

Kompetencefonde webservice API beskrivelse

Indhold

1. Introduktion	3
2. Termer og forkortelser	3
3. Systemarkitektur	4
3.1 Aktører og roller	5
4. Webservice beskrivelse.....	5
4.1 WSDL url.....	5
4.2 Metode: getCourseParticipation	5
Input	6
Output	6
Definition af status Gennemført.....	7
Definition af status RegistreringKomplet	7
Beregning af tilstedeværelses timer	7
5. Sikkerhed.....	7
5.1 Overordnet om adgang til webservices på efteruddannelse.dk	7
5.2 Timestamp	7
5.3 Signering.....	8
5.4 Kryptering	8
5.5 Servicen konkret	8
6. Implementering af Webserviceklient.....	8
6.1 Eksempel service og klient	9
6.2 Medfølgende filer	9
6.3 Hello World service endpoints	9
7. XSD Definitioner.....	10

Historik			
Dokumentid.	Dato	Udarb. af	Ændringer
A114.8968.1	02.07.2012	Hoym	Initiel version
A114.8968.2	03.09.2012	Hoym	Kommentarer indarbejdet
A114.8968.3	24.09.2012	Hoym	Mindre korrektioner

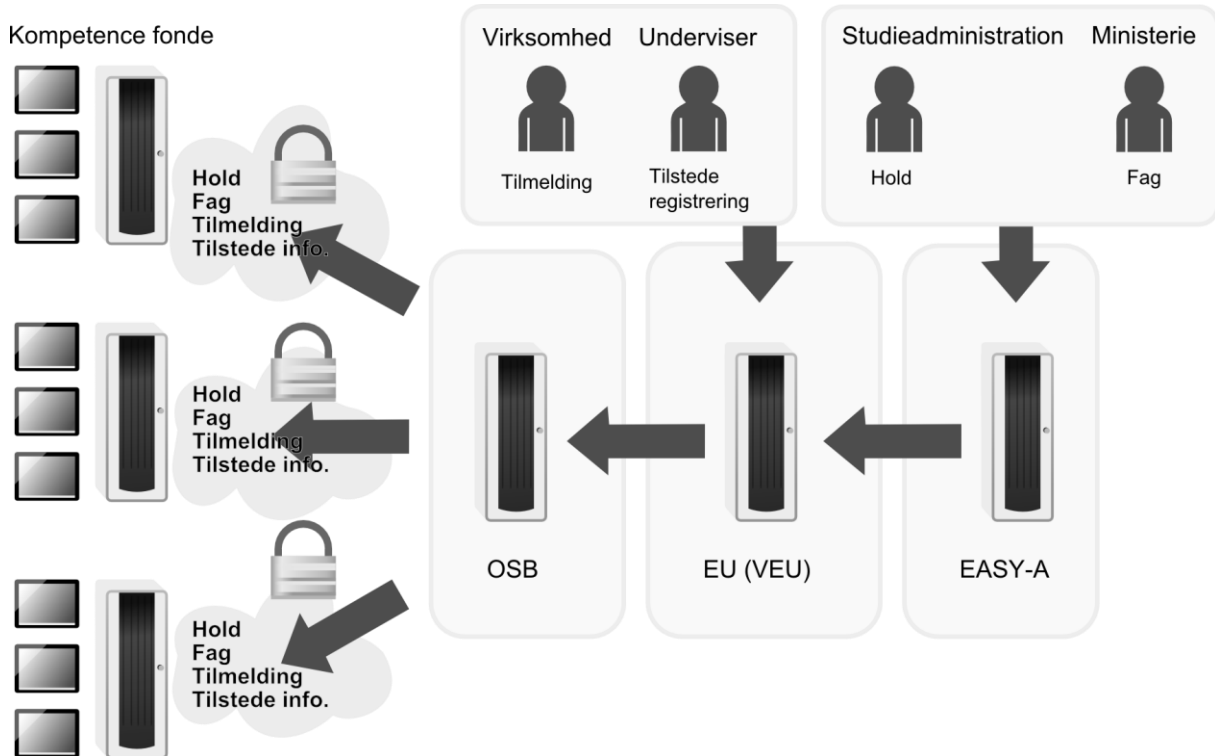
1. Introduktion

For at lette arbejdsgangen i forbindelse med ansøgning om uddannelsestilskud. Skal kompetencefonde kunne få adgang til at trække oplysninger, for specifikke personer fra EfterUddannelse.dk. For hver person der er forespurgt på returneres kursusinformation, pris og om kursistens tilstedeværelse.

2. Termer og forkortelser

EU	EfterUddannelse.dk er en landsdækkende tilmeldings- og ansøgningsportal for erhvervsrettet voksen- og efteruddannelse, herunder AMU-kurser. På portalen kan virksomheder og kursister finde kurser, foretage tilmeldinger, ansøge digitalt om VEU-godtgørelse og befordringstilskud samt få overblik over kursusaktiviteter og status for ansøgninger.
EASY-A	EASY-A er erhvervsskolerne og erhvervsakademiernes studieadministrative system. EU får kursusstamdata fra uddannelsesstedernes EASY-A baser
OSB	Oracle Service Bus. Komponent i EUs system arkitektur der tilbyder eksterne systemer sikker adgang til EUs webservices.
Hold/kursus	Et hold (kursus) benyttes i som et redskab til at sammenkoble en række elever, der skal følge et givent undervisningsforløb (specifikke fag/amu-mål). Ved oprettelsen af et hold specificerer uddannelsesstedet, hvordan det specifikke undervisningsforløb tilrettelægges – herunder start- og slutdato for hold og tilknyttede fag.

3. Systemarkitektur



OSB Oracle Service Bus: Værktøj til at sammensætte og administrere webservices. OSB'en håndterer WS-Security (signering og kryptering) og delegerer til interne webservices for alle EU webservices, så dette ikke skal konfigureres for hver enkelt webservice. OSB afvikles på en Oracle Weblogic Server.

EU (VEU): Er den centrale server der implementerer forretningslogikken for både brugerflade og services i EU.dk. Afvikles på en Oracle Weblogic Server.

EASY-A: Er baseret på Oracle Forms og integrerer til EU vha. webservices.

3.1 Aktører og roller

Webservicen aggregerer en delmængde af de data, der skabes i EU.dk og fra de fødesystemer som EU.dk integrerer til. De aktører der leverer data er:

Ministeriet for Børn og Undervisning

Leverer staminformation omkring de fag, som udbydes.

Studieadministrationen på uddannelsesstederne

Definerer selve afviklingen af fagene ved at definere undervisningsperioder og priser gennem EASY-A (holdstamoplysninger).

Virksomheder

Tilmelder medarbejdere gennem EU.dk på de hold/kurser, der er oprettet af uddannelsesstederne vha. EASY-A.

Undervisere/studieadministration

Registrerer tilstedeværelse for medarbejdere (kursister) gennem EU eller direkte i EASY-A.

4. Webservice beskrivelse

Webservicen består af en metode der aggregerer hold-, fag-, tilmeldings- og tilstedeværelses data for gennemførte kurser (afsluttede).

Det er kun muligt at hente oplysninger om tilmeldinger, der er knyttet til virksomheder, dvs. tilmeldinger af typen MEDARBEJDER, SELVSTÆNDIG og VIRKSOMHED.

Tilmeldinger af typen "PRIVATPERSON" og "LEDIG" sorteres fra, når EU.dk returnerer oplysninger – også selvom de er inkluderet i liste af cpr-numre, som der forespørges på.

Bemærk desuden at virksomhederne – via en virksomhedsindstilling på virksomhedens side på EfterUddannelse.dk – har mulighed for at angive, at de ikke ønsker dataudveksling med kompetencefonde. Hvis virksomheden ikke ønsker dataudveksling med kompetencefonde, vil dette fremgå af svaret fra EU.

4.1 WSDL url

<http://<server:port>/Kompetenceudviklingsfonde/KompetenceudviklingsfondeProxyService?wsdl>

4.2 Metode: getCourseParticipation

For at modtage oplysninger fra servicen sendes et request med en liste af CPR-numre over de personer, der ønskes oplysninger om. Foruden listen af CPR-numre kan der angives en række parametre, der afgrænser det returnerede resultat.

UvmFagListe: afgrænser søgningen til kun at vise gennemførte kurser/hold, hvori der indgår bestemte fag.

MatchEksakteFag: sat til true fremsøges de hold, som indeholder alle fag i den UvmFagListe. Holdene må gerne indeholde andre fag.

MatchEksakteFag: sat til false fremsøges de hold, som indeholder mindst et af fagene i den angivne liste.

Er UvmFagListen tom har MatchEksakteFag ingen indflydelse på resultatet.

Fra- og Til Dato: afgrænser søgningen til kun at vise afsluttede kurser i det angivne tidsrum. Hvis det kun er fra eller til dato der er udfyldt laves åben søgning. Ved afsluttede kurser forstås kurser hvor kursets slutdato ligger før dags dato. Alle hold hvis periode overlapper med den periode der er angivet med Fra-og Til Dato vil blive returneret. F.eks. medtages et hold der afvikles i perioden: 15/5 – 16/6

Hvis der søges på:

1/5 – 31/5
1/6 – 30/6
17/5 – 1/6

Input

Xsd felt	Beskrivelse	Type
FraDato	Fra dato i søgeinterval for afsluttede kurser (optional)	Date
TilDato	Til dato i søgeinterval for afsluttede kurser (optional)	Date
UvmFag	5-cifrede Uvm fag-kode	String(5)
MatchEksakteFag	Skal der kun søges på hold der indeholder de angivne fag	Boolean
Cpr	CPR for forespurgte person(er). Med eller uden bindestreg (-) Krævet felt.	String(10)

Output

Xsd felt	Beskrivelse	Type
Cpr	CPR. 10 cifret kode, uden bindestreg (-)	String(10)
VirksomhedTillader-Dataudveksling	Tillader virksomheden udveksling af kursusdata	Boolean
HoldStartDato	Start dato for kursus	Date
HoldSlutDato	Slut dato for kursus	Date
HoldBetegnelse	Betegnelse for kursus	String
HoldKviknummer	Kursus id	String
ArbejdsCvr	CVR nummer angivet på tilmeldingen	String
ArbejdsadressePnr	Pnr for arbejdsadresse angivet på tilmeldingen	String
EmailType	Type på email fra tilmelding (Arbejdsgiver/Kursist)****	String
Emailadresse	Emailadresse fra tilmeldingen****	String
SkoleNavn	Skolens navn	String
SkoleDsnr	Skolens Ds-nr.	String
SkoleTelefonNr	Skolens telefon nummer	String
UvmFag	5-cifrede Uvm fag-kode	String(5)
Niveau	Fagets niveau, typisk bindestreg for AMU hold	String
FagBetegnelse	Fagets betegnelse (Betegnelsen stammer fra den betegnelse som er blevet godkendt i Ministeriet for Børn og Undervisning).	
StartDato	Startdato for en periode for et fag	Date
SlutDato	Slutdato for en periode for et fag	Date
VarighedDage	Samlet varighed for faget.	Decimal
TimerPrDag	Antal timer pr. dag	Decimal
Pris	Pris for det enkelte fag*	Decimal
Gennemført	Markering for om faget er gennemført	String
RegistreringKomplet	Markering for om fraværsregistrering er gennemført i hele perioden**	Boolean
TilstedeværDato	En enkelt dag i kurset	Date
FaktiskTilstedevær	Registrerede tilstedeværende timer for dato**	Decimal
TimerPrDag	Antal Timer pr. dag	Decimal
Tillægspris	Evt. tillægspriser***	Decimal
TillægsprisÅrsag	Evt. tillægsprisårsag ***	String

* Bemærk: Vi er ikke i stand til at oplyse om opkrævet beløb og betalt beløb på en tilmelding, men kan oplyse om de beløb der burde opkræves, som er de beløbet der er angivet her. Se uddybende beskrivelse forklaring på beregningen af prisen nedenfor.

** Se uddybende beskrivelse for beregning af tilstedeværelses timer nedenfor.

*** Pristillæg er udifferentierede. Bemærk der kan forekomme pristillæg på både hold og fag niveau.

**** Afhænger af a e-mail adressen er registreret i forbindelse med tilmeldingen.

Beregning af priser i EU.dk

1. Når prisen vises i direkte relation til en tilmelding (i prioriteret orden):
 - a. Hvis der ved tilmeldingen oplyses at kursisten er ledig vises den fulde pris.
 - b. For AMU-kurser gælder desuden at hvis der ved tilmelding oplyses at kursisten er fra Grønland eller Færøerne og ikke er bosiddende eller har beskæftigelse i Danmark vises den fulde pris.
 - c. Hvis der ved tilmelding som højeste uddannelsesniveau oplyses "Videregående uddannelse" eller "Andre uddannelser på videregående niveau", vises den fulde pris.
 - d. Hvis der ved tilmelding som højeste uddannelsesniveau oplyses "Anden uddannelse", så vises fuld pris Hvis der ved en tilmelding er angivet at kursisten deltager i et GVU-forløb vises en pris på 0 kr.

- e. I situationer, hvor de oplysninger, der er nævnt som beregningsgrundlag for prisen under de foregående punkter endnu ikke er angivet, så vises prisen baseret på stats-tilskud
- f. I alle andre situationer vises prisen baseret på statstilskud.

Evt. tillægspriser er ikke regulerede for statstilskud uanset hvilken type kursus der er tale om, dvs. tillægspriser er ens uanset om deltagerbetalingen er baseret på statstilskud eller om der skal betales fuld pris pga. videregående uddannelse fx.

Definition af status Gennemført

Gennemført kan antage flg. værdier

Ikke tilladt	Uddannelsesstedet tillader ikke, at der udleveres tilstedeværelsesoplysninger.
Ikke registreret	Der er endnu ikke registreret tilstedeværelse for personen
Gennemført med kursusbevis	Personen har gennemført kurset og har fået udleveret kursusbevis
Gennemført uden kursusbevis	Personen har gennemført kurset men har ikke fået udleveret kursusbevis
Ikke gennemført	Personen har ikke gennemført kurset

Definition af status RegistreringKomplet

True	Tilstedeværelse er registreret for alle tilstededage
False	Der er ikke registreret tilstedeværelse for alle tilstededage

Beregning af tilstedeværelses timer

FaktiskTilstede: beregnes på baggrund af undervisningsdagens normal timetal – evt. fravær.
TimerPrDag: undervisningsdagens normal timetal.

Eksempler:

FaktiskTilstede = 0, TimerPrDag = 7,4 -> Kursisten er registreret som fuldt fraværende
FaktiskTilstede = 5, TimerPrDag = 7,4 -> Kursisten er registreret som delvist fraværende
FaktiskTilstede = 7,4, TimerPrDage = 7,4 -> Kursisten er registreret som fuldt tilstede

5. Sikkerhed

5.1 Overordnet om adgang til webservices på efteruddannelse.dk

Adgang til webservices til EU kan kun ske efter aftale med Uni-C. Ønskes fonden at teste op mod UNI-C's test system skal der eksplicit åbnes for en firewall, for en given ip-adresse, for at services kan tilgås.

Webservicen er offentlig tilgængelig over http men der er benyttet WS-Security for at sikre fortrolighed, dataintegritet og autentificering.

<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>

Til signering bruges virksomhedscertifikater der skal være signeret af rodcertifikat fra DanID.

Kryptering foregår med Uni-C certifikat (offentlige nøgle).

Det er kun adgang til de udstillede services, hvis der bruges virksomhedscertifikater hvor certifikatets CVR nummer er registreret som kompetencefond i EfterUddannelse.dk

5.2 Timestamp

På både klient og serverside tilføjes SOAP headeren et timestamp.

Se side 36 i specifikationen:

<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>

5.3 Signering

For at sikre dataintegritet og autentificeringen signeres visse elementer i Webservicekaldet. Signatur tilføjes både på server og klient.

Signeringsalgoritmerne der skal bruges er specificeret i Policy elementerne i Servicens WSDL

Følgende elementer signeres

Element	Namespace
Timestamp	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd
Token	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd
Body	http://schemas.xmlsoap.org/soap/envelope/

5.4 Kryptering

For at sikre fortrolighed krypteres Body elementet i webservicekaldet.

Body elementet krypteres både i request og response.

Krypteringsalgoritmerne der skal bruges er specificeret i Policy elementerne i Servicens WSDL

Serverens certifikat med public nøgle til kryptering findes i WSDL filen under

`wsse:BinarySecurityToken`

Følgende elementer krypteres

Element	Namespace
Body	http://schemas.xmlsoap.org/soap/envelope/

5.5 Servicen konkret

Som beskrevet ovenfor bruges både kryptering og signering. Til det formål bruges Sign.xml og Entrypt.xml som er predefinerede elementer i WSDL'en:

Entrypt.xml : This policy requires clients to encrypt the SOAP body with 3DES-CBC. The key wrapping algorithm is RSA 1.5 A symmetric key for the Triple Data Encryption Standard (3DES) is generated by the client and encrypted for the recipient with RSA 1.5.

Sign.xml : This policy requires clients to sign the SOAP body. It also requires that the client add a signed timestamp to the wsse:Security header. Any system header present in the message must also be signed (system headers are WSRM headers, WS-Addressing headers. And the WSS timestamp) The digital signature algorithm is RSA-SHA1. Exclusive XML canonicalization is used.

6. Implementering af Webserviceklient

Den tekniske implementering af en Webservice klient for kompetencefonde, følger samme mønster som andre Webservices i EU.dk, idet der skal programmeres(genereres) en webserviceklient til det interface, der definerer webservicen.

Historisk har der været en del vanskeligheder for 3. part ift. at få hul igennem til webservices på EfterUddannelse.dk grundet sikkerhedselementerne. Da der benyttes forskellige værktøjer og opsætninger har det samtidigt været vanskeligt at yde støtte og fejlsøgning på den enkelte klient. For at imødegå dette problem, er der udviklet en simpel Hello World service og tilhørende klient, med det formål at udgøre et fælles referencepunkt.

6.1 Eksempel service og klient

Hello World Servicen er identisk med de øvrige services, der udbydes hvad angår sikkerhedskonfiguration. Beskedformatet for servicen er dog minimalt. Servicen tager som input et virksomhedsnavn og giver som output virksomhedsnavn efterfulgt af CVR nummeret, der findes i certifikatet, der signeres med.

Hello World-klienten kan kalde servicen. Adgang til klienten og den tilhørende kildekode giver mulighed for at afvikle et kald i mod testmiljøet, og inspiration til, hvordan en klient kan konstrueres fra bunden. Hello World klienten er udviklet i Java og er baseret på Apaches CXF standard framework til kald af policybaserede webservices. Der er naturligvis også andre muligheder, så som Axis frameworket og custom kode. Det er ikke et krav at CXF anvendes, der er blot taget udgangspunkt heri, da det synes at være det pt. mest standardiserede framework.

På .Net siden er der inkluderet eksempelkode - venligst stillet til rådighed af Facilia F.M.B.A. Sidstnævnte kode retter sig ikke imod HelloWorld servicen, men sikkerhedskonfigurationen vil være den samme.

6.2 Medfølgende filer

Hello World klienten findes i filen EfteruddannelseWebserviceClient.zip. Filen inkluderer en binær udgave af klienten samt kildekode. Læs filen readme.txt i zip filen for nærmere beskrivelse af hvordan klienten afvikles, samt hvordan koden er struktureret i projektet.

Eksempelkoden fra Facilia findes i filen EfteruddannelseEksempelFraFACILIA.zip.

6.3 Hello World service endpoints

Hello World servicen kan nås på følgende endpoints

HELLO WORLD SERVICE	
TEST	http://test.efteruddannelse.dk:8780/HelloWorld/HelloWorldService?wsdl
DRIFT	http://efteruddannelse.dk:7789/HelloWorld/HelloWorldService?wsdl

7. XSD Definitioner

WSDL (Uden WS-Security)

```
<?xml version='1.0' encoding='UTF-8'?>
<!-- Published by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is Oracle JAX-WS 2.1.5. -->
<!-- Generated by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is Oracle JAX-WS 2.1.5. -->
<definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://www.logica.com/veupublic/v100"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/" target-
Namespace="http://www.logica.com/veupublic/v100" name="KompetenceUdvFondeWebservice_v100">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://www.logica.com/veupublic/v100" schemaLoca-
tion="http://172.29.80.62:8201/VEU_v100/KompetenceUdvFondeWebservice_v100?xsd=1"/>
    </xsd:schema>
    <xsd:schema>
      <xsd:import namespace="http://www.logica.com/veu/kompetenceudviklingsfonde/FremsogTilstedeverResponse" schemaLoca-
tion="http://172.29.80.62:8201/VEU_v100/KompetenceUdvFondeWebservice_v100?xsd=2"/>
    </xsd:schema>
    <xsd:schema>
      <xsd:import namespace="http://www.logica.com/veu/kompetenceudviklingsfonde/FremsogTilstedeverRequest" schemaLoca-
tion="http://172.29.80.62:8201/VEU_v100/KompetenceUdvFondeWebservice_v100?xsd=3"/>
    </xsd:schema>
  </types>
  <message name="getCourseParticipation">
    <part name="parameters" element="tns:getCourseParticipation"/>
  </message>
  <message name="getCourseParticipationResponse">
    <part name="parameters" element="tns:getCourseParticipationResponse"/>
  </message>
  <portType name="KompetenceUdvFondeWebservice_v100">
    <operation name="getCourseParticipation">
      <input message="tns:getCourseParticipation"/>
      <output message="tns:getCourseParticipationResponse"/>
    </operation>
  </portType>
  <binding name="KompetenceUdvFondeWebservice_v100PortBinding" type="tns:KompetenceUdvFondeWebservice_v100">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
    <operation name="getCourseParticipation">
      <soap:operation soapAction=""/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
    </operation>
  </binding>
  <service name="KompetenceUdvFondeWebservice_v100">
    <port name="KompetenceUdvFondeWebservice_v100Port" binding="tns:KompetenceUdvFondeWebservice_v100PortBinding">
      <soap:address location="http://172.29.80.62:8201/VEU_v100/KompetenceUdvFondeWebservice_v100"/>
    </port>
  </service>
</definitions>
```

Request:

```
<xs:element name="FremsogTilstedeverRequest">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="FraDato" type="xs:date" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

<xs:element name="TilDato" type="xs:date" minOccurs="0"/>
<xs:element name="UvmFagListe" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="MatchEksakteFag" type="xs:boolean"/>
      <xs:element name="UvmFag" type="type:UvmFag" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="CprListe">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Cpr" type="type:CPRNummer" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>

```

Response:

```

<xs:element name="FremsoegTilstedeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Elev" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Cpr" type="type:CPRNummer"/>
            <xs:element name="VirksomhedTilladerDataudveksling" type="xs:boolean"/>
            <xs:element name="HoldListe" minOccurs="0">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="Hold" maxOccurs="unbounded">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="HoldStartDato" type="xs:date"/>
                        <xs:element name="HoldSlutDato" type="xs:date"/>
                        <xs:element name="HoldBetegnelse" type="xs:string"/>
                        <xs:element name="Kviknummer" type="xs:string"/>
                        <xs:element name="ArbejdsOplysninger">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="ArbejdsCvr" type="type:CvrNummer"/>
                              <xs:element name="ArbejdsadressePnr" type="type:PNummer"/>
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                        <xs:element name="EmailOplysninger" minOccurs="0">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="EmailType" type="type:EmailTypeEnum"/>
                              <xs:element name="Emailadresse" type="xs:string"/>
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                        <xs:element name="Kursusudbyder">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="SkoleNavn" type="xs:string"/>
                              <xs:element name="SkoleDsnr" type="xs:string"/>
                              <xs:element name="SkoleTelefonNr" type="xs:integer"/>
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                      </xs:sequence>
                    </xs:complexType>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

</xs:element>
<xs:element ref="ns1:PristillegListe" minOccurs="0"/>
<xs:element name="FagListe" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Fag" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="UvmFag" type="type:UvmFag"/>
            <xs:element name="Niveau"/>
            <xs:element name="FagBetegnelse" type="xs:string"/>
            <xs:element name="VarighedDage" type="xs:decimal"/>
            <xs:element ref="ns1:PeriodeListe"/>
            <xs:element name="Pris" type="xs:decimal"/>
            <xs:element ref="ns1:PristillegListe" minOccurs="0"/>
            <xs:element name="Gennemfort" nillable="true"/>
            <xs:element name="TilstedeverListe">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="RegistreringKomplet" type="xs:boolean"/>
                  <xs:element name="Tilstedever" maxOccurs="unbounded">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="Dato" type="xs:date"/>
                        <xs:element name="FaktiskTilstedever" type="xs:decimal"/>
                        <xs:element ref="ns1:TimerPrDag" />
                      </xs:sequence>
                    </xs:complexType>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="PristillegListe" type="type:PristillegListe"/>
<xs:element name="PeriodeListe" type="type:PeriodeListe"/>

```