.1.6	Organization.....	28
11.2	Konformitetskrav for HelseAPI klienter .....	28
11.2.1	Oppførsel .....	29
11.2.2	Patient.....	29
11.2.3	Practitioner.....	29
11.2.4	Organization.....	29
<b>12</b>	<b>Generelle sikkerhetsvurderinger</b> .....	<b>31</b>
12.1	Juridisk / Avtale .....	31
12.2	Krav til risikovurderinger .....	31
12.3	Sikring av kommunikasjonen mot HelseAPI .....	31
12.4	Autentisering .....	31
12.5	Autorisasjon / tilgangskontroll .....	31
12.6	Logging / Audit .....	32
12.7	Responshåndtering ved tilgangsnekt.....	32
<b>13</b>	<b>Eksempler</b> .....	<b>34</b>
13.1	no-helseapi-Practitioner-example.json .....	34
13.2	no-helseapi-Practitioner-example.xml .....	35

# 1 Introduksjon

[FHIR](#) er en standard for utveksling av helsedata elektronisk og består primært av to komponenter:

1. Ressurser - en samling av informasjonsmodeller som definerer dataelementer, avhengigheter og relasjoner for de forretningsobjektene som er mest relevant i helsevesenet.
2. API - en samling av veldefinerte grensesnitt for samhandling mellom to applikasjoner.

I helsevesenet er datasettet for "forretningsobjekter" ikke universelt definert, men det er en ideell og pågående evolusjonær, konsensusbasert prosess for å standardisere på et kjernesett med felles forretningsobjekter, inkludert ting som "en pasient", "en prosedyre", "en observasjon", "en ordre", osv.

## 2 Formål med dokumentet, målgruppe og avgrensning

### 2.1 Formål

Formålet med dokumentet er å tydeliggjøre premisser for teknisk tilrettelegging av EPJ for FHIR.

### 2.2 Målgruppe

Målgruppen for dokumentet er tekniske ressurser, testpersonell og prosjektledelse med ansvar for å tilrettelegge EPJ for FHIR.

### 2.3 Avgrensning

Dokumentet er avgrenset til de tjenestene og behovene som skal understøtte en implementasjon av FHIR i EPJ.

Dokumentet omfatter ikke:

- etablering av driftsrutiner
- etablering av overvåkningsrutiner

### 2.4 Definisjoner

- Helsepersonell – en person autorisert som helsepersonell i henhold til helsepersonelloven § 48. Rettighetene til ulike autorisasjoner er forskjellig, f.eks. kan kun leger verifisere kritisk informasjon.
- Innbygger – en person som bor innen et spesifikt område. Et område her kan for eksempel være et land, fylke, kommune, by og sted.
- Pasient – en person som får behandling av helsepersonell for psykisk og/eller fysisk sykdom/lidelse eller skade, på sykehus/klinikk og/eller andre behandlingsinstitusjoner.
- EPJ – Elektronisk pasientjournal
- EHR – Electronic Health Record
- FHIR – Fast Healthcare Interoperability Resources
- API – Application Programming Interface
- REST – Representational State Transfer

### 3 HelseAPI profiler

- [HelseAPI Address](#)
- [HelseAPI Encounter](#)
- [HelseAPI Organization](#)
- [HelseAPI Patient](#)
- [HelseAPI Practitioner](#)
- [NO basis HumanName](#)

## 4 HelseAPI konformitetskrav

Seksjonen [Konformitetskrav](#) skisser de krav HelseAPI tjenere og klienter skal følge, beskrivelse av de spesifikke profilene, RESTful-interaksjoner og søkeparametere som må støttes. Implementatører må benytte seg av FHIR-kallet [/metadata](#) for detaljer om RESTful-operasjonene, bestemte profiler og søkeparametere som gjelder for hver av de enkelte norske aktørene. De individuelle HelseAPI profilene setter krav til strukturelle begrensninger og terminologibindinger.

## 5 Felles definisjoner, tolkninger og krav

Denne seksjonen beskriver essensielle definisjoner og tolkninger som brukes i HelseAPI IG.

Konformitetsverbene som benyttes i integrasjonsguiden er definert i [konformitetsreglene for FHIR](#). Følgende norske oversettelser er benyttet for verbene:

- SKAL (SHALL)
- SKAL IKKE (SHALL NOT)
- BØR / BØR IKKE (SHOULD / SHOULD NOT)
- KAN (MAY)

### 5.1 Felles klinisk datasett - Dataelementer

Nr	Dataelement	HelseAPI profil	FHIR ressurs
(1)	Pasientens personidentifikator	<a href="#">HelseAPI Patient</a>	<a href="#">Patient</a>
(2)	Pasientens navn	<a href="#">HelseAPI Patient</a>	<a href="#">Patient</a>
(3)	Kjønn	<a href="#">HelseAPI Patient</a>	<a href="#">Patient</a>
(4)	Fødselsdato	<a href="#">HelseAPI Patient</a>	<a href="#">Patient</a>

### 5.2 Må støttes (mustSupport)

I konteksten av HelseAPI SKAL *mustSupport* på ethvert dataelement tolkes som følger:

- HelseAPI tjenere SKAL inneha kapabilitet til å inkludere dataelementet som en del av forespørselen som spesifisert av HelseAPI tjener kapabilitetserklæring.
- HelseAPI klienter SKAL inneha kapabilitet til å prosessere ressursinstanser som inneholder dataelementet uten å generere en feilmelding eller forårsake en applikasjonsfeil. Med andre ord HelseAPI klienter BØR inneha kapabilitet for å kunne vise dataelementene for menneskelig bruk, lagring eller andre hensikter.
- I situasjoner der informasjon om et bestemt dataelement ikke er tilstede, og årsaken til fravær er ukjent, SKAL IKKE HelseAPI tjenere inkludere dataelementet som en del av responsen.
- Ved forespørsel mot HelseAPI tjenere SKAL HelseAPI klienter tolke manglende dataelementer for en ressursinstans som at data ikke er tilstede i HelseAPI tjenerens system.
- MERK: Lesere av disse dokumentene rådes til å ha en forståelse om [FHIR terminologi](#), [FHIR RESTful API](#), [FHIR datatyper](#), [FHIR søk](#) og [FHIR Resource](#) formatet før de implementer HelseAPI kravene.

### 5.3 Referanser i HelseAPI profilene

Mange av profilene i denne guiden har [referanser](#) til andre FHIR ressurser som også er HelseAPI profiler. Disse er definert i de formelle profildefinisjonene. For referanser der det ikke formelt er definert en HelseAPI profil BURDE den refererte ressursen være en HelseAPI profil dersom denne eksisterer.



## 5.4 Syntaks leseoperasjoner

Interaksjoner er definert med følgende syntaks:

```
GET [base]/[Resource-type]/[id] {parameters}
```

- HTTP-verbet GET brukes ved lesing av en ressurs
- Innhold omkapslet med [] er påkrevet og erstattes med den nødvendige strengen
  - base: tjenestens rot-URL (f.eks: "<https://fhir-open-api-r4.smarthealthit.org>")
  - Resource-Type: typenavnet til ressursen (f.eks: "Patient")
  - id: den logiske identifikatoren til ressursen (f.eks: "504202")
- Innhold omkapslet med {} er valgfritt
  - parameters: URL-parametere som definert for den spesifikke interaksjonen (f.eks: "?\_format=xml")

For mer informasjon se [FHIR RESTful API](#).

## 5.5 Syntaks søkeoperasjoner

I den simpleste form utføres et søk ved en GET-operasjon i RESTful-rammeverket:

```
GET [base]/[Resource-type]?name=value&...
```

I det ovenstående eksemplet på et RESTful-søk er parameterne et sekvenspar med name=[value] kodet i URL-en. Søkeparameterne er definert pr. ressurs. Se [FHIR søk](#) for mer informasjon vedrørende søk.

## 5.6 Syntaks for søk som er begrenset til en pasient

Det er flere potensielle muligheter å søke etter ressurser som er assosiert med en spesifikk pasient avhengig av kontekst og implementasjon. Følgende søk vil ende opp med det samme resultatet:

1. En eksplisitt angitt pasient ved bruk av "patient" parameteren som begrenser resultatet innenfor den ressurstypen det søkes på til den angitte pasienten.
  - **GET [base]/[Resource-type]?patient=504202{&otherparameters}**
  - Det er flere variasjoner av denne syntaksen:
    - GET [base]/[Resource-type]?subject=[id]{&otherparameters}
    - GET [base]/[Resource-type]?subject:Patient.identifier=[system]][code]
    - GET [base]/[Resource-type]?subject=Patient/[id]{&otherparameters}
    - GET [base]/[Resource-type]?subject.\_id=[id]{&otherparameters}
    - GET [base]/[Resource-type]?subject:Patient=[id]{&otherparameters}
    - GET [base]/[Resource-type]?subject:Patient=Patient/[id]{&otherparameters}

- GET [base]/[Resource-type]?subject:Patient=[https://%5Burl%5D/Patient/id]{&otherparameters}
  - GET [base]/[Resource-type]?subject:Patient.\_id=[id]{&otherparameters}
  - GET [base]/[Resource-type]?patient:Patient=[https://%5Burl%5D/Patient/id]{&otherparameters}
  - GET [base]/[Resource-type]?patient:Patient.identifier=[system][code]
2. Pasienten KAN være en implisitt del av konteksten, slik som ved bruk av SMART. I disse tilfellene kan pasient-parameteren utelates:
- GET [base]/[Resource-type]{?otherparameters}

## 5.7 Søk på tvers av plattformer

Det er ikke et krav om at HelseAPI tjenere skal støtte eksterne URL-er som går på tvers av deres systemmiljø(er).

## 5.8 Veiledning på begrensning av antall resultater ved søk

For å kunne begrense antall søkeresultater som returneres i hvert enkelt kall, kan serveren velge å returnere resultatene som en serie med sider. Søkeresultatet inneholder ett sett med URL-er som klienten kan benytte når den gjør en forespørsel om flere sider fra søkeresultatet, for mer informasjon se [FHIR paginering](#).

## 5.9 Håndtering av versjonskonflikter

[Tapte oppdateringer](#) der to klienter oppdaterer den samme ressursen, og nummer to skriver over endringene til nummer én kan unngås ved å bruke en kombinasjon av headerene ETag og If-Match.

For å støtte oppunder håndtering av versjonskonflikter SKAL servere alltid returnere en ETag:

```
HTTP 200 OK
Date: Sat, 09 Feb 2013 16:09:50 GMT
Last-Modified: Sat, 02 Feb 2013 12:02:47 GMT
ETag: W/"23"
Content-Type: application/fhir+json
```

ETag SKAL være identisk med versjonsidentifikatoren for ressursen på serveren. (Burde vi si noe om formatet? Spesifikasjonen sier: *Servers are allowed to generate the version id in whatever fashion that they wish, so long as they are valid according to the id data type, and are unique within the address space of all versions of the same resource.*)

Ved oppdatering av en ressurs SKAL klienten angi headeren If-Match som siterer ETag headeren som ble returnert av serveren:

```
PUT /Patient/347 HTTP/1.1
If-Match: W/"23"
```

Ved oppdateringer SKAL Serveren kreve at klienten angir headeren If-Match, dersom headeren mangler SKAL serveren returnere HTTP statuskoden 412 Pre-condition failed.

For mer informasjon, se [Managing Resource Contention](#).

## 5.10 Responshåndtering

Både ved bruk av REST og FHIR med REST er det viktig at svar fra en tjener følger standardisering av HTTP status koder, for anbefalinger vedrørende responshåndtering ved tilgangsnøkk se [Sikkerhet](#).

HTTP standarden definerer følgende status kodeserier:

- 1xx-serien er reservert for lavnivå HTTP-ting.
- 2xx-serien er reservert for vellykkede kall der alt går som planlagt.
- 3xx-serien er reservert for omadressering av trafikk.
- 4xx-serien er reservert for å svare på feil gjort av klienten, f.eks. feil inputdata eller at de ber om ting som ikke eksisterer. Disse forespørslene skal være idempotente, og skal ikke endre tjenerens tilstand.
- 5xx-serien er reservert for svar når tjeneren gjør en feil. Ofte blir disse feilene returnert av applikasjonene og normalt utenfor utviklerens rekkevidde. Dette er for å sikre at klientene faktisk mottar et svar. Klienten kan ikke muligens vite tilstanden til serveren når et 5xx-svar mottas, og derfor bør disse unngås.

## 6 Profiler

### 6.1 Profiler definert som en del av HelseAPI implementasjonsguiden

Følgende profiler definerer ett sett med minimum forventninger for å registrere, søke og hente ut helsedata assosiert med en pasient:

- [HelseAPI Address](#)
- [HelseAPI Encounter](#)
- [HelseAPI Organization](#)
- [HelseAPI Patient](#)
- [HelseAPI Practitioner](#)
- [NO basis HumanName](#)

### 6.2 HelseAPI Address

Denne profilerte identifiserer ett sett med minimum forventninger til datatypen [Address](#), ved registrering, søk og uthenting av en enkelt adresse. Den definerer hvilke kjerneelementer, utvidelser, vokabularer og kodeverk som SKAL være tilstede når man benytter denne profilen.

#### 6.2.1 Obligatoriske elementer og terminologi

Følgende dataelementer er obligatoriske.

**Hver enkelt Address må ha:**

1. Postnummer
2. Poststed

I tillegg følgende elementer "må støttes" ([Must Support](#)).

**Dersom data er tilgjengelig skal Address inkludere:**

1. Adresselinje
2. Kommune
3. Fylke
4. Land
5. Bruksområde for adresse, angir om dette er en hjemme-, arbeidsadresse, midlertidig eller gammel adresse
6. Type adresse, angir om dette er en postadresse, fysisk adresse eller begge deler
7. Offisiell, angir om dette er en adresse registrert i folkeregisteret for personer eller enhetsregisteret for organisasjoner

**Profilspesifikk implementeringsveiledning**

- Ingen

#### 6.2.1.1 Eksempler

- *Legg inn link til eksempel*

## 6.2.2 Formelle detaljer om profilen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-helseapi-Address>

Publisert den (*en-eller-annen-dato*) (ev. *som et utkast*) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på den norske basisprofilen [no-basis-Address](#).

Tekstlig sammendrag

### Sammendrag av obligatoriske krav

1. Ett postnummer i address.postalCode.
2. Ett poststed i address.city.

### Sammendrag av "må støttes" krav

1. Én eller flere adresselinjer i Address.line.
2. Én kommune i Address.district.
3. Ett fylke i Address.state.
4. Ett land i Address.country.
5. Ett bruksområde for adresse i Address.use fra kodeverket [AddressUse](#).
6. Én type adresse i Address.type fra kodeverket [AddressType](#).
7. Én offisiell i Address.extension = NO basis Official

## 6.3 HelseAPI Encounter

**NB: Denne ressursen har ikke vært gjennom en formell profileringsprosess, og skal derfor ikke implementeres. Følgende profil er et diskusjonsgrunnlag for en fremtidig profil på Encounter.**

Denne profilen identifiserer ett sett med minimum forventninger til en [Encounter](#) ressurs, ved registrering, søk og hente grunnleggende møteinformasjon for en enkelt pasient. Den definerer hvilke kjerneelementer, utvidelser, vokabularer og kodeverk som SKAL være tilstede når man benytter denne profilen.

Eksempelscenarioer:

- Forespørsel etter en bestemt pasientsammenkomst
- Forespørsel om nylige pasientsammenkomster
- Registrere eller oppdatere møteinformasjon

### 6.3.1 Obligatoriske elementer og terminologi

Følgende dataelementer er obligatoriske.

**Hver enkelt Encounter må ha:**

1. Status
2. Type møte
3. Referanse til en pasient

I tillegg, følgende dataelementer "må støttes" ([Must Support](#))

Dersom data er tilgjengelig skal Encounter inkludere:

1. Identifikator
2. Deltager
3. Periode
4. Årsakskode
5. Diagnose
6. Utskrivningsårsak
7. Lokasjon

#### Profilspecifikk implementeringsveiledning

- Ingen

##### 6.3.1.1 Eksempler

- *Legg inn link til eksempel*

### 6.3.2 Formelle detaljer om profilen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-helseapi-Encounter>

Publisert den *(en-eller-annen-dato)* (ev. som et utkast) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på [Encounter](#).

#### Tekstlig sammendrag

##### Sammendrag av obligatoriske krav

1. Én status i Encounter.status fra kodeverket [EncounterStatus](#).
2. Én type møte i Encounter.type fra kodeverket (*trenger norsk kodeverk som ev. referer til relevant kodeverk, se [US Core EncounterType](#)*).
3. Én referanse til en pasient i Encounter.subject.

##### Sammendrag av "må støttes" krav

1. Én eller flere identifikatorer i Encounter.identifier.
2. Én eller flere personer som er involvert i møtet, som ikke er pasienten selv, i Encounter.participant.
3. Én periode som beskriver start- og sluttidspunkt for møtet i Encounter.period
4. Én eller flere årsakskoder i Encounter.reasonCode fra kodeverket [EncounterReason](#).
5. Én eller flere diagnoser i Encounter.diagnosis.condition.
6. En eller flere utskrivningsårsaker i Encounter.hospitalization.dischargeDisposition fra eksempelkodeverket [dischargeDisposition](#).
7. En eller flere lokasjoner i Encounter.location

### 6.3.3 Obligatoriske og anbefalte RESTful-interaksjoner

Nedenfor finner du en oversikt over de RESTful FHIR-interaksjonene som er påkrevet - for eksempel søk, og les operasjoner - for denne profilen. Se samsvarkrav for en komplett liste av støttede RESTful-interaksjoner.

GET [base]/Encounter/[id]

**Eksempel:** GET [base]/Encounter/12345

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Encounter.

-----  
 GET [base]/Encounter?patient=[id]

**Eksempel:** GET [base]/Encounter?patient=504202

*Støtte:* Obligatorisk å støtte referansesøk der pasient er kriterie

*Implementeringsnotater:* Søker etter alle Encounter for en pasient. Henter en bundle med alle Encounter ressurser for den spesifiserte pasienten. patient ([hvordan utføre et referansesøk](#)).

-----  
 GET [base]/Encounter?patient=[id]&period=[date]

**Eksempel:** GET [base]/Encounter?patient=504202&period=ge2019-01-01

*Støtte:* Obligatorisk å støtte søk med et datointervall

*Implementeringsnotater:* Henter en bundle med alle Encounter ressurser for den spesifiserte pasienten for en gitt tidsperiode. patient ([hvordan utføre et referansesøk](#)) og period ([hvordan utføre et datosøk](#)).

## 6.4 HelseAPI Organization

Denne profilen identifiserer ett sett med minimum forventninger til en [Organization](#) ressurs, søk og uthenting av en organisasjon assosiert med en pasient eller leverandør. Den definerer hvilke kjerneelementer, utvidelser, vokabularer og kodeverk som SKAL være tilstede når man benytter denne profilen.

### Eksempelscenarioer:

- Forespørsel om en organisasjon ved hjelp av navn, organisasjonsnummer eller RESH-ID.

### 6.4.1 Obligatoriske elementer og terminologi

Følgende dataelementer er obligatoriske.

#### Hver enkelt Organization må ha:

- En identifikator
- Et navn
- En adresse

#### Dersom data er tilgjengelig skal Organization inkludere:

- En status som indikerer om organisasjonen er aktiv
- Én eller flere kontaktpunkter

#### Profilsesifikk implementeringsveiledning

- Ingen

#### 6.4.1.1 Eksempler

- *Legg inn link til eksempel*

### 6.4.2 Formelle detaljer om profilen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-helseapi-Organization>

Publisert den (*en-eller-annen-dato*) (ev. som et utkast) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på den norske basisprofilen [no-basis-Organization](#).

## Tekstlig sammendrag

**Sammendrag av obligatoriske krav**

1. Én eller flere identifikatorer i Organization.identifier
  - Hver Organization.identifier må ha
    - En identifier.system der følgende identifikatorserier er lovlig:
      - 2.16.578.1.12.4.1.2.101 (Organisasjonsnummer)
      - 2.16.578.1.12.4.1.2.102 (RESH-ID)
    - En identifier.value som er unik innenfor systemet
2. Ett eller flere navn i Organization.name
3. Én eller flere adresser i Organization.address

**Sammendrag av "må støttes" krav**

1. En boolsk verdi i Organization.active, dersom den er utelatt skal verdien tolkes til true
2. Én eller flere kontaktpunkter i Organization.telecom

### 6.4.3 Obligatoriske og anbefalte RESTful-interaksjoner

Nedenfor finner du en oversikt over de RESTful FHIR-interaksjonene som er påkrevet - for eksempel søk, og les operasjoner - for denne profilen. Se samsvarkrav for en komplett liste av støttede RESTful-interaksjoner.

GET [base]/Organization/[id]

**Eksempel:** GET [base]/Organization/708302

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Organization.

-----

GET [base]/Organization?identifier=[system]][code]

**Eksempel:** GET [base]/Organization?identifier=urn:oid:2.16.578.1.12.4.1.2.101|915933149

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Organization ved hjelp av en identifikator som for eksempel organisasjonsnummer, identifier ([hvordan utføre et token-søk](#)).

-----

GET [base]/Organization?name=[name]

**Eksempel:** GET [base]/Organization?name=direktoratet%20for%20e-helse

*Støtte:* Obligatorisk

*Implementeringsnotater:* Søk basert på ett dataelement, name ([hvordan utføre et strengsøk](#)).

-----

GET [base]/Organization?name=[name]

**Eksempel:** GET [base]/Organization?name=verkstedveien

*Støtte:* Bør støtte Organization-søk på adresse

*Implementeringsnotater:* Søk basert på ett dataelement, address ([hvordan utføre et strengsøk](#)).



## 6.5 HelseAPI Patient

Denne profilen identifiserer ett sett med minimum forventninger til en [Patient](#) ressurs, ved registrering, søk og uthenting av demografi og annen administrativ informasjon om en enkelt pasient. Den definerer hvilke kjerneelementer, utvidelser, vokabularer og kodeverk som SKAL være tilstede når man benytter denne profilen.

### Eksempelscenarioer:

- Forespørsel om pasientens demografiske data ved hjelp av fødselsnummer (FNR). FNR er identifiserbar ved `identifier.system`.
- Forespørsel om pasientens demografiske data ved hjelp av etternavn og fødselsdato
- Referere fra andre ressurser

### 6.5.1 Obligatoriske elementer og terminologi

Følgende dataelementer er obligatoriske.

#### Hver enkelt Patient må ha:

1. Identifikator, eksempelvis fødselsnummer
2. Navn

I tillegg, følgende dataelementer "må støttes" ([Must Support](#))

#### Dersom data er tilgjengelig skal Patient inkludere:

1. Fødselsdato
2. Kjønn

#### Profilsesifikk implementeringsveiledning

- I henhold til [lov om personnavn](#) er følgende kombinasjoner av `HumanName.family` lovlig:
  - Hansen
  - Olsen-Hansen

#### 6.5.1.1 Eksempler

- *Legg inn link til eksempel*

### 6.5.2 Formelle detaljer om profilen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-helseapi-Patient>

Publisert den (*en-eller-annen-dato*) (ev. *som et utkast*) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på den norske basisprofilen [no-basis-Patient](#).

#### Tekstlig sammendrag

##### Sammendrag av obligatoriske krav

1. Én eller flere personidentifikatorer i `Patient.identifier`
  - Hver `Patient.identifier` må ha:
    - En `identifier.system` der følgende identifikatorserier er lovlig:
      - 2.16.578.1.12.4.1.4.1 (fødselsnummer)
      - 2.16.578.1.12.4.1.4.2 (d-nummer)
      - 2.16.578.1.12.4.1.4.3 (felles hjelpenummer)
    - En `identifier.value` som er unik innenfor systemet

## Tekstlig sammendrag

## 2. Ett eller flere navn i Patient.name

- Hver Patient.name må ha:
  - En name.family
  - En name.given

**Sammendrag av "må støttes" krav**

## 1. En fødselsdato i Patient.birthDate

## 2. Ett kjønn i Patient.gender

- Patient.gender er bundet til kodeverket [AdministrativeGender](#)

### 6.5.3 Obligatoriske og anbefalte RESTful-interaksjoner

Nedenfor finner du en oversikt over de RESTful FHIR-interaksjonene som er påkrevet - for eksempel søk, og les operasjoner - for denne profilen. Se samsvarkrav for en komplett liste av støttede RESTful-interaksjoner.

GET [base]/Patient/[id]

**Eksempel:** GET [base]/Patient/504202

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Patient.

POST [base]/Patient/\_search{identifiser=[system]][code]}

**Eksempel:**

POST [base]/Patient/\_search{identifiser=urn:oid:2.16.578.1.12.4.1.4.1|13116900216}

Content-Type: application/x-www-form-urlencoded

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Patient ved hjelp av en identifikator som for eksempel fødselsnummer, identifiser ([hvordan utføre et token-søk](#)). Søket utføres ved bruk av HTTP-verbet POST og parameterne url-kodes på body i henhold til [application/x-www-form-urlencoded](#)

GET [base]/Patient?identifiser=[system]][code]

**Eksempel:** GET [base]/Patient?identifiser=urn:oid:2.16.578.1.12.4.1.4.1|13116900216

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Patient ved hjelp av en identifikator som for eksempel fødselsnummer, identifiser ([hvordan utføre et token-søk](#)).

GET [base]/Patient?name=[string]&birthdate=[date]

**Eksempel:** GET [base]/Patient?name=Danser&birthdate=1969-11-13

*Støtte:* Obligatorisk

*Implementeringsnotater:* Søk basert på minimum 2 pasientelementer, name ([hvordan utføre et strengsøk](#)) og birthdate ([hvordan utføre et datosøk](#)).

GET [base]/Patient?name=[string]&gender=[code]

**Eksempel:** GET [base]/Patient?family=Danser&gender=female

*Støtte:* Obligatorisk

*Implementeringsnotater:* Søk basert på minimum 2 pasientelementer, name ([hvordan utføre et strengsøk](#)) og gender ([hvordan utføre et token-søk](#)).

-----

GET [base]/Patient?family=[string]&gender=[code]

**Eksempel:** GET [base]/Patient?family=Danser&gender=female

*Støtte:* Bør støtte Patient-søk på etternavn og kjønn

*Implementeringsnotater:* Søk basert på minimum 2 pasientelementer, family ([hvordan utføre et strengsøk](#)) og gender ([hvordan utføre et token-søk](#)).

-----

GET [base]/Patient?given=[string]&gender=[code]

**Eksempel:** GET [base]/Patient?given=Line&gender=female

*Støtte:* Bør støtte Patient-søk på fornavn og kjønn

*Implementeringsnotater:* Søk basert på minimum 2 pasientelementer, given ([hvordan utføre et strengsøk](#)) og gender ([hvordan utføre et token-søk](#)).

## 6.6 HelseAPI Practitioner

Denne profilen identifiserer ett sett med minimum forventninger til en Practitioner ressurs, ved registrering, søk og uthenting av demografi og annen administrativ informasjon om et enkelt helsepersonell. Den definerer hvilke kjerneelementer, utvidelser, vokabularer og kodeverk som SKAL være tilstede når man benytter denne profilen.

### Eksempelscenarioer:

- Forespørsel om ett helsepersonell ved navn eller HPR-nummer
- Referere fra andre ressurser

### 6.6.1 Obligatoriske elementer og terminologi

Følgende dataelementer er obligatoriske.

Hver enkelt helsepersonell må ha:

1. Identifikator, eksempelvis HPR-nummer
2. Navn

### Profilspesifikk implementeringsveiledning

- Ingen

#### 6.6.1.1 Eksempler

- [no-helseapi-Practitioner-example.json](#)
- [no-helseapi-Practitioner-example.xml](#)

### 6.6.2 Formelle detaljer om profilen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-core-Practitioner>

Publisert den (*en-eller-annen-dato*) (ev. *som et utkast*) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på den norske basisprofilen [no-basis-Practitioner](#).

## Tekstlig sammendrag

**Sammendrag av obligatoriske krav**

1. Én eller flere personidentifikatorer i Practitioner.identifier
  - Hver Practitioner.identifier må ha:
    - Én identifier.system der følgende identifikatorserier er lovlig:
      - 2.16.578.1.12.4.1.4.1 (Fødselsnummer)
      - 2.16.578.1.12.4.1.4.2 (D-nummer)
      - 2.16.578.1.12.4.1.4.4 (HPR-nummer)
    - Én identifier.value som er unik innenfor systemet
2. Ett eller flere navn i Practitioner.name
  - Hver Practitioner.name må ha:
    - En name.family
    - En name.given

**6.6.3 Obligatoriske og anbefalte RESTful-interaksjoner**

Nedenfor finner du en oversikt over de RESTful FHIR-interaksjonene som er påkrevet - for eksempel søk, og les operasjoner - for denne profilen. Se samsvarkrav for en komplett liste av støttede RESTful-interaksjoner.

GET [base]/Practitioner/[id]

**Eksempel:** GET [base]/Practitioner/504202

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Practitioner.

POST [base]/Practitioner/\_search{identifier=[system]][code]}

**Eksempel:**

POST [base]/Practitioner/\_search{identifier=urn:oid:2.16.578.1.12.4.1.4.1|13116900216}  
Content-Type: application/x-www-form-urlencoded

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Practitioner ved hjelp av en identifikator som for eksempel fødselsnummer, identifier ([hvordan utføre et token-søk](#)). Søket utføres ved bruk av HTTP-verbet POST og parameterne url-kodes på body i henhold til [application/x-www-form-urlencoded](#)

GET [base]/Practitioner?identifier=[system]][code]

**Eksempel:** GET [base]/Practitioner?identifier=urn:oid:2.16.578.1.12.4.1.4.4|4759982

*Støtte:* Obligatorisk

*Implementeringsnotater:* Henter en enkelt Practitioner ved hjelp av en identifikator som for eksempel HPR-nummer, identifier ([hvordan utføre et token-søk](#)).

GET [base]/Practitioner?family=[string]&given=[string]

**Eksempel:** GET [base]/Practitioner?family=Mikkelsen&given=Anne

*Støtte:* Obligatorisk

*Implementeringsnotater:* Søk basert på minimum 2 helsepersonellelementer, family ([hvordan utføre et strengsøk](#)) og given ([hvordan utføre et strengsøk](#)).

## 6.7 NO basis HumanName

Denne profilerte identifiserer ett sett med minimum forventninger til datatypen [HumanName](#), ved registrering, søk og uthenting av et enkelt navn. Den definerer hvilke kjerneelementer, utvidelser, vokabularer og kodeverk som SKAL være tilstede når man benytter denne profilen.

Obligatoriske elementer og terminologi

Følgende dataelementer er obligatoriske

**Hver enkelt HumanName må:**

1. Etternavn
2. Fornavn

I tillegg følgende elementer "må støttes" ([Must Support](#)).

**Dersom data er tilgjengelig skal HumanName inkludere:**

1. Mellomnavn
2. Brukskontekst for navnet

**Profilspesifikk implementeringsveiledning**

- I henhold til [lov om personnavn](#) er følgende kombinasjoner av HumanName.family lovlig:
  - Hansen
  - Olsen Hansen
  - Olsen-Hansen
- Kardinalitet for HumanName.given er endret fra 0..\* til 1..1, dette betyr at flere fornavn separeres ved hjelp av mellomrom.

### 6.7.1 Eksempler

- *Legg inn link til eksempel*

### 6.7.2 Formelle detaljer om profilen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-basis-HumanName>

Publisert den (*en-eller-annen-dato*) (ev. *som et utkast*) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på den norske basisprofilen [no-basis-HumanName](#).

Tekstlig sammendrag

#### **Sammendrag av obligatoriske krav**

1. Ett og opptil to etternavn i HumanName.family.
2. Ett eller flere fornavn i HumanName.given

#### **Sammendrag av "må støttes" krav**

1. Ett mellomnavn i HumanName.extension = NO core MiddleName utvidelsen

Tekstlig sammendrag

2. Én brukskontekst for navnet i `HumanName.use` fra kodeverket [NameUse](#).

## 7 Utvidelser (extensions)

### 7.1 Utvidelser (extensions) definert som en del av HelseAPI implementasjonsguiden

Følgende utvidelser (extensions) er definert som en del av HelseAPI implementasjonsguiden:

- [NO basis AddressOfficial](#)
- [NO basis MiddleName](#)
- [NO basis PropertyInformation](#)
- [NO basis UrbanDistrict](#)
- [patient-nationality](#)

### 7.2 NO basis AddressOfficial

Brukskontekst: [HelseAPI Address](#)

#### 7.2.1 Formelle detaljer om utvidelsen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-basis-address-official>

Publisert den (*en-eller-annen-dato*) (ev. som et utkast) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på [Extension](#).

Tekstlig sammendrag
<p><b>Sammendrag av obligatoriske krav</b></p> <ol style="list-style-type: none"> <li>1. En boolsk verdi som indikerer om dette er pasienten eller personen sin folkeregistrerte adresse</li> </ol>

### 7.3 NO basis MiddleName

Brukskontekst: [NO basis HumanName](#)

#### 7.3.1 Formelle detaljer om utvidelsen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-basis-middlename>

Publisert den (*en-eller-annen-dato*) (ev. som et utkast) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på [Extension](#).

Tekstlig sammendrag
<p><b>Sammendrag av obligatoriske krav</b></p> <ol style="list-style-type: none"> <li>1. Ett mellomnavn i HumanName.Extension = NO basis MiddleName som konformerer til <a href="#">Lov om personnavn</a></li> </ol>

## 7.4 NO basis PropertyInformation

Brukskontekst: [HelseAPI Address](#)

### 7.4.1 Formelle detaljer om utvidelsen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-basis-propertyinformation>

Publisert den (*en-eller-annen-dato*) (ev. som et utkast) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på [Extension](#).

#### Tekstlig sammendrag

##### Sammendrag av obligatoriske krav

1. Et heltall i castradalUnitNumber som angir gårdsnummeret for eiendommen tilhørende adressen
2. Et heltall i propertyUnitNumber som angir bruksnummeret for eiendommen tilhørende adressen
3. Et heltall i leaseholdNumber som angir festnummeret for eiendommen tilhørende adressen
4. Et heltall i condominiumUnitNumber
5. En kodeverdi bundet til kodeverket [Kommunennummer og regionale spesialkoder \(urn:oid:2.16.578.1.12.4.1.1.3402\)](#)
  - En valueCoding.system som angir kodeverket urn:oid:2.16.578.1.12.4.1.1.3402
  - En valueCoding.code der kodeverdi er hentet fra angitt kodeverk
  - En valueCoding.display der den tekstlige verdien er hentet fra angitt kodeverk

## 7.5 NO basis UrbanDistrict

Brukskontekst: [HelseAPI Patient](#)

### 7.5.1 Formelle detaljer om utvidelsen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.no/fhir/StructureDefinition/no-basis-urban-district>

Publisert den (*en-eller-annen-dato*) (ev. som et utkast) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på [Extension](#).

#### Tekstlig sammendrag

##### Sammendrag av obligatoriske krav

1. En urbanDistrict.valueCoding bundet til kodeverket [Bydelsnummer \(urn:oid:2.16.578.1.12.4.1.1.3403\)](#)
  - En valueCoding.system som angir kodeverket urn:oid:2.16.578.1.12.4.1.1.3403
  - En valueCoding.code der kodeverdi er hentet fra angitt kodeverk
  - En valueCoding.display der den tekstlige verdien er hentet fra angitt kodeverk



## 7.6 patient-nationality

Brukskontekst: [HelseAPI Patient](#)

### 7.6.1 Formelle detaljer om utvidelsen og profilinnhold

Den offisielle URL-en for profilen er: <http://hl7.org/fhir/StructureDefinition/patient-nationality>

Publisert den *(en-eller-annen-dato)* (ev. som et utkast) av HL7 Norge teknisk styringskomite.

Denne profilen bygger på [Extension](#).

Tekstlig sammendrag

#### Sammendrag av obligatoriske krav

1. Ett eller flere statsborgerskap/nasjonalitet i Patient.Extension = patient-nationality
  - En nationality.code som er bundet til kodeverket [ISO-3166](#)
  - En nationality.period som angir tidsperioden statsborgerskapet/nasjonaliteten var i effekt

## 8 Konformitetskrav

Denne seksjonen beskriver konformitetskravene HelseAPI tjenere og HelseAPI klienter. Identifiserer hvilke profiler, RESTful operasjoner og søkeparametere som MÅ støttes. NB: De individuelle HelseAPI profilene setter krav til strukturelle begrensninger og terminologibindinger.

### 8.1 Konformitetskrav for HelseAPI tjenere

- FHIR versjon: 4.0.0
- Støttede formater: xml, json
- Publisert: (*en-eller-annen-dato*)
- Publisert av: HL7 Norge - Teknisk styringskomité

Denne seksjonen beskriver de forventede kapabilitetene til en HelseAPI tjener som skal sørge for å svare på forespørsler fra HelseAPI klienter.

#### 8.1.1 Oppførsel

Beskrivelse:

HelseAPI tjenere SKAL:

- Støtte ressursprofilen HelseAPI Patient.
- Implementere RESTful-oppførsel slik den er beskrevet i FHIR spesifikasjonen.
- Kunne returnere følgende HTTP-statuskoder
  - 200 - OK: Operasjonen var en suksess
  - 400 - Bad Request: Ressursen kan ikke parses eller feilet en eller flere valideringsregler
  - 401 - Unauthorized: Forespørselen manglet nødvendig autorisasjon
  - 404 - Not Found: Ressurstypen støttes ikke / endepunktet eksisterer ikke
  - 410 - Gone: Ressursen er slettet
- Støtte ressursformatet *json* for alle HelseAPI interaksjoner.
- Deklarere en CapabilityStatement som identifiserer profiler, operasjoner og søkeparametere som HelseAPI tjeneren støtter.

HelseAPI tjenere BURDE:

- Støtte ressursformatet *xml* for alle HelseAPI interaksjoner.
- Angi hvilken HelseAPI profil tjeneren konformer til ved å angi FHIR meta.profile attributtet for hver instans.

#### 8.1.2 Sikkerhet

HelseAPI tjenere SKAL:

- Implementere sikkerhetskravene slik de er dokumentert i seksjonen [Sikkerhet](#).
- Sikre at hver API-forespørsel inkluderer en gyldig Authorization token: Authorization: Bearer {server-specific-token-here}
- Avvise uautoriserte forespørsler ved å returnere en HTTP 401 Unauthorized eller 404 Not Found statuskode i henhold til seksjonen som omhandler responshåndtering ved tilgangsnekt.

### 8.1.3 Profilinteraksjon

- HelseAPI tjenerer SKAL tilgjengeliggjøre interaksjonene [read](#) og [search](#).
- HelseAPI tjenerer BURDE tilgjengeliggjøre interaksjonene [vread](#) og [history-instance](#).

#### 8.1.3.1 Søkeparametere

Ressurstype	Støttede profiler	Støttede søk
Patient	<a href="#">HelseAPI Patient</a>	identifiser, name, family, given, gender, birthdate, name + gender, name + birthdate, family + gender, given + gender
Practitioner	<a href="#">HelseAPI Practitioner</a>	identifiser, name
Organization	<a href="#">HelseAPI Organization</a>	identifiser, name, address

### 8.1.4 Patient

Profiler: [HelseAPI Patient](#)

Interaksjoner og søkekriterier:

1. En HelseAPI tjener SKAL kunne returnere en pasient ved hjelp av følgende interaksjon:  
GET [base]/Patient/[id]
2. En HelseAPI tjener SKAL kunne returnere en pasient ved hjelp av en identifikator:  
POST [base]/Patient?identifiser=[system]][code]
3. En HelseAPI tjener SKAL kunne returnere en pasient når et minimum av 2 av følgende søkeparametere er samtidig tilstede:
  - name
  - gender
  - birthdate
    - F.eks:  
GET [base]/Patient?name=[string]&gender=[token]

Søkeparametere:

Konformitet	Parameter	Type
SKAL	identifiser	token
SKAL	name	string
SKAL	family + gender	string + token
SKAL	given + gender	string + token
SKAL	name + gender	string + token
SKAL	name + birthdate	string + date

### 8.1.5 Practitioner

Profiler: [HelseAPI Practitioner](#)

Interaksjoner og søkekriterier:

1. En HelseAPI tjener SKAL kunne returnere et helsepersonell ved hjelp av følgende interaksjon:  
GET [base]/Practitioner/[id]
2. En HelseAPI tjener SKAL kunne returnere et helsepersonell ved hjelp av en identifikator:  
POST [base]/Practitioner?identifiser=[system]][code]
3. En HelseAPI tjener SKAL kunne returnere et helsepersonell ved hjelp av et navn:  
GET [base]/Practitioner?family=[string]&given=[string]

Søkeparametere:

Konformitet	Parameter	Type
SKAL	identifiser	token
SKAL	name	string

### 8.1.6 Organization

Profiler: [HelseAPI Organization](#)

Interaksjoner og søkekriterier:

1. En HelseAPI tjener SKAL kunne returnere en organisasjon ved hjelp av følgende interaksjon:  
GET [base]/Organization/[id]
2. En HelseAPI tjener SKAL kunne returnere en organisasjon ved hjelp av en identifikator:  
GET [base]/Organization?identifiser=[system]][code]
3. En HelseAPI tjener SKAL kunne returnere en organisasjon ved hjelp av et navn:  
GET [base]/Organization?name=[string]
4. En HelseAPI tjener BURDE kunne returnere en organisasjon ved hjelp av en adresse:  
GET [base]/Organization?address=[string]

Søkeparametere:

Konformitet	Parameter	Type
SKAL	identifiser	token
SKAL	name	string
BURDE	address	string

## 8.2 Konformitetskrav for HelseAPI klienter

- FHIR versjon: 4.0.0
- Støttede formater: xml, json
- Publisert: *(en-eller-annen-dato)*
- Publisert av: HL7 Norge - Teknisk styringskomité

Denne seksjonen beskriver de forventede kapabilitetene til en HelseAPI klient som er ansvarlig for å opprette og initiere forespørsler om informasjon for en gitt pasient. Den komplette listen av FHIR profiler, RESTful interaksjoner og søkeparametre støttet av HelseAPI tjenere er beskrevet i seksjonen Konformitetskrav for HelseAPI tjenere.

## 8.2.1 Oppførsel

Beskrivelse:

HelseAPI klienten BURDE:

- Støtte uthenting og forespørsler av en eller flere av HelseAPI profilene ved å bruke RESTful interaksjoner og søkeparametere beskrevet i HelseAPI tjenerens CapabilityStatement.

Sikkerhet:

HelseAPI klienten SKAL:

- Implementere de sikkerhetskrav som er beskrevet i seksjonen [Sikkerhet](#).

## 8.2.2 Patient

Profiler: [HelseAPI Patient](#)

Interaksjoner og søkekriterier:

1. En klient BURDE kunne utføre en forespørsel om en pasient ved hjelp av følgende interaksjon:  
GET [base]/Patient/[id]
2. En klient BURDE kunne utføre en forespørsel om en pasient ved hjelp av en identifikator:  
POST: [base]/Patient?identifiser=[system]][code]
3. En klient BURDE kunne utføre en forespørsel om en pasient ved å benytte ett minimum av 2 søkeparametere. For å unngå for brede søk, BURDE f.eks. et søk på kjønn inkludere etternavn og fornavn.
  - name
  - gender
  - birthdate
    - F.eks:  
GET [base]/Patient?family=[string]&given=[string]&gender=[token]

## 8.2.3 Practitioner

Profiler: [HelseAPI Practitioner](#)

Interaksjoner og søkekriterier:

1. En klient BURDE kunne utføre en forespørsel om et helsepersonell ved hjelp av følgende interaksjon:  
GET [base]/Practitioner/[id]
2. En klient BURDE kunne utføre en forespørsel om et helsepersonell ved hjelp av en identifikator:  
POST [base]/Practitioner?identifiser=[system]][code]
3. En klient BURDE kunne utføre en forespørsel om et helsepersonell ved hjelp av et navn:  
POST [base]/Practitioner?family=[string]&given=[string]

## 8.2.4 Organization

Profiler: [HelseAPI Organization](#)

Interaksjoner og søkekriterier:

1. En klient BURDE kunne utføre en forespørsel om en organisasjon ved hjelp av følgende interaksjon:  
GET [base]/Organization/[id]

2. En klient BURDE kunne utføre en forespørsel om en organisasjon ved hjelp av en identifikator:  
GET [base]/Organization?identifiser=[system]][code]
3. En klient BURDE kunne utføre en forespørsel om en organisasjon ved hjelp av et navn:  
GET [base]/Organization?name=[string]
4. En klient BURDE kunne utføre en forespørsel om en organisasjon ved hjelp av adresse:  
GET [base]/Organization?address=[string]

## 9 Generelle sikkerhetsvurderinger

HelseAPI er laget for å håndtere alle typer informasjon inkludert sensitiv personinformasjon. Dette betyr at sikkerheten må vurderes nøye for å unngå at uvedkommende får tak i informasjonen som er tilgjengelig eller får tilgang til noen tjenester via HelseAPI.

De som implementerer HelseAPI (for eksempel ved støtte for SMART on FHIR og FHIR) må være oppmerksom på krav til sikkerhet og gjennomføre en vurdering av risiko og sikkerhet innenfor flere områder før HelseAPI kan etableres.

Ved tilkobling til helsenetten binder man seg til [Normens](#) krav til informasjonssikkerhet. Selv om det er ikke er et krav om at HelseAPI tilbys via helsenetten, BURDE en tilbyder av HelseAPI vurdere sikkerhetskravene som er definert i Normen.

I tillegg til dette har [Direktoratet for e-helse](#) utgitt referansearkitekturer for [datadeling](#) og [dokumentdeling](#). Under er det kort beskrevet noen områder og krav, men hver implementasjon må vurdere sikkerheten ved sin egen implementasjon.

### 9.1 Juridisk / Avtale

Ved etablering av HelseAPI SKAL det vurderes hvilke juridiske og avtalemessige krav som gjelder basert på informasjonen som tilbys. Dette kan inkludere databehandleravtaler eller andre juridiske krav for etablering av HelseAPI.

### 9.2 Krav til risikovurderinger

Ved etablering av HelseAPI SKAL det gjennomføres en risikovurdering, og denne skal dokumenteres for senere revisjon ved behov

### 9.3 Sikring av kommunikasjonen mot HelseAPI

For HelseAPI basert på REST, er det viktig av generell HTTP sikkerhet implementeres og det SKAL benyttes TLS eller annet transportsikring ved overføring av informasjon mot HelseAPI. Direktoratet for e-helse har utgitt en referansearkitektur for [datadeling](#) som burde vurderes.

For å støtte nettleaserbaserte klientprogrammer med flere integrasjoner, BURDE tjenerne implementere cross-origin Resource sharing (CORS)

### 9.4 Autentisering

HelseAPI i produksjon SKAL i utgangpunktet autentisere klienten (og bruker) som får tilgang og autentiseringen skal være i henhold til kravene for informasjonen som tilbys. HelseAPI kan likevel tilby endepunkter eller tjenester som ikke krever autentisering dersom informasjonen som tilbys ikke krever dette, og et eksempel på et slikt endepunkt er "capability statement"

### 9.5 Autorisasjon / tilgangskontroll

Når klient og bruker er *autentisert* må også tilgangen *autoriseres* for å sikre at informasjon og tjenester kun tilbys de som skal ha tilgang.

HelseAPI SKAL verifisere korrekt autorisasjon basert på informasjonen og tjenesten som tilbys. Dette gjelder uavhengig av om informasjonen skrives eller leses fra HelseAPI.

Normen inneholder informasjon og krav om autorisasjon og det finnes bl.a. veiledere for tilgangsstyring.

## 9.6 Logging / Audit

Ved tilgang til HelseAPI SKAL det logges nok informasjon til at det i etterkant er mulig å verifisere tilgangsbeslutningen.

## 9.7 Responshåndtering ved tilgangsnekt

Både ved bruk av REST og FHIR med REST er det viktig at svar fra en tjener følger standardisering av HTTP status koder. HTTP standarden definerer følgende status kodeserier:

- 1xx-serien er reservert for lavnivå HTTP-ting.
- 2xx-serien er reservert for vellykkede kall der alt går som planlagt.
- 3xx-serien er reservert for omadressering av trafikk.
- 4xx-serien er reservert for å svare på feil gjort av klienten, f.eks. feil inputdata eller at de ber om ting som ikke eksisterer. Disse forespørslene skal være idempotente, og skal ikke endre tjenerens tilstand.
- 5xx-serien er reservert for svar når tjeneren gjør en feil. Ofte blir disse feilene returnert av applikasjonene og normalt utenfor utviklerens rekkevidde. Dette er for å sikre at klientene faktisk mottar et svar. Klienten kan ikke muligens vite tilstanden til serveren når et 5xx-svar mottas, og derfor bør disse unngås.

Når klienter nektes tilgang, må en tjener gi nok informasjon slik at det dekker klientens behov for en god brukervennlighet. Tjeneren må returnere en http status kode som klienten kan presentere for brukeren sin. Nekting av tilgang kan skyldes flere forhold, for eksempel at bruker ikke er autorisert til å benytte API-et, bruker er ikke autorisert til å få tilgang til bestemte data eller at innbygger har sperret tilgang til dataene for brukeren.

Det er viktig når klienter nektes tilgang at svaret fra en tjener blir valgt med omhu. Dersom tjeneren returnerer for mye informasjon, kan dette avsløre sensitiv informasjon. For eksempel kan svaret fra en spørring etter en pasient sin diagnose som brukeren ikke har tilgang til, kunne gi opplysninger om at data om diagnosen faktisk finnes hos tjeneren. Dette kan brukeren sette sammen med annen informasjon som kan resultere i at brukeren kan forstå at pasienten har diagnosen.

FHIR-standardens har noen anbefalinger vedrørende håndtering av dette som vi ønsker å ta med som generelle anbefalinger for bruk av REST.

### Sammendrag:

Resultatet som gis av tjener når tilgang til data nektes må tilpasses basert på konteksten tilgangen nektes. Eksempler på en slik tilpasning:

1. Dersom det finnes data som bruker ikke skal vite at finnes, så må det ikke skilles mellom "ikke funnet" og "ikke tilgang".  
Ved benytte status kode 404 "not found" når bruker ikke har tilgang, så kan ikke bruker forstå at dataene finnes.
2. Dersom bruker ikke skal vite at pasient har sperret dataene, så må ikke bruker vite når han/hun har fått og ikke fått tilgang.  
Ved benytte status kode 404 "not found" når pasienten har sperret brukerens tilgang til dataene, så kan ikke bruker forstå at det finnes en spørring.
3. Dersom det er OK at bruker vet at han/hun ikke har tilgang, men ikke skal forstå hvordan han/hun skal kunne få tilgang, så bør kode 403 "forbidden" benyttes fremfor koden 401 "unauthorized" når tilgang nektes.
4. Dersom en tjener tillater å returnere lister med ressurser anbefales det å returnere en tom liste som et vellykket kall når bruker ikke har tilgang til å se listen. Da kan bruker ikke skille mellom der det ikke finnes data og der brukeren ikke har tilgang (og det finnes data).





## 10 Eksempler

- [no-helseapi-Practitioner-example.json](#)
- [no-helseapi-Practitioner-example.xml](#)

### 10.1 no-helseapi-Practitioner-example.json

```
{
  "resourceType": "Practitioner",
  "id": "example",
  "meta": {
    "profile": [
      "http://hl7.no/fhir/StructureDefinition/no-helseapi-Practitioner"
    ]
  },
  "identifier": [
    {
      "system": "urn:oid:2.16.578.1.12.4.1.4.4",
      "value": "9144889"
    }
  ],
  "name": [
    {
      "family": "Koman",
      "given": [
        "Magnar"
      ]
    }
  ],
  "telecom": [
    {
      "system": "phone",
      "value": "75589889",
      "use": "work"
    },
    {
      "system": "fax",
      "value": "75589899"
    },
    {
      "system": "phone",
      "value": "98128921",
      "use": "mobile"
    },
    {
      "system": "email",
      "value": "mako@bottomline.no",
      "use": "work"
    }
  ],
  "address": [
    {
      "use": "work",
      "line": [
        "Dreyfushammarn 23"
      ],
      "city": "BODØ",
      "postalCode": "8012"
    }
  ],
  "qualification": [
    {
```

```

    "code": {
      "coding": [
        {
          "system": "urn:oid:2.16.578.1.12.4.1.1.9060",
          "code": "LE",
          "display": "Lege"
        }
      ]
    }
  },
  {
    "code": {
      "coding": [
        {
          "system": "urn:oid:2.16.578.1.12.4.1.1.7704",
          "code": "1",
          "display": "Autorisasjon"
        }
      ]
    }
  },
  {
    "code": {
      "coding": [
        {
          "system": "urn:oid:2.16.578.1.12.4.1.1.7426",
          "code": "152",
          "display": "Ortopedisk kirurgi"
        }
      ]
    }
  }
]
}

```

## 10.2no-helseapi-Practitioner-example.xml

```

<Practitioner xmlns="http://hl7.org/fhir">
  <id value="example"/>
  <meta>
    <profile value="http://hl7.no/fhir/StructureDefinition/no-helseapi-Practitioner"/>
  </meta>
  <identifier>
    <!-- HPR number for the Practitioner -->
    <system value="urn:oid:2.16.578.1.12.4.1.4.4"/>
    <value value="9144889"/>
  </identifier>
  <name >
    <family value="Koman"/>
    <given value="Magnar"/>
  </name>
  <telecom>
    <system value="phone"/>
    <value value="75589889"/>
    <use value="work"/>
  </telecom>
  <telecom>
    <system value="fax"/>
    <value value="75589899"/>
  </telecom>

```

```

<telecom>
  <system value="phone"/>
  <value value="98128921"/>
  <use value="mobile"/>
</telecom>
<telecom>
  <system value="email"/>
  <value value="mako@bottomline.no"/>
  <use value="work"/>
</telecom>
<address>
  <!-- Postal address for the practitioner -->
  <!-- Norwegian basisprofil for Address resource adding Norwegian
specific property information, official use of address and further
explanation of the use for the data-elements in a Norwegian address. -->
  <use value="work"/>
  <line value="Dreyfushammarn 23"/>
  <city value="BODØ"/> <!-- Poststed -->
  <postalCode value="8012"/> <!-- postnummer -->
</address>
<!-- Kvalififikasjon Kategori helsepersonell (OID=9060) -->
<qualification>
  <code>
    <coding>
      <system value="urn:oid:2.16.578.1.12.4.1.1.9060"/>
      <code value="LE"/>
      <display value="Lege"/>
    </coding>
  </code>
</qualification>
<!-- Kvalififikasjon autorisasjonskode Godkjenningstype HPR (OID=7704) -
->
<qualification>
  <code>
    <coding>
      <system value="urn:oid:2.16.578.1.12.4.1.1.7704"/>
      <code value="1"/>
      <display value="Autorisasjon"/>
    </coding>
  </code>
</qualification>
<!-- Kvalifikasjon Helsepersonellregisterets (HPR) klassifikasjon av
spesialiteter (OID=7426)-->
<qualification>
  <code>
    <coding>
      <system value="urn:oid:2.16.578.1.12.4.1.1.7426"/>
      <code value="152"/>
      <display value="Ortopedisk kirurgi"/>
    </coding>
  </code>
</qualification>
</Practitioner>

```