
Systemarkitekturdokument for Healthcare@Public

MATHIAS HILLERUP LARSEN 05809
SIMON TOFT BRUUN 05766
ANDERS FÆRCH NIELSEN 05744

Revisions historie

Dato	Version	Beskrivelse	Forfatter
09.10.2008	1.0	Første version	Gruppe 08023
15.12.2008	2.0	Gennemrettet version	Gruppe 08023

Indholdsfortegnelse

1. INTRODUKTION	5
1.1 Formål og omfang	5
1.2 Referencer	5
1.3 Definitioner og forkortelser.....	5
1.4 Dokumentstruktur og læsevejledning	6
1.5 Dokumentets rolle i en iterativ udviklingsproces.....	6
2. SYSTEM OVERSIGT	7
2.1 System kontekst	8
2.2 System introduktion	8
3. SYSTEMETS GRÆNSEFLADER	9
3.1 Grænseflader til person aktører.....	9
3.1.1 Administrator	9
3.1.2 Praktiserende læge	9
3.1.3 Hospitalslæge.....	9
3.1.4 Sygeplejerske	9
3.1.5 Sundhedspersonale.....	9
3.1.6 Pårørende	10
3.1.7 Vagt.....	10
3.2 Grænseflader til eksterne system aktører	11
3.3 Grænseflader til hardware aktører.....	11
3.4 Grænseflader til eksterne software aktører	11
3.4.1 SIH Presentation Framework.....	11
4. USE CASE VIEW	12
4.1 Oversigt over arkitektursignifikante Use Cases.....	12
5. LOGISK VIEW.....	12
5.1 Oversigt	12
5.1.1 Subsystem – Central Administration System + Relatives	12
5.1.2 Subsystem PatientData Plugin	13
5.1.3 Subsystem Reminder Plugin	13
5.1.4 Subsystem – Central Surveillance System.....	13
5.2 Arkitektursignifikante designpakker.....	14
5.2.1 Sih. PublicServices.Bll	14
5.2.2 Sih. PublicServices.Dal.....	15
5.2.3 Sih. PublicServices.Dto	17
5.2.1 Framework.PlugIn.BLL.....	19
5.2.2 Framework.PlugIn.DTO	20
5.3 Use Case realiseringer.....	21
5.3.1 Subsystem – Administrator.....	22
5.3.2 Subsystem – Relatives	35
5.3.3 Subsystem – PatientData Plugins	37
5.3.4 Subsystem – Reminder Plugin	50
5.3.1 Subsystem – Central Surveillance System.....	53
5.4 Klassebeskrivelser.....	57
5.4.1 Subsystem: Central Administration System	57

5.4.2 Subsystem: Relatives	58
5.4.1 Subsystem: PatientDataPlugin	58
5.4.2 Subsystem: ReminderPlugin	60
5.4.3 Subsystem: Central Surveillance System	60
6. DEPLOYMENT VIEW	62
6.1 Oversigt over systemkonfiguration	62
6.1.1 Deployment af ASP.NET applikationer	62
6.1.1 Deployment af Central Survailliance System og Plugins	63
6.2 Systemkonfiguration	63
6.3 Node-beskrivelser	63
6.3.1 Node Database	63
6.3.2 Node Framework	63
6.3.3 Node SQLcmd beskrivelse	63
7. Data View	64
8. Generelle designbeslutninger	65
8.1 HealthCare@Central struktur	65
8.2 WPF og Plugin	65
8.3 Databasestrukturen	65
8.4 Implementeringsværktøjer	66
9. STØRRELSE OG YDELSE	66
9.1 ASP.NET applikationerne	66
9.2 Plugins	66
9.3 Central Survailliance System	66
10. KVALITET	67
10.1 ASP.NET applikationerne	67
10.2 Plugins	67
10.3 Central Survailliance System	67

1. INTRODUKTION

1.1 Formål og omfang

Formålet med dette dokument er at dokumentere designet af delsystemet HealthCare@Public(H@P). Alle aspekter af designet er beskrevet heri, og ved at læse dette dokument kan man opnå fuldt indblik i designet af H@P.

Test af systemet er beskrevet i separate testdokumenter.

Dette dokument henvender sig til den interesserede læser, som ønsker at studere designet af H@P nærmere. Det kan være for at videreudvikle H@P eller forstå og lære af designet med henblik på et andet system. Det forventes at læseren er på et teknisk niveau svarende til ingeniørstuderende på 7. semester.

1.2 Referencer

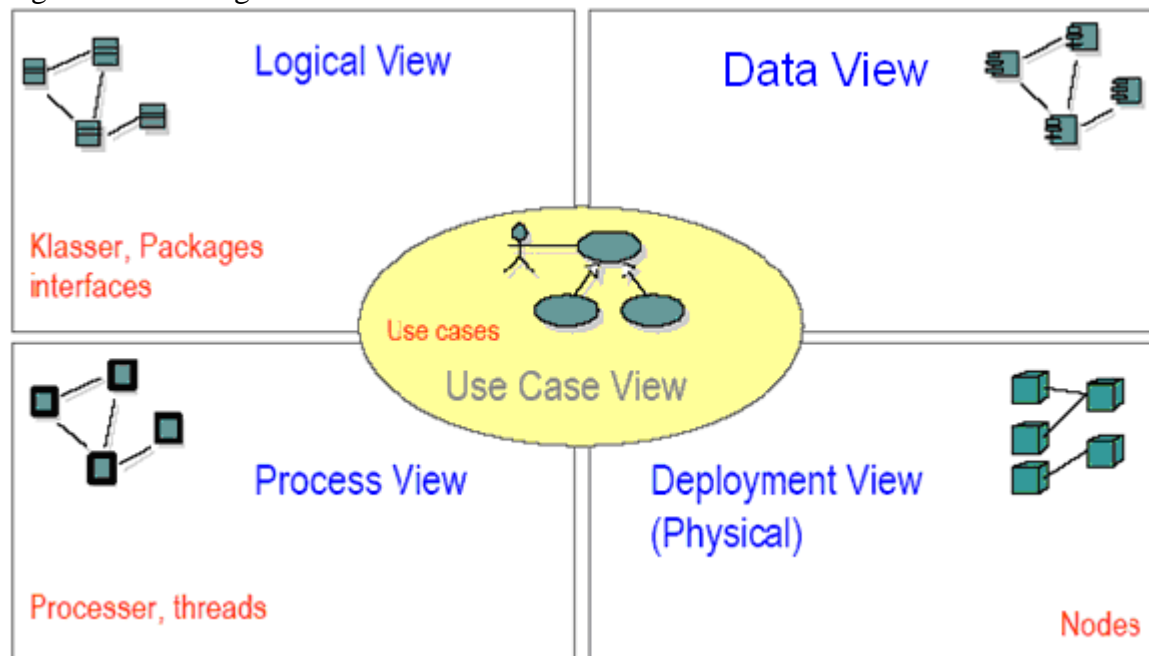
- [1] Kravspecifikation for HealthCare@Public.
- [2] Accepttestrapport for HealthCare@Public.
- [3] Systemarkitektur dokument for SIH Public Services.

1.3 Definitioner og forkortelser

- **Person:** Generel betegnelse for de persontyper der er knyttet til systemet (Sygeplejerske, Praktiserende Læge, Pårørende, osv...)
- **Patient:** En person der får foretaget målinger i HealthCare@Home
- **Ældre:** En anden betegnelse for en Patient i systemet
- **HealthCare:** Betegnelse for det overordnede system, der inkluderer Public, Home og Central
- **HealthCare@Public:** Det er navnet/betegnelsen for de klienter, der er til rådighed for Personer i systemet. HealthCare@Public er også navnet på projektet.
- **HealthCare@Central:** Det er navnet/betegnelsen for serveren hvor MS SQL databasen fysisk ligger, samt der hvor ASP.NET applikationen og vores WCF services er publiceret.
- **HealthCare@Home:** Det er navnet/betegnelsen for hjemmet, hvor patienten har bopæl. Det er i HealthCare@Home, at basestation står placeret.
- **Relatives:** ASP.NET applikation der giver pårørende mulighed for at se status for patienter relateret til dem.
- **Basestation:** Målestationen der er opstillet hos en patient
- **BLL:** Business Logic Layer
- **DAL:** Data Access Layer
- **DTO:** Data Transfer Object
- **BF-gruppen:** Jacob Brinth og Kenneth Fiil gruppe
- **SIH APUS:** Navnet på det projekt som BF-gruppen udvikler.
- **WCF:** Windows Communication Foundation
- **WPF:** Windows Presentation Foundation
- **AJAX:** Asynchronous JavaScript and XML
- **XHTML:** Extensible Hypertext Markup Language
- **CSS:** Cascading Style Sheet
- **LINQ:** Language Integrated Query
- **IIS:** Internet Information Services.

1.4 Dokumentstruktur og læsevejledning

4+1 modellen er anvendt. Den er bygget op over fire forskellige måder at se en use case på. Figuren herunder giver et overblik over hvordan 4+1 modellen kan fortolkes



Figur 1: 4+1 modellen

1.5 Dokumentets rolle i en iterativ udviklingsproces

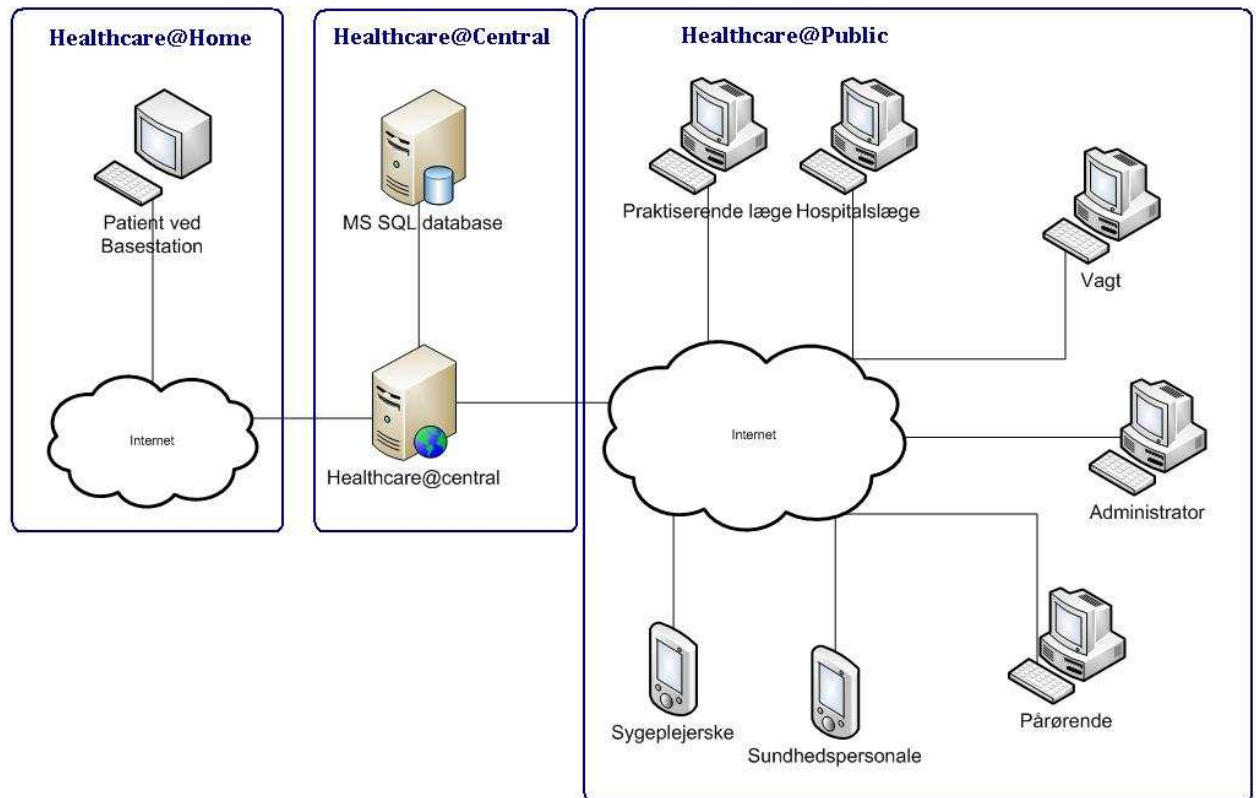
Udviklingen vil foregå gennem 4 iterationer, hvor der vil blive arbejdet iterativt med dokumentet. I iterationerne vil der blive arbejdet med:

- Udarbejdelse af use cases i kravspecifikation.
- Design af use cases.
 - Sekvensdiagrammer.
 - Klassebeskrivelser.
 - Klassediagrammer.
- Implementering af use cases.
- Deltest af implementering.
- Accepttest af implementering.
- Udarbejdelse af projektrapport.

I de følgende iterationer vil der blive arbejdet ved hjælp af de erfaringer som opnås igennem forløbet af de tidligere iterationer.

2. SYSTEM OVERSIGHT

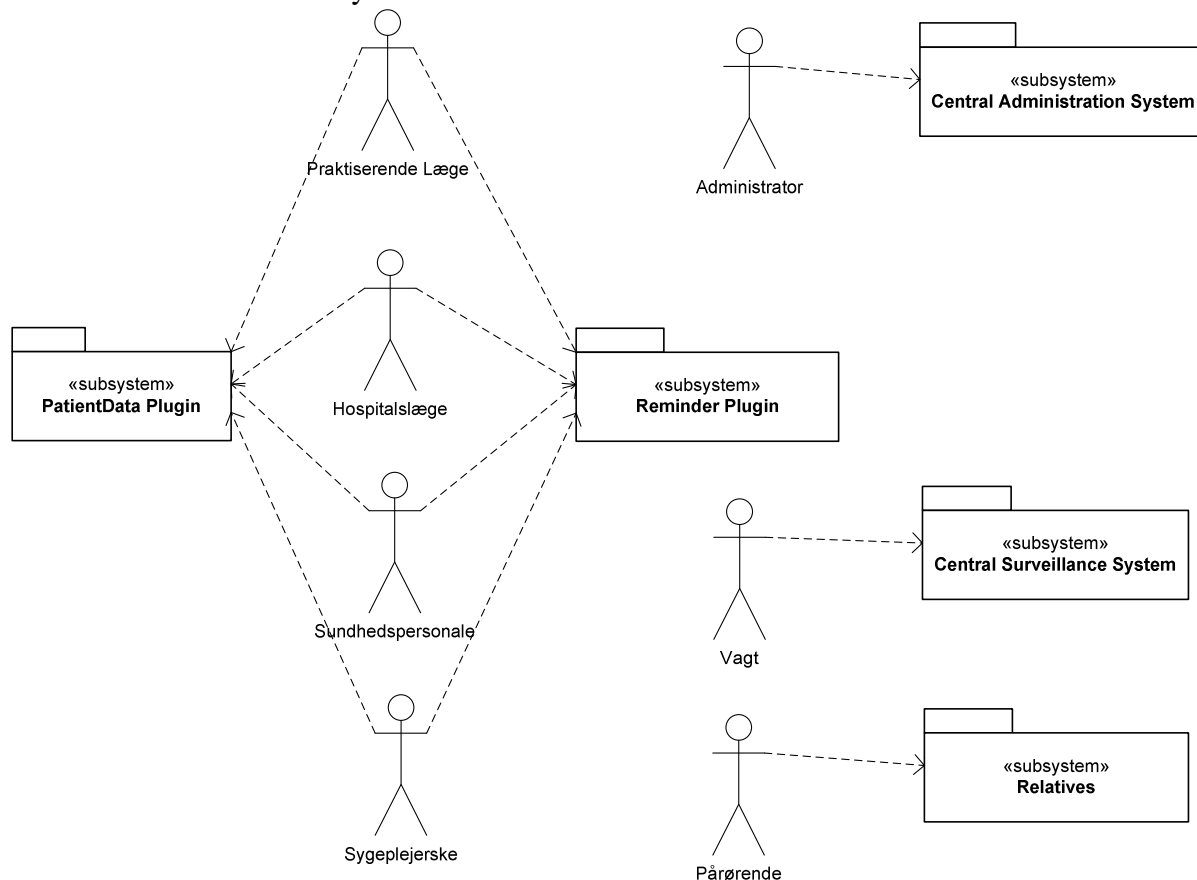
H@P består overordnet set af den kasse der har navnet Public. Dog vil der foretages en del kodning af WCF services som vil blive placeret på HealthCare@Central også.



Figur 2: Systemoversigt over hele Healthcare.

2.1 System kontekst

Se hvordan dette ser ud i System doc for HealthCare@Public.



Figur 3: Aktør kontekstdiagram

2.2 System introduktion

Hovedformålet med H@P er at gøre, informationer opsamlet hos en Patient, tilgængelige for praktiserende læger, hospitalslæger, sundhedsassistent, sygeplejersker og pårørende. Det skal også være muligt at sætte alarmniveauer for de målinger der foretages på patienten. Disse alarmniveauer skal så kunne håndteres af en vagt. Det tredje mål med systemet er at der skal kunne oprettes påmindelser for patienten så denne kan huske at tage medicin, måle blodtryk, næste hjemmeplejerbesøge og så videre. Systemets sidste mål er at tilbyde central administration på en generisk og brugervenlig måde.

3. SYSTEMETS GRÆNSEFLADER

3.1 Grænseflader til person aktører

H@P grænser op mod Administrator, Praktiserende læge, Hospitalslæge, Pårørende, Sygeplejerske, Hjemmeplejer og Vagt.

3.1.1 Administrator

Aktøren Administrator er den daglige administrator af H@P.

Typisk administrator opgaver:

- Opret, rediger og slet patienter samt andre personer i systemet.
- Tilknyt en patient til en læge.
- Supportere læger og andre brugeren af systemet.
- Håndtere person- og udstyrstyper.

3.1.2 Praktiserende læge

Aktøren Praktiserende læge anvender H@P til at se påmindelser og nøgledata for sine tilknyttede patienter.

Typisk Praktiserende læge opgaver:

- Se nøgledata.
- Redigere og tilføje påmindelser til patienten.

3.1.3 Hospitalslæge

Aktøren Hospitalslæge anvender H@P til at se påmindelser og nøgledata for sine tilknyttede patienter, typisk imens disse er indlagt.

Typisk hospitalslæge opgaver:

- Se nøgledata.
- Redigere og tilføje påmindelser til patienten.

3.1.4 Sygeplejerske

Aktøren Sygeplejerske anvender primært H@P i forbindelse med besøg hjemme hos patienten.

Typisk sygeplejerske opgaver:

- Opret påmindelser omkring indtag af medicin.
- Ændringer i oplysninger omkring medicin (typisk en instruks fra læge).
- Se medicin oplysninger.

3.1.5 Sundhedspersonale

Aktøren Sundhedspersonale anvender primært H@P i forbindelse med besøg hjemme hos patienten.

Typisk Sundhedspersonale opgaver:

- Se nøgledata
- Opret påmindelser omkring indtag af medicin.

- Se medicin oplysninger.

3.1.6 Pårørende

Aktøren Pårørende er typisk interesseret i at vide hvis der sker noget med den patient de er pårørende til. Derfor vil de kunne se disse data via en web applikation.

Typisk pårørende opgaver:

- Se seneste målinger og alarm for den relaterede patient

3.1.7 Vagt

Aktøren Vagt er den person der overvåger alle alarmer der kommer fra patienter i det område som han er tilknyttet. Vagten kunne også være en Sygeplejerske.

Typisk vagt opgaver:

- Håndtering af de alarmer der kommer.
- Kontakt til Falck, Pårørende og hjemmeplejen.

3.2 Grænseflader til eksterne system aktører

Kommunikationen til databasen som indeholder de persistente data foregår over et netværk.

3.3 Grænseflader til hardware aktører

Navn	Kommunikations Type	Bemærkninger
Database server	Ethernet	Tilgangen til databaseserveren foregår via netværk

3.4 Grænseflader til eksterne software aktører

3.4.1 SIH Presentation Framework

Subsystemerne Se PatientData og Reminders er begge implementeret som plugins til den eksterne software applikation SIH Presentation Framework udviklet af BF-Gruppen. Der er defineret interfaces til plugins og usercontrols der implementeres af os.

4. USE CASE VIEW

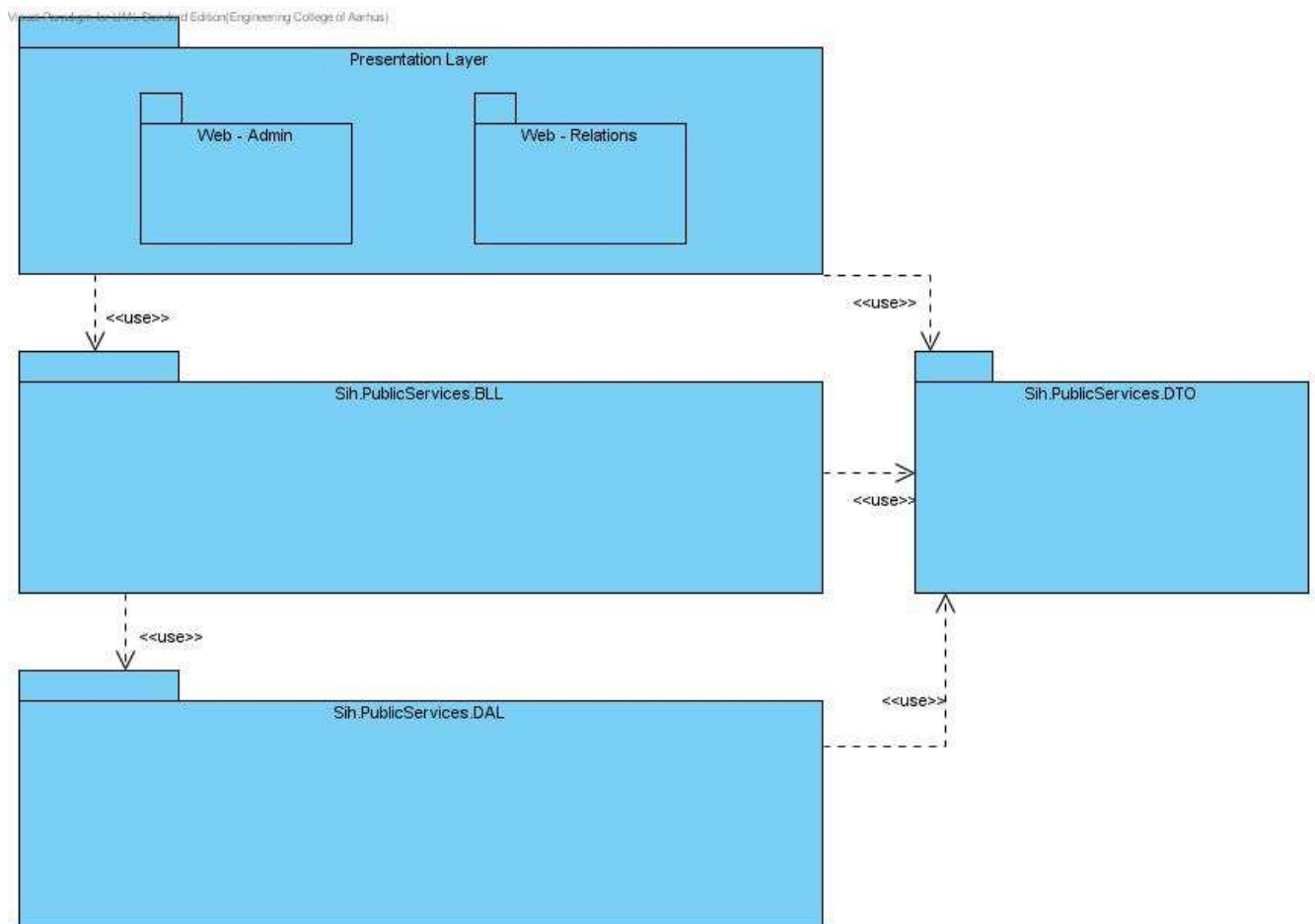
4.1 Oversigt over arkitektursignifikante Use Cases

Se Kravspecifikation for HealthCare@Public [1].

5. LOGISK VIEW

5.1 Oversigt

5.1.1 Subsystem – Central Administration System + Relatives



Figur 4: Package diagram for Subsystem – Administrator + Relatives.

5.1.2 Subsystem PatientData Plugin

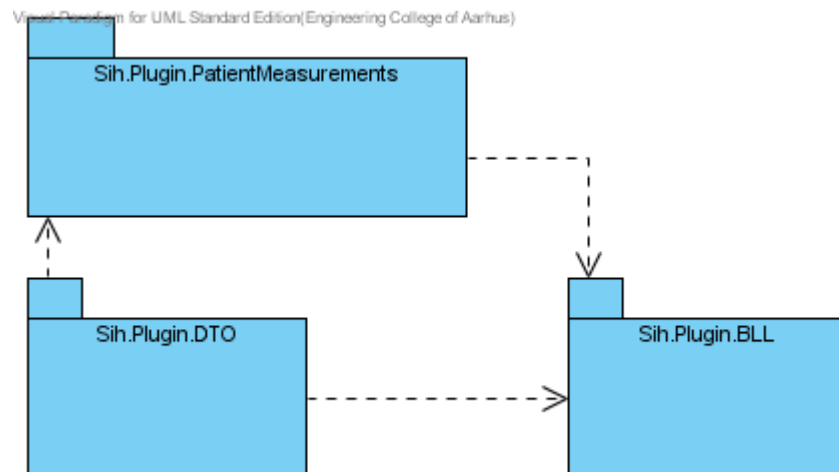


Figure 5: Package diagram for Subsystem – PatientData Plugin

5.1.3 Subsystem Reminder Plugin

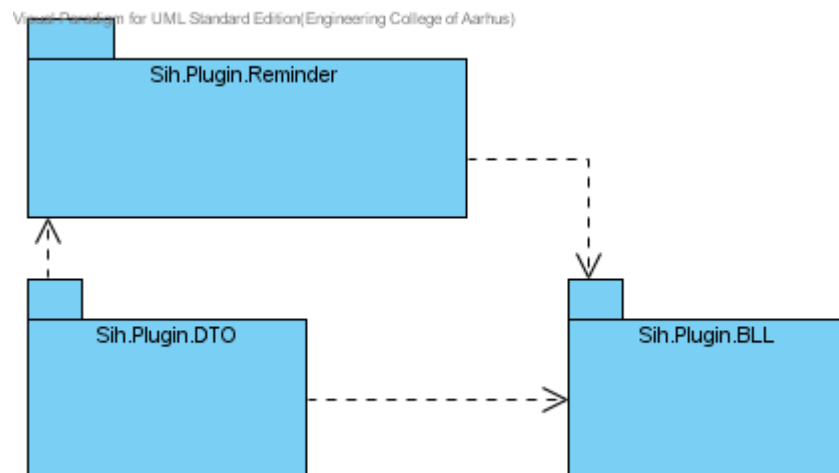


Figure 6: Package diagram for Subsystem – Reminder Plugin

5.1.4 Subsystem – Central Surveillance System

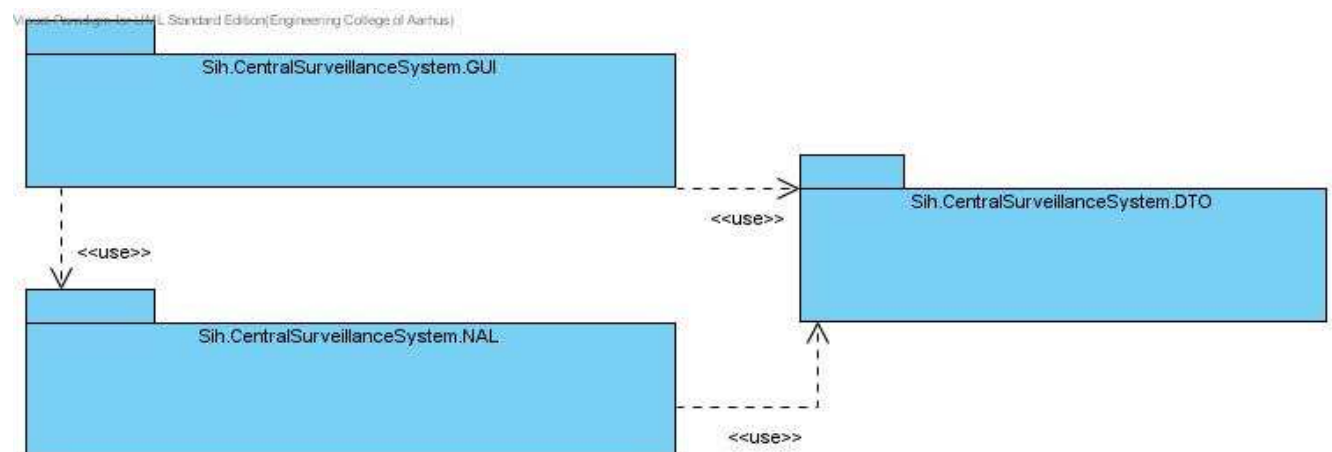
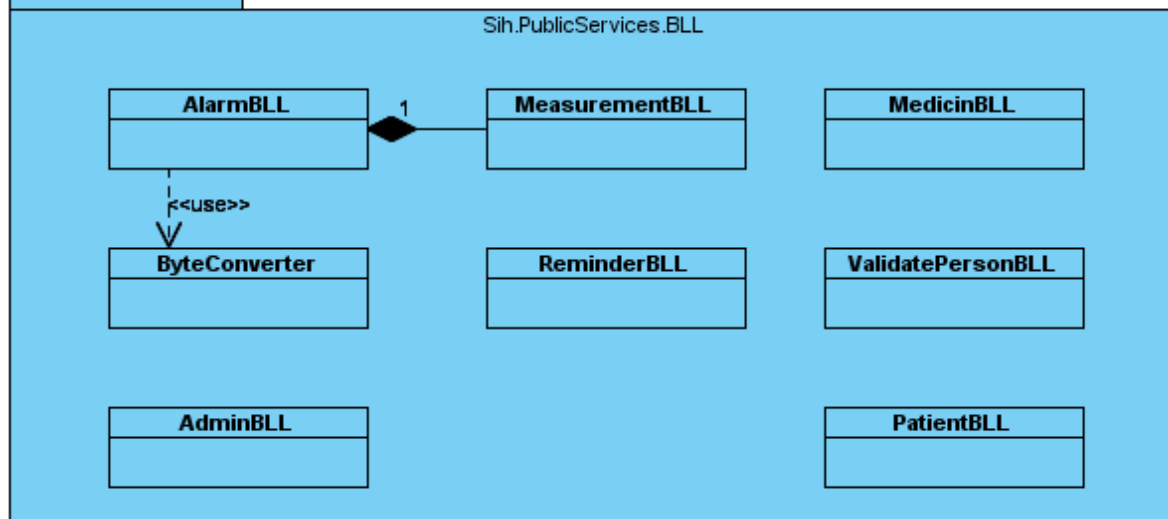


Figure 7: Package diagram for Subsystem – Central Surveillance System

5.2 Arkitektursignifikante designpakker

5.2.1 Sih. PublicServices.BLL

Visual Paradigm for UML: Standard Edition (Engineering College of Aarhus)



Figur 8: Klassesdiagram for Sih.PublicServices.BLL

Dette lag indeholder alt business logik for SIH Public Services. Denne pakke bruges også af en ASP.NET applikation der bruges til administration.

5.2.1.1 ByteConverter

Static klasse der formaterer den rå bytestream af data fra databasen om til det format den er blevet lagt ned i og vice versa.

5.2.1.2 AlarmBLL

Denne klasse håndterer alt logik omkring alarmer.

5.2.1.3 AdminBLL

Denne klasse er en stor samling af funktioner til administrative opgaver i forbindelse med Sih.

5.2.1.4 MeasurementBLL

Denne klasse gør det muligt at indrapporterer en måling.

5.2.1.5 ReminderBLL

Denne klasse indeholder logik for håndtering af påmindelser.

5.2.1.6 MedicinBLL

Denne klasse indeholder logik for håndtering af medicinering af patienter.

5.2.1.7 ValidatePersonBLL

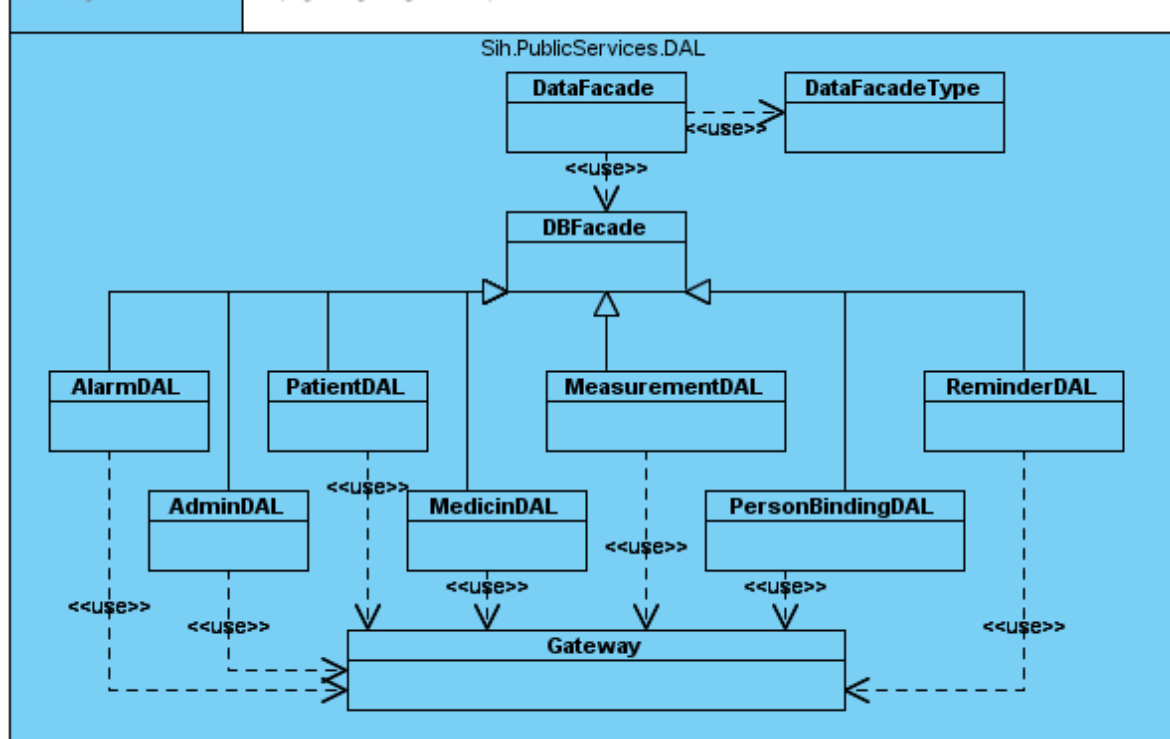
Denne klasse indeholder logik for validering af en person.

5.2.1.8 PatientBLL

Denne klasse indeholder logik for håndtering af patienter. Herunder også at få lister af tilknyttede patienter ud. Det er også denne klasse der bruges til at få måledata ud for en patient.

5.2.2 Sih. PublicServices.Dal

Visual Paradigm for UML, Standard Edition (Engineering College of Aarhus)



Figur 9: Klassesdiagram for Sih.PublicServices.Dal

Dette lag indeholder alt data access. Det er lavet med et Factory pattern kombineret med et Facade pattern. Til kommunikation med databasen er der anvendt et Gateway pattern.

5.2.2.1 DataFacade

Factory klasse der gør det muligt at få returneret en DAL klasse af typen DBFacade

5.2.2.2 DataFacadeType

De typer det er muligt at få returneret ved hjælp af vores factory.

5.2.2.3 DBFacade

Basisklasse for alle DAL klasser.

5.2.2.4 AlarmDAL

Indeholder alt kommunikation med databasen i forbindelse med alarmer.

5.2.2.5 AdminDAL

Indeholder alt kommunikation med databasen i forbindelse med administration.

5.2.2.6 PatientDAL

Indeholder alt kommunikation med databasen i forbindelse med patienter.

5.2.2.7 MedicinDAL

Indeholder alt kommunikation med databasen i forbindelse med medicinering.

5.2.2.8 MeasurementDAL

Indeholder alt kommunikation med databasen i forbindelse med at lagre målinger.

5.2.2.9 PersonBindingDAL

Indeholder alt kommunikation med databasen i forbindelse med sammenbinding af personer og udstyr.

5.2.2.10 ReminderDAL

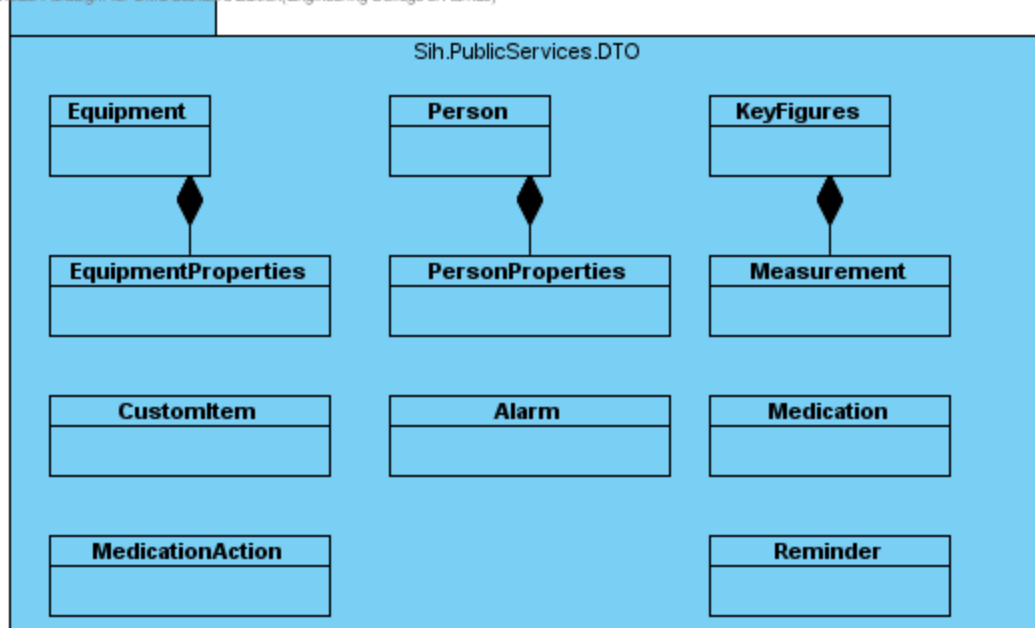
Indeholder alt kommunikation med databasen i forbindelse med påmindelser.

5.2.2.11 Gateway

Denne statiske klasse wrapper ExecuteQuery, ExecuteNonQuery og diverse andre basis funktioner.

5.2.3 Sih. PublicServices.Dto

Visual Programming for UML: Standard Edition (Engineering College of Aarhus)



Figur 10: Klassediagram for Sih.PublicServices.Dto

Dette lag indeholder alle data transfer objekter som bliver anvendt i HealthCare@Central.

5.2.3.1 Person

DTO til en Person. Dette DTO har et array af PersonProperties DTO.

5.2.3.2 PersonProperties

DTO til en person egenskab. Det kan være Name = "FirstName" og Value = "Mathias".

5.2.3.3 Equipment

DTO til et stykke udstyr. Dette DTO har et array af EquipmentProperties DTO'er.

5.2.3.4 EquipmentProperties

DTO til en udstyrsegenskab. Det kan være Name = "MAC" og Value = "{00:00:00:a1}"

5.2.3.5 KeyFigures

DTO til nøgletal for en patient. Dette DTO indeholder et array af Measurement DTO'er.

5.2.3.6 Measurement

DTO til målinger.

5.2.3.7 CostumItem

DTO til simple data. Typisk hvis disse data normalt skal loades i en DropDownList og man kun har brug for en displaymember og en valuemember.

5.2.3.8 Alarm

DTO der kan indeholde en alarm.

5.2.3.9 Reminder

DTO der kan indeholde en påmindelse.

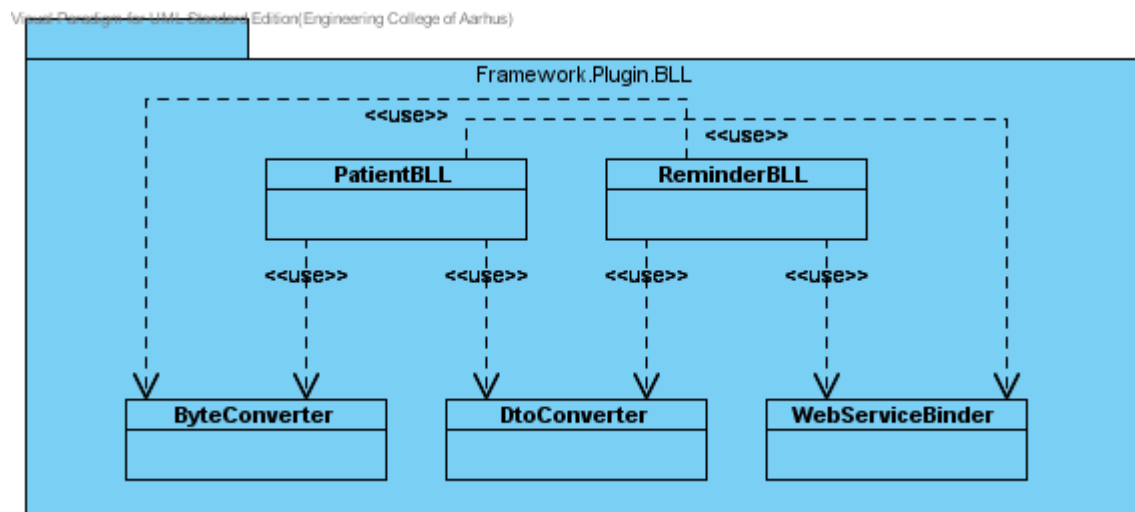
5.2.3.10 Medication

DTO der kan indeholde information om et medikament til en patient.

5.2.3.11 MedicationAction

DTO der kan indeholde et action tilknyttet et medikament. Det kan være at denne medicin er preudleveret tirsdag middag.

5.2.1 Framework.Plugin.BLL



Figur 11: Klassediagram for Framework.Plugin.BLL

Dette lag indeholder den logik der bruges i de plugins der bliver udviklet til SIH Presentation Framework.

5.2.1.1 PatientBLL

Logik klasse der opretter forbindelse til, håndterer og wrapper funktioner fra WCF Servicen PatientService, der er nærmere beskrevet i systemarkitekturdokument for Sih.PublicServices.

5.2.1.2 ReminderBLL

Logik klasse der opretter forbindelse til, håndterer og wrapper funktioner fra WCF Servicen ReminderService der er nærmere beskrevet i systemarkitekturdokument for Sih.PublicServices[3].

5.2.1.3 ByteConverter (statisk)

Logik klasse der konverterer en rå måling fra databasen om til et array af strenge man så kan caste til det format man forventer den pågældende måling i. Klassen fungerer også modsat.

5.2.1.4 DtoConverter (statisk)

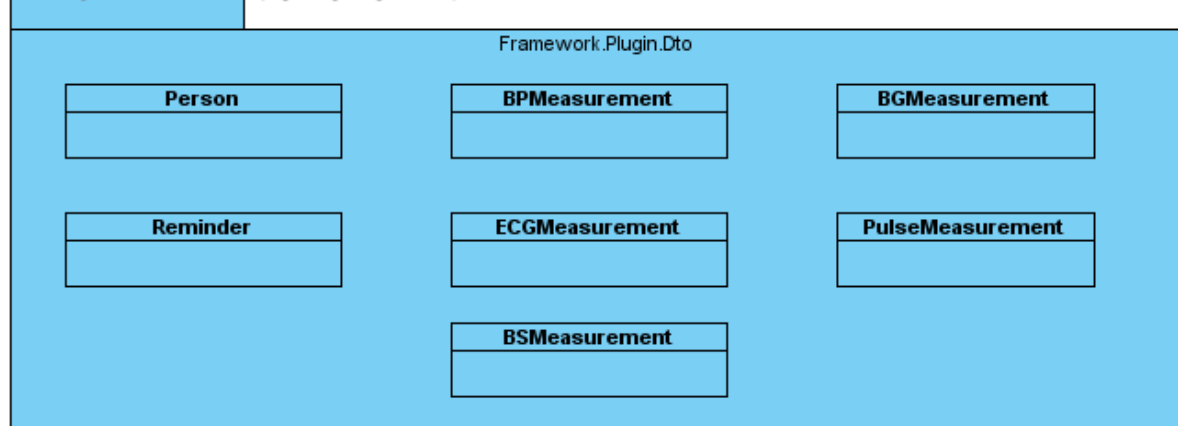
Når der modtages en måling via webservicen er det som et bytearray. Ved hjælp af ByteConverter klassen bliver denne byte streng delt op. Har man eksempelvis en blodtryksmåling som er gemt som "120 / 80", så har byteConverter givet os et string[] med 120 på plads 0 og 80 på plads 1. Disse to tal bliver så via DtoConverter lagt i en BPMeasurement(DTO for en blodtryksmåling). Grunden til dette er at BPMeasurement senere hen kan bruges direkte som en data source i GUI controllerne.

5.2.1.5 WebServiceBinder (statisk)

Erstatter det der normalt bliver gjort i App.Config filen ved binding til en webservice. Den beskriver binding og endpoint dynamisk.

5.2.2 Framework.Plugin.DTO

Visual Paradigm for UML - Standard Edition (Engineering College of Aarhus)



Figur 12: Klassediagram for Framework.Plugin.BLL

Dette lag indeholder en række data transfer objects der også benyttes som datasources til GUI controllerne.

5.2.2.1 Person

Kan indeholde basis informationer om en person.

5.2.2.2 Reminder

Kan indeholde data omkring en påmindelse for en given person.

5.2.2.3 BPMeasurement

Kan indeholde en blodtryksmåling.

5.2.2.4 ECGMeasurement

Kan indeholde en EKG måling.

5.2.2.5 BSMeasurement

Kan indeholde en blodsuktermåling.

5.2.2.6 BGMeasurement

Kan indeholde en blodgasmåling.

5.2.2.7 PulseMeasurement

Kan indeholde en pulsmåling.

5.3 Use Case realiseringer

Til dokumentation af vores use case realiseringen er der tegnet sekvensdiagrammer og et klassediagram til hver af use case'ene.

Klassediagrammerne er tegnet således at de viser use case realiseringen med udgangspunkt i den - eller de - specifikke klasser som direkte relatere sig til use casen. Der er kun vist første led til andre klasser som den - eller de - klasser har forbindelser til. Der bliver ikke vist hvad de efterfølgende klasser har associationer til.

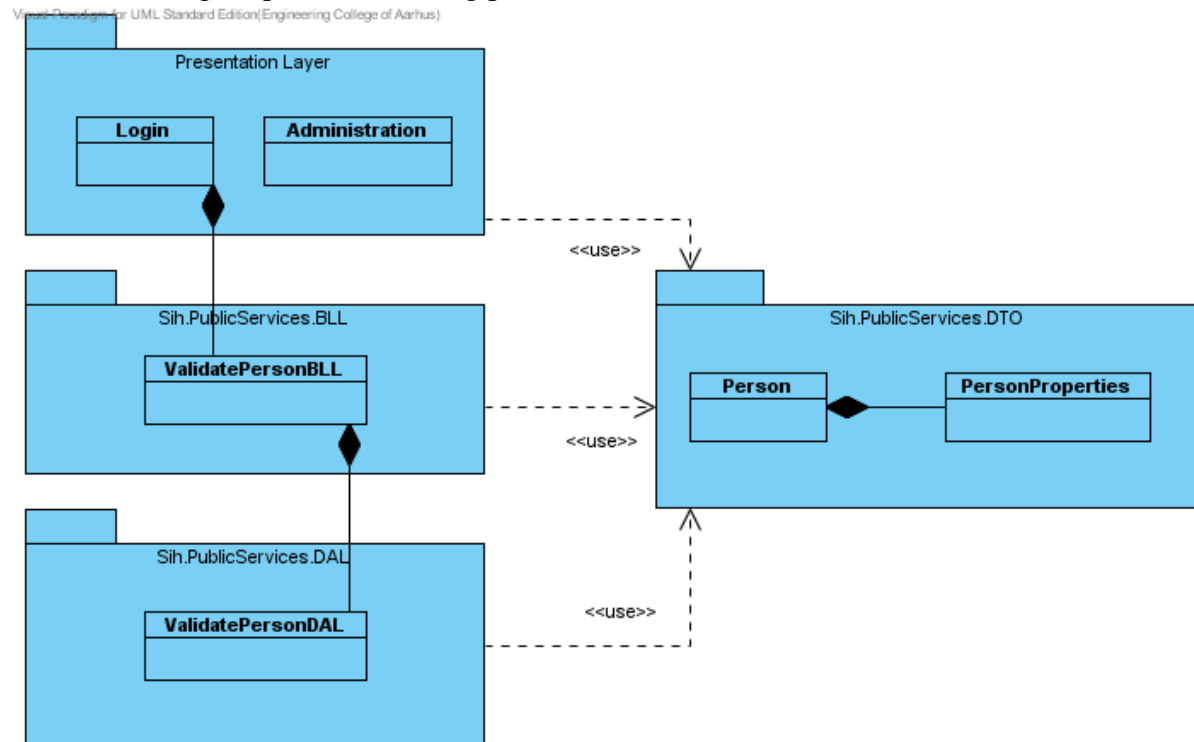
Sekvens diagrammerne tegnes så vidt muligt efter UML 2.0 standarden. Dog har det ikke alle steder været muligt at tegne sekvens diagrammerne for use case realiseringen fornuftigt og samtidig overholde standarden.

For at simplificere diagrammerne så det er muligt at bevare overblikket, er nogle af kaldene blevet udeladt, da det ville give nogle håbløse diagrammer.

5.3.1 Subsystem – Administrator

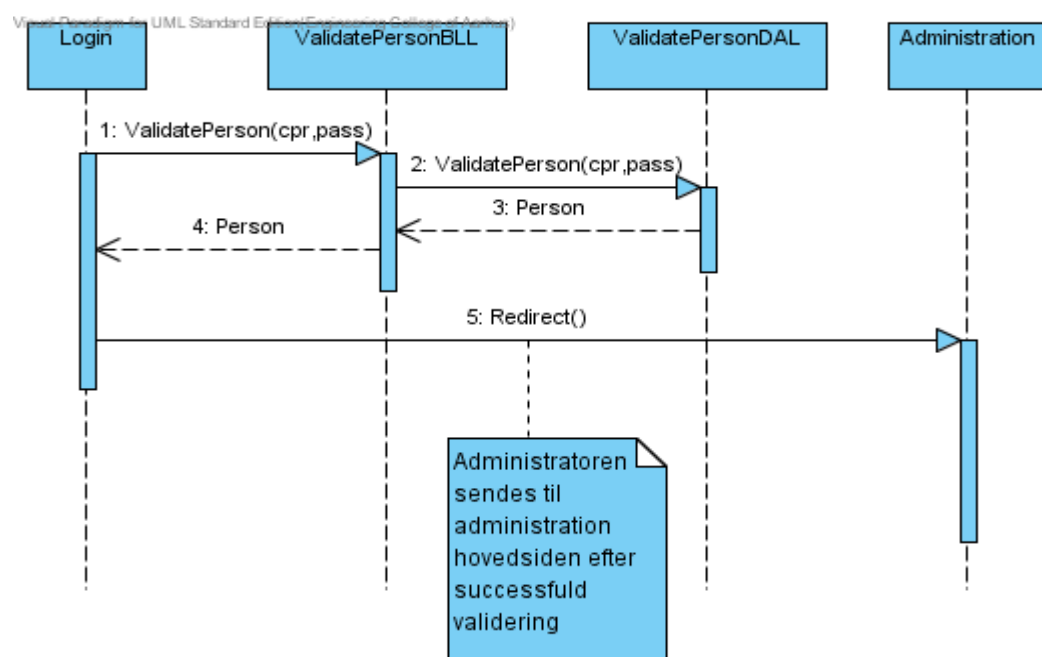
5.3.1.1 Use case 1. Login

Denne use case indeholder en ASP.NET side der kalder ned i business logik laget for at validere et brugerinput af CPR nr og password.



Figur 13: Klassediagram for use case realisering af use case 1

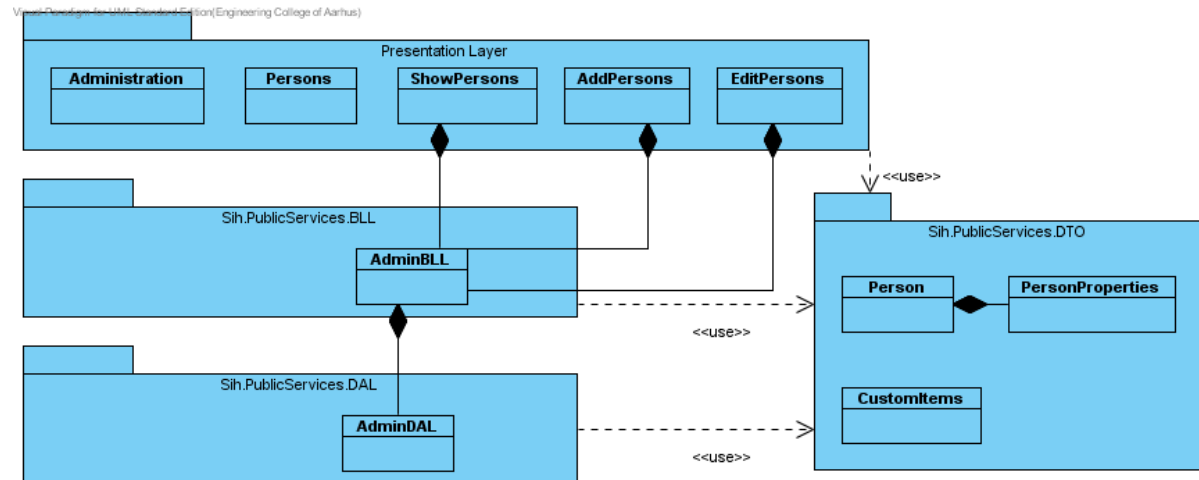
5.3.1.1.1 Normalforløb for use case 1:



Figur 14: Sekvensdiagram for use case realisering af use case 1.

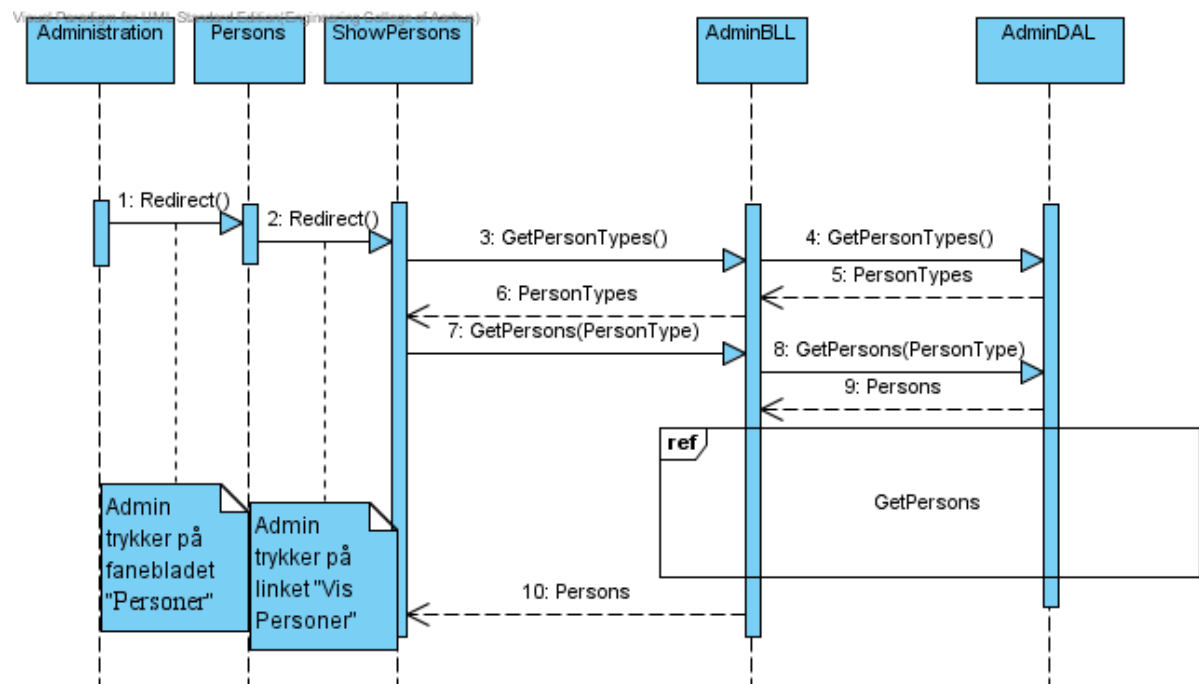
5.3.1.2 Use case 2. Håndter Person

Denne use case indeholder en asp.net side der kalder ned i business logik laget for at hente, tilføje, ændre eller slette en Person i systemet.



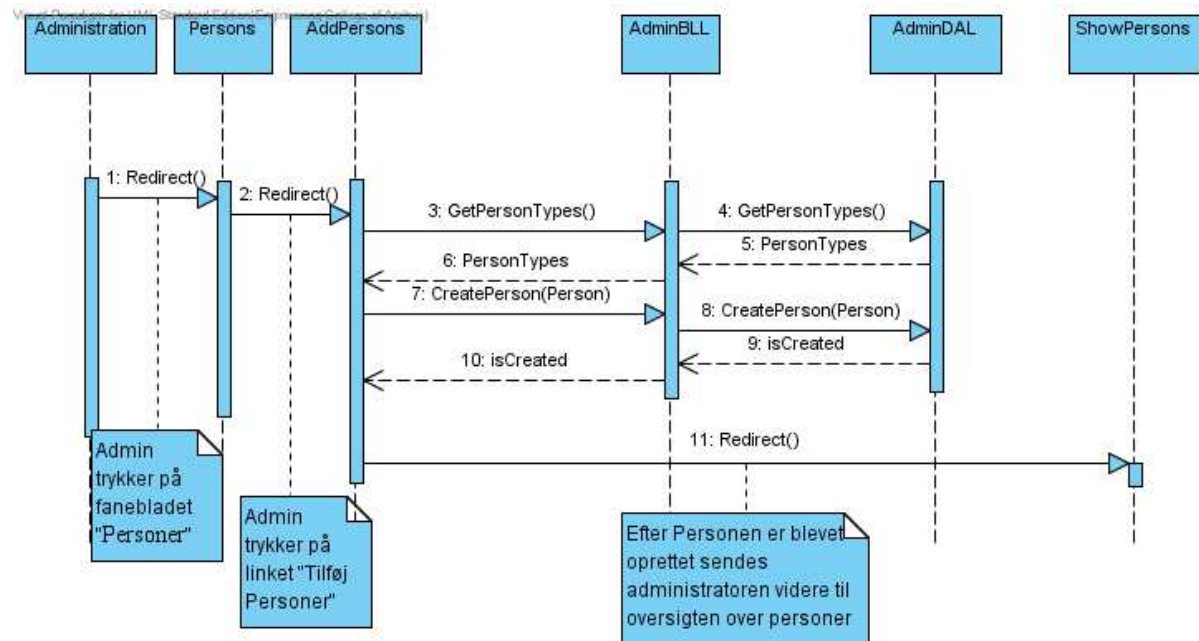
Figur 15: Klassediagram for use case realisering af use case 2

5.3.1.2.1 Normalforløb for use case 2:



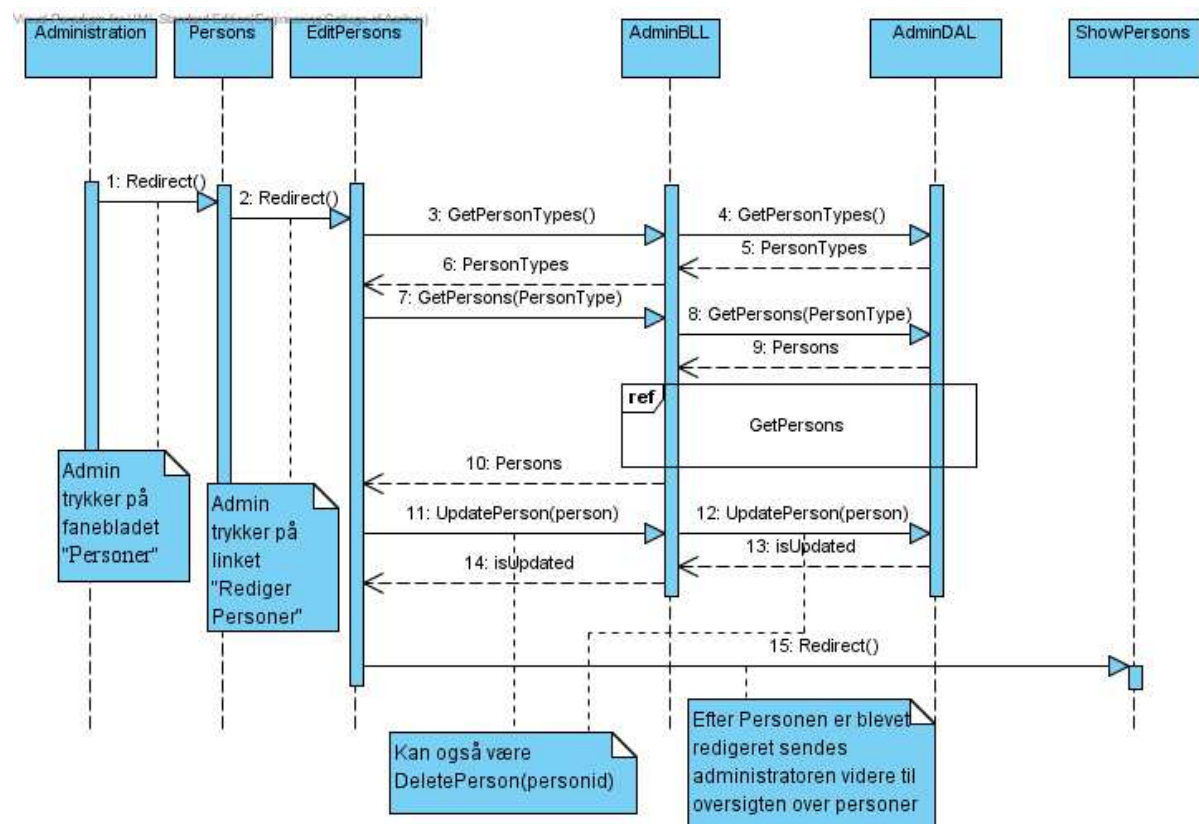
Figur 16: Sekvensdiagram for use case realisering af use case 2 - normalforløb.

5.3.1.2.2 Alternativt forløb for use case 2 – Tilføj Person



Figur 17: Sekvensdiagram for use case realisering af use case 2 – Alternativforløb(Tilføj).

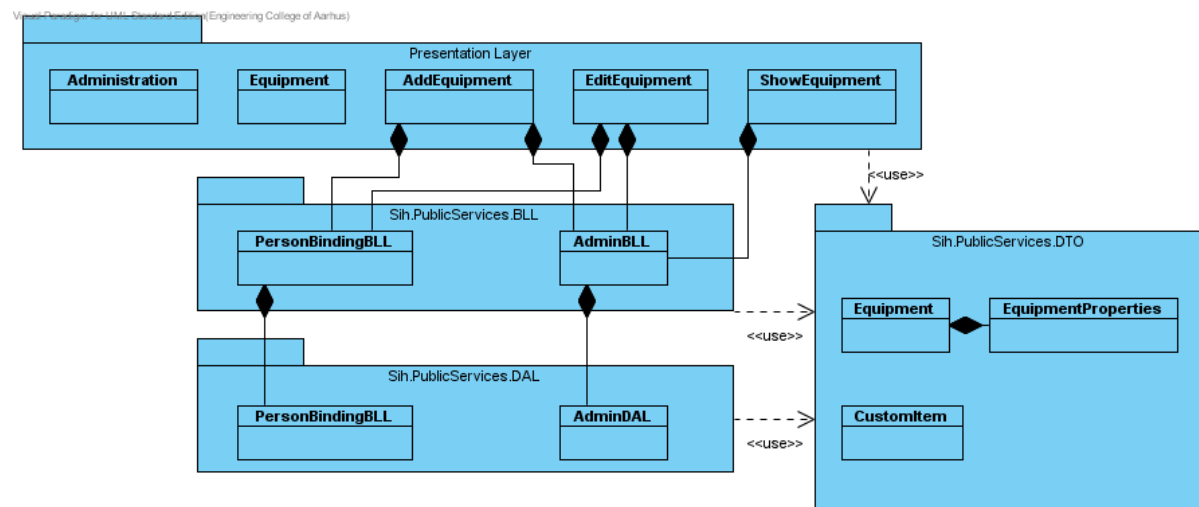
5.3.1.2.3 Alternativt forløb for use case 2 – Rediger/Slet Person



Figur 18: Sekvensdiagram for use case realisering af use case 2 – Alternativforløb(Rediger/slet).

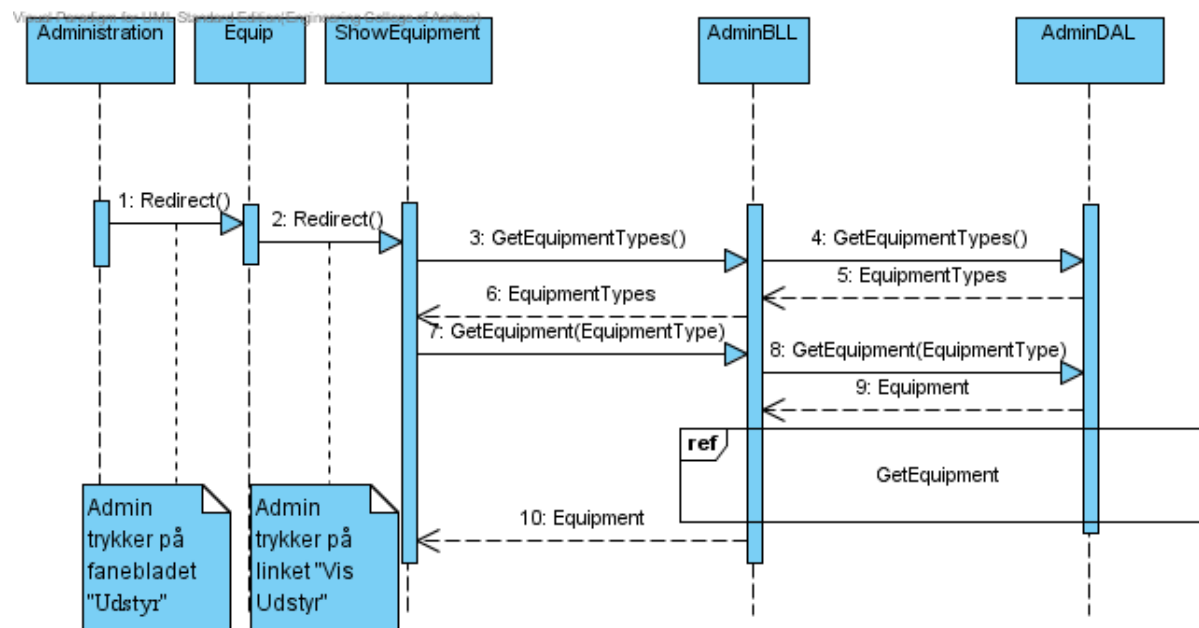
5.3.1.3 Use case 3. Håndter Udstyr

Denne use case indeholder en asp.net side der kalder ned i business logik laget for at hente, tilføje, ændre eller slette et stykke udstyr i systemet.



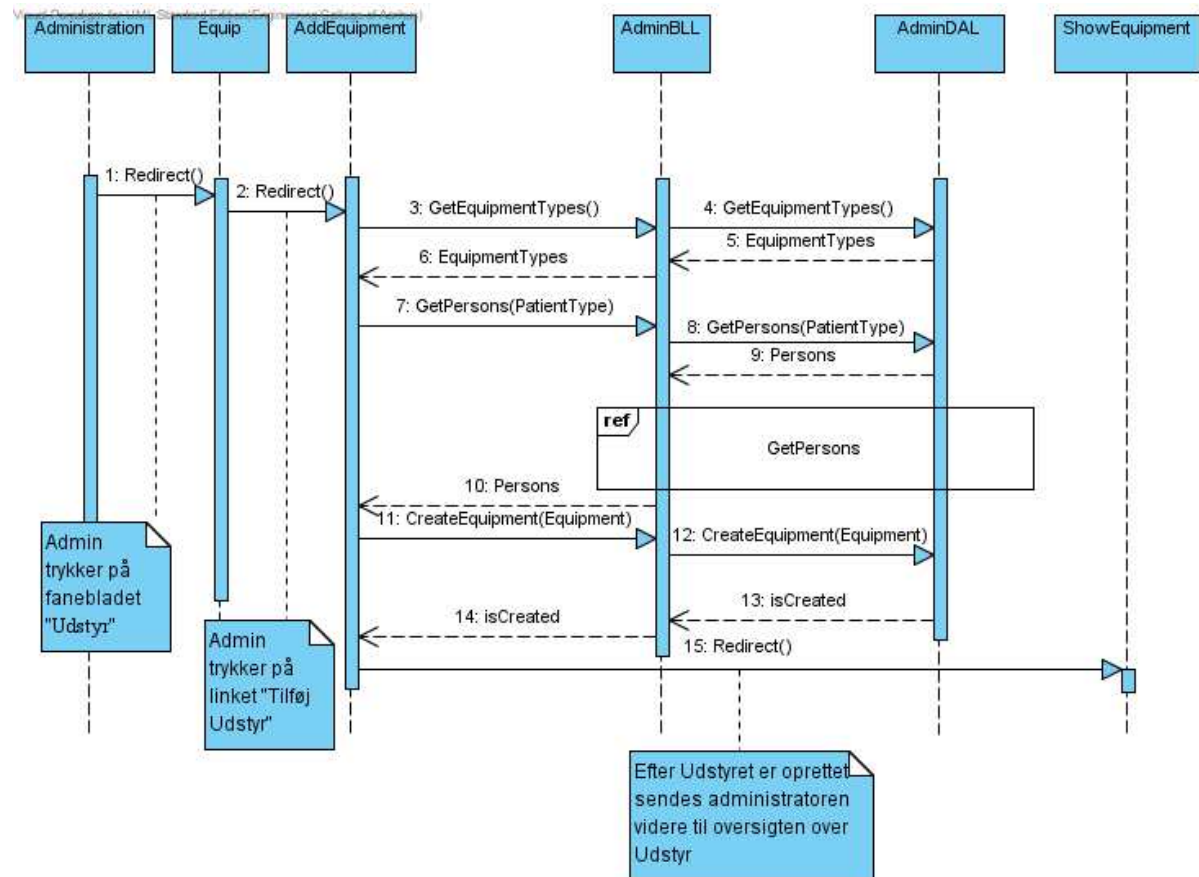
Figur 19: Klassediagram for use case realisering af use case 3

5.3.1.3.1 Normalforløb for use case 3:



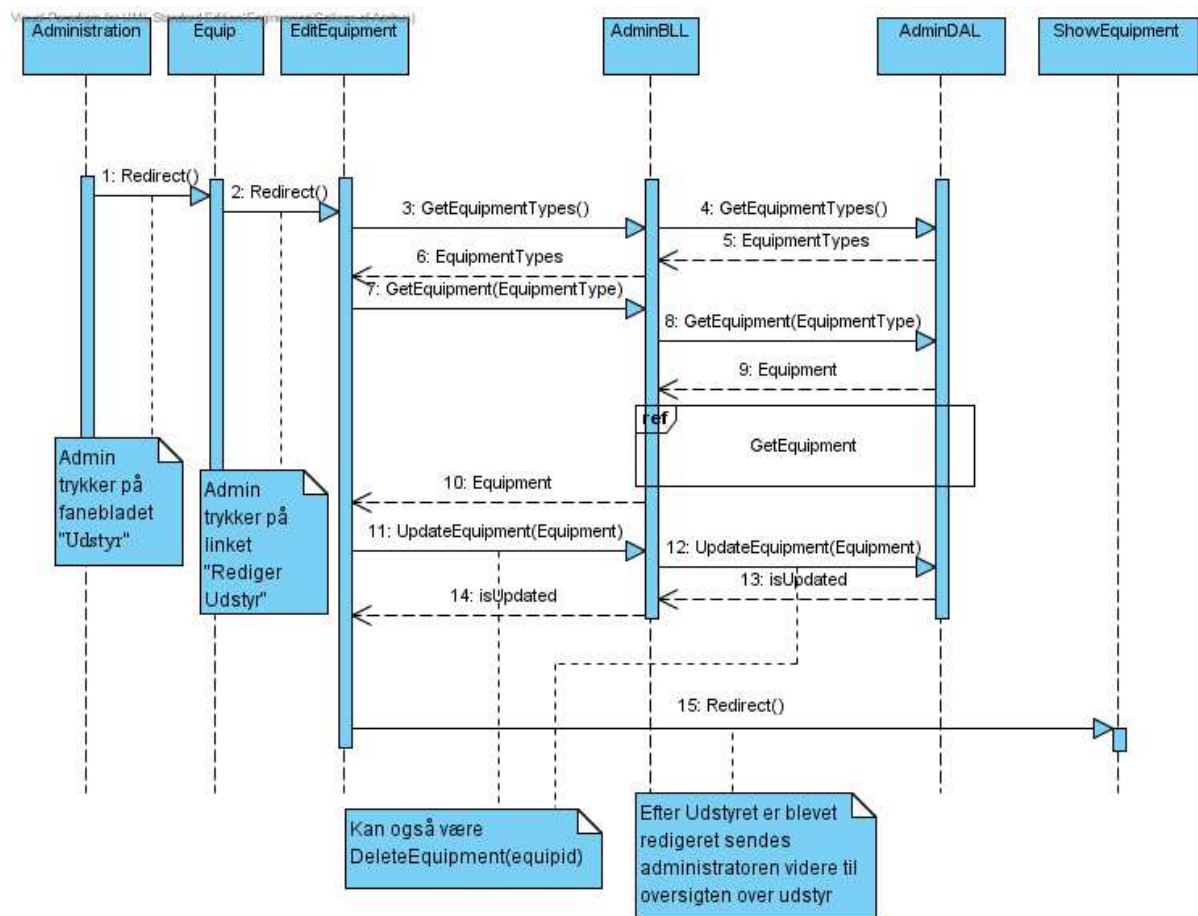
Figur 20: Sekvensdiagram for use case realisering af use case 3 - normalforløb.

5.3.1.3.2 Alternativt forløb: Tilføj udstyr



Figur 21: Sekvensdiagram for use case realisering af use case 3 – Alternativforløb(Tilføj).

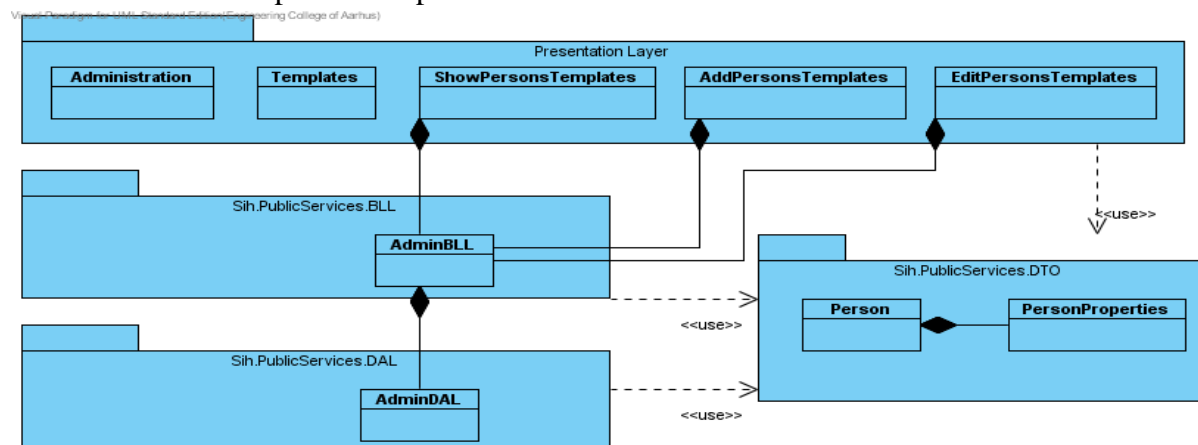
5.3.1.3.3 Alternativt forløb: rediger/slet udstyr



Figur 22: Sekvensdiagram for use case realisering af use case 3 – Alternativforløb(Rediger/slet).

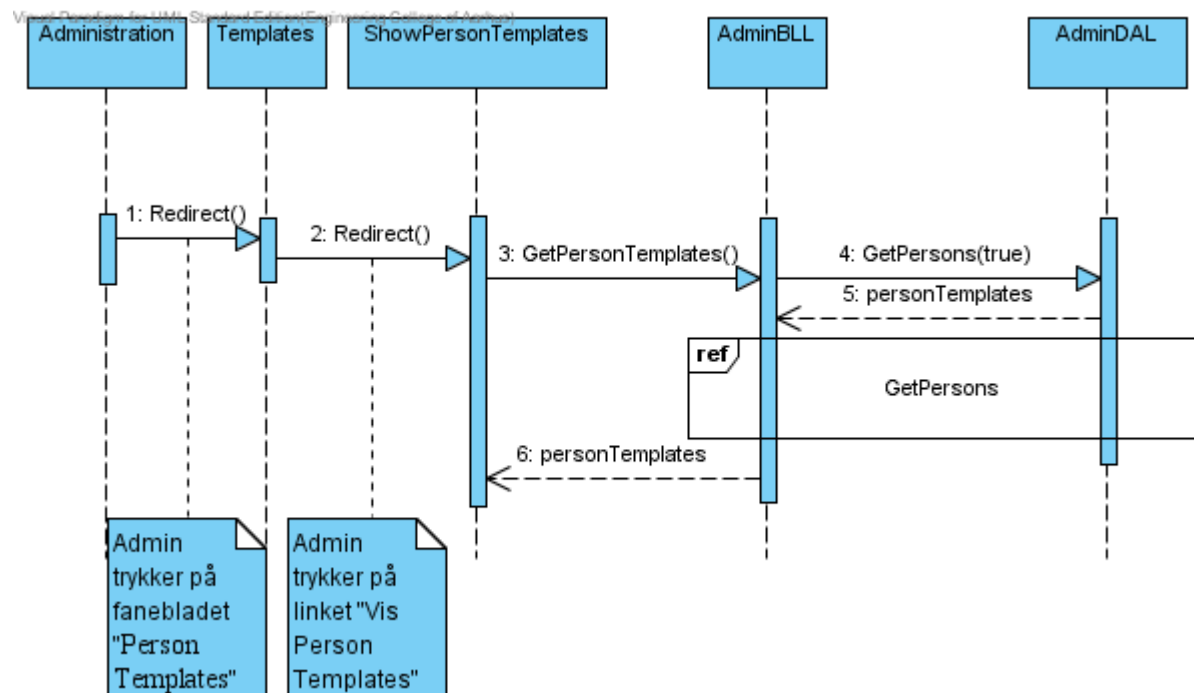
5.3.1.4 Use case 4. Håndter Person Templates

Denne use case indeholder en asp.net side der kalder ned i business logik laget for at hente og vise de eksisterende person templates.



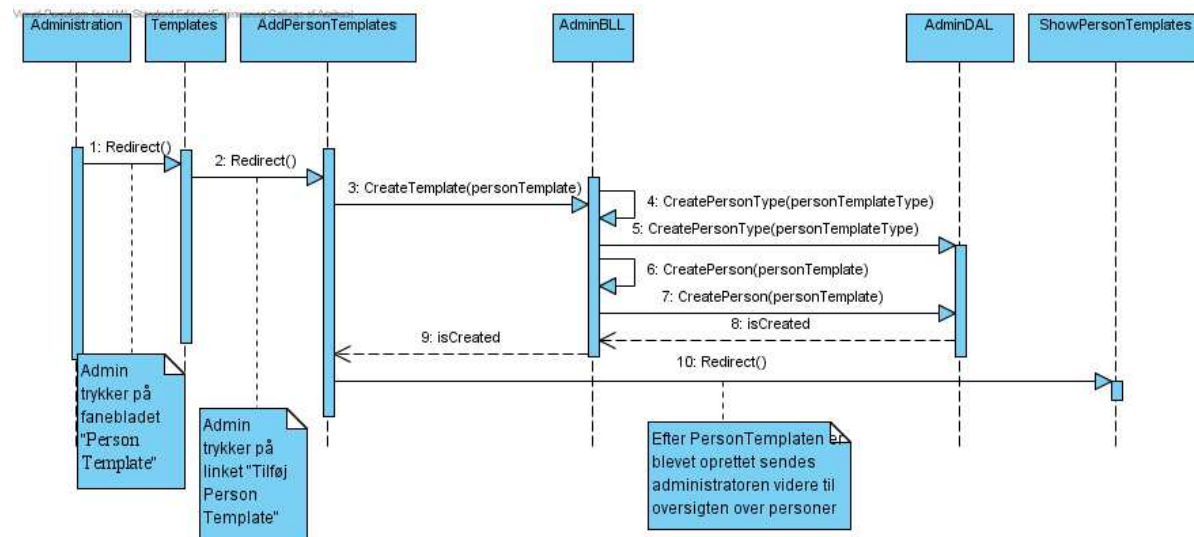
Figur 23: Klassediagram for use case realisering af use case 4

5.3.1.4.1 Normalforløb for use case 4:



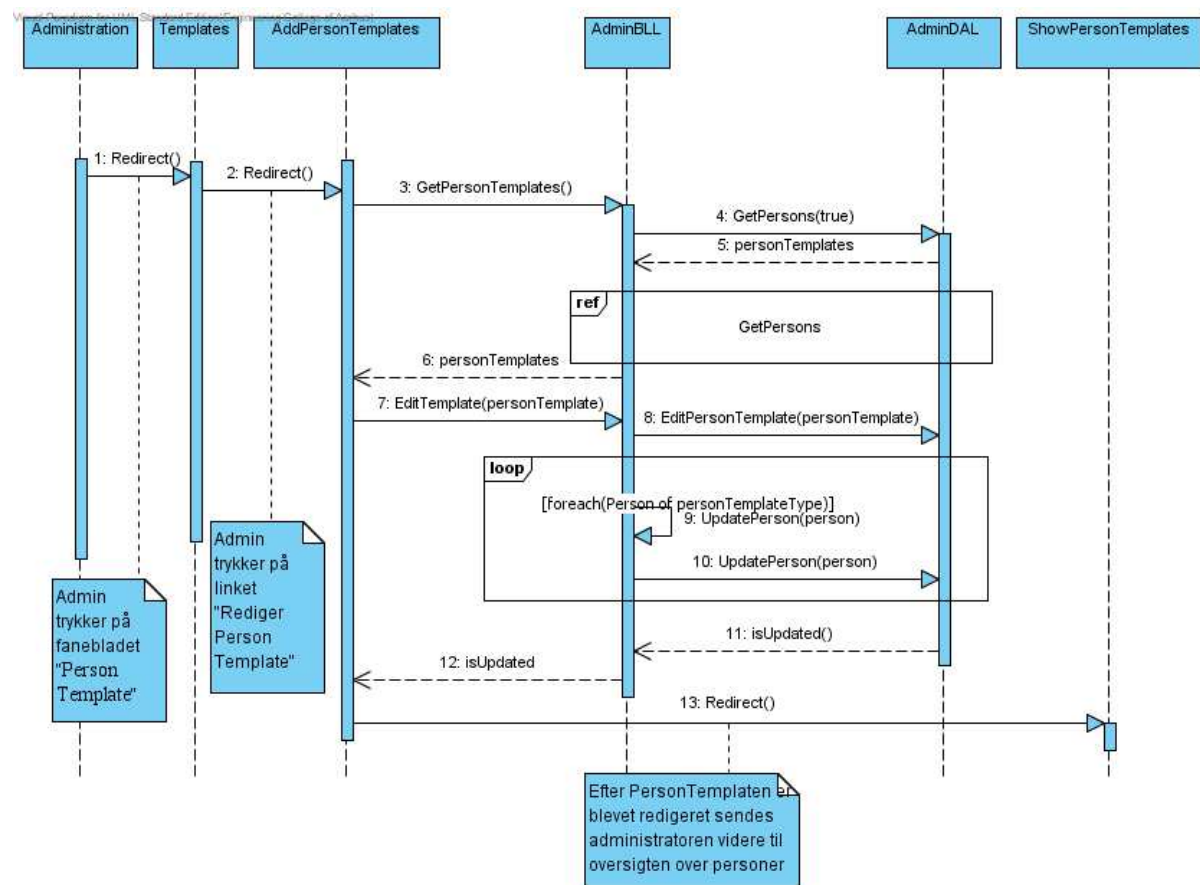
Figur 24: Sekvensdiagram for use case realisering af use case 4 - normalforløb.

5.3.1.4.2 Alternativt forløb - tilføj person template(type):



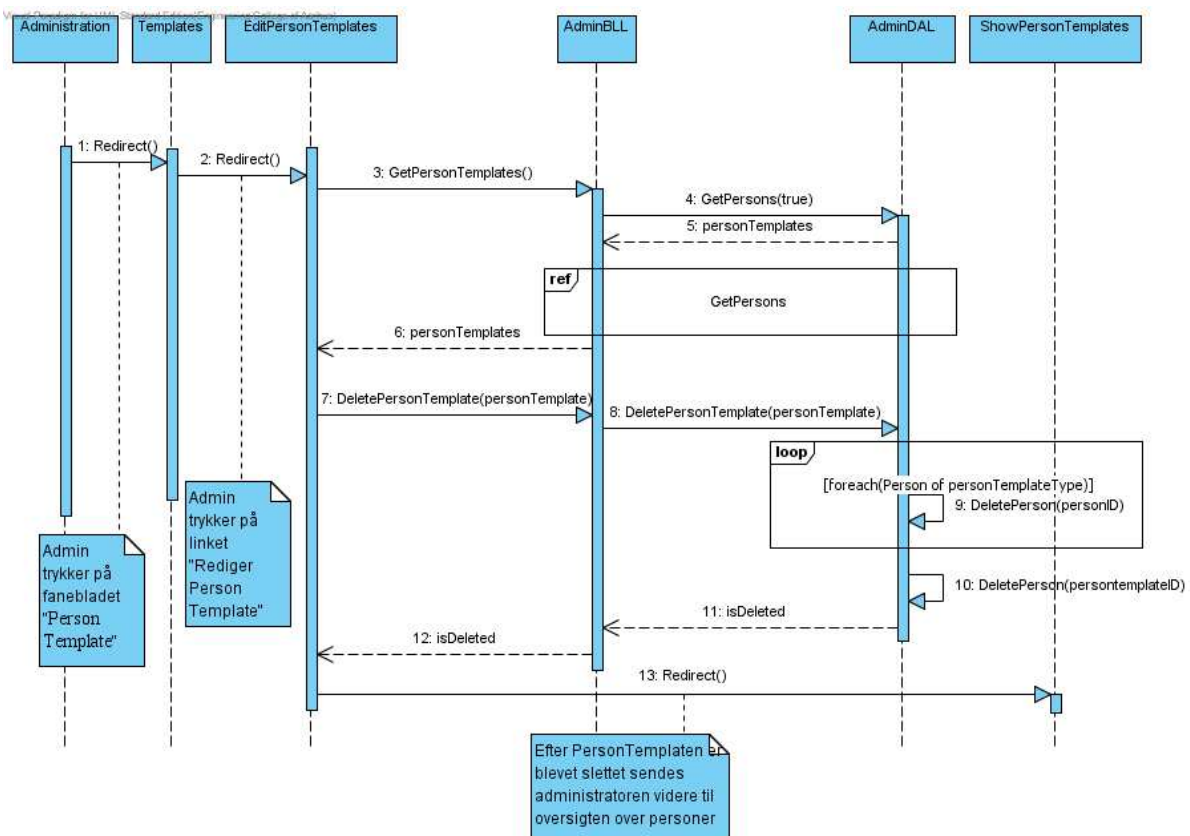
Figur 25: Sekvensdiagram for use case realisering af use case 4 – Alternativforløb(Tilføj).

5.3.1.4.3 Alternativt forløb – Rediger en person template(type)



Figur 26: Sekvensdiagram for use case realisering af use case 4 – Alternativforløb(Rediger).

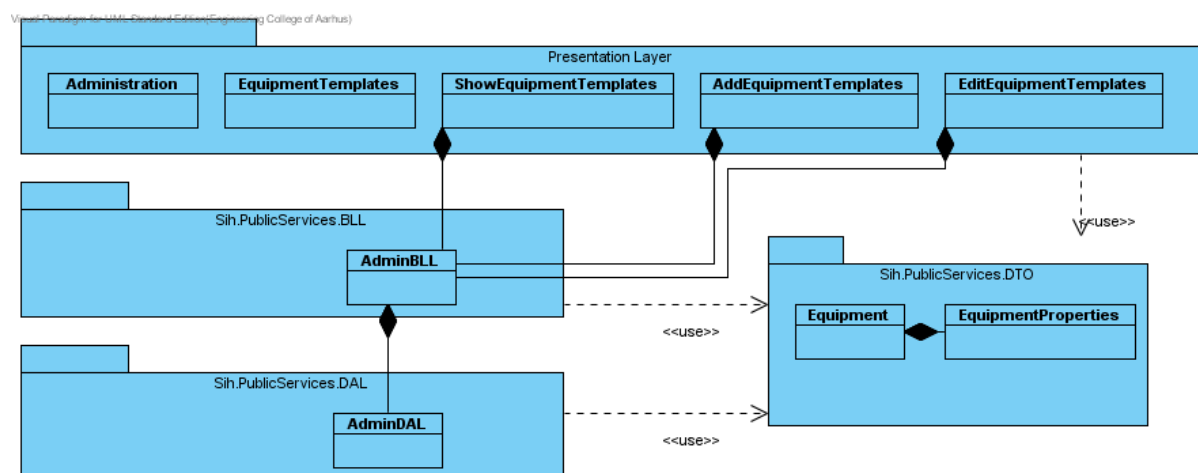
5.3.1.4.4 Alternativt forløb – Slet en person template:



Figur 27: Sekvensdiagram for use case realisering af use case 4 – Alternativforløb(Slet).

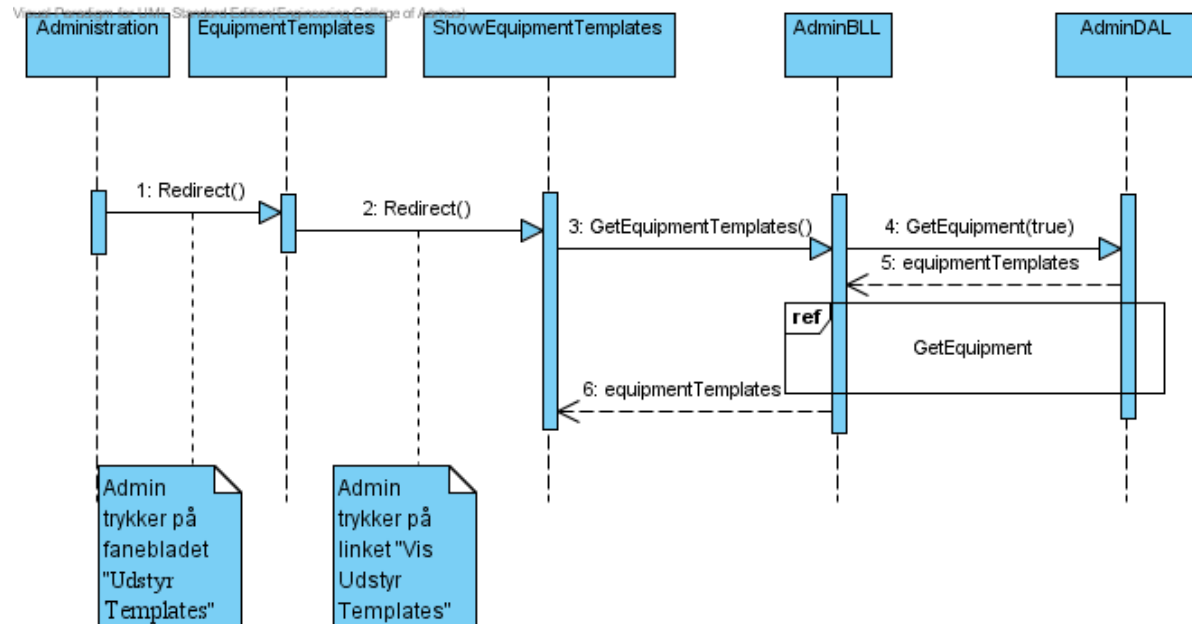
5.3.1.5 Use case 5. Håndter Udstyrs Templates

Denne use case indeholder en asp.net side der kalder ned i business logik laget for at hente og vise de eksisterende udstyrs templates.



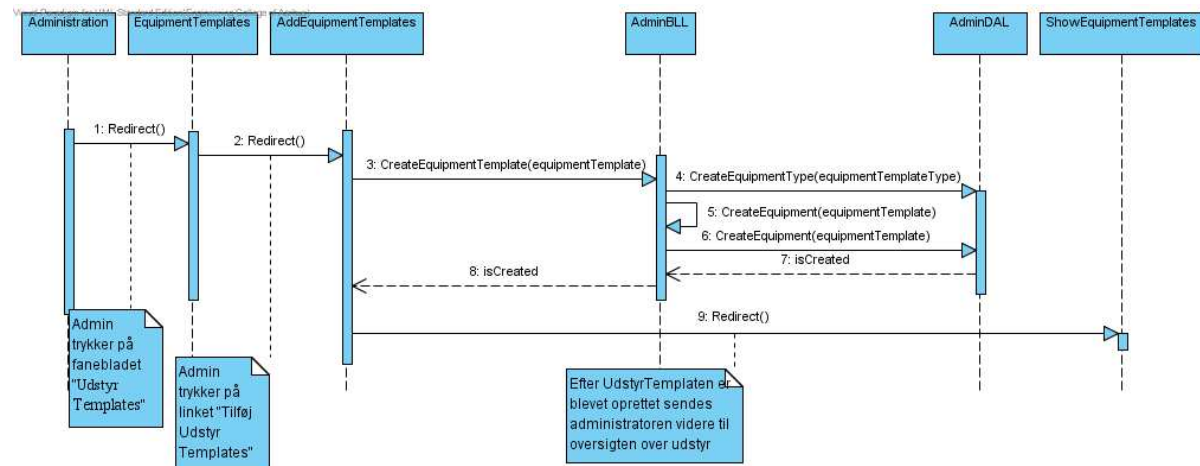
Figur 28: Klassediagram for use case realisering af use case 5

5.3.1.5.1 Normalforløb for use case 5:



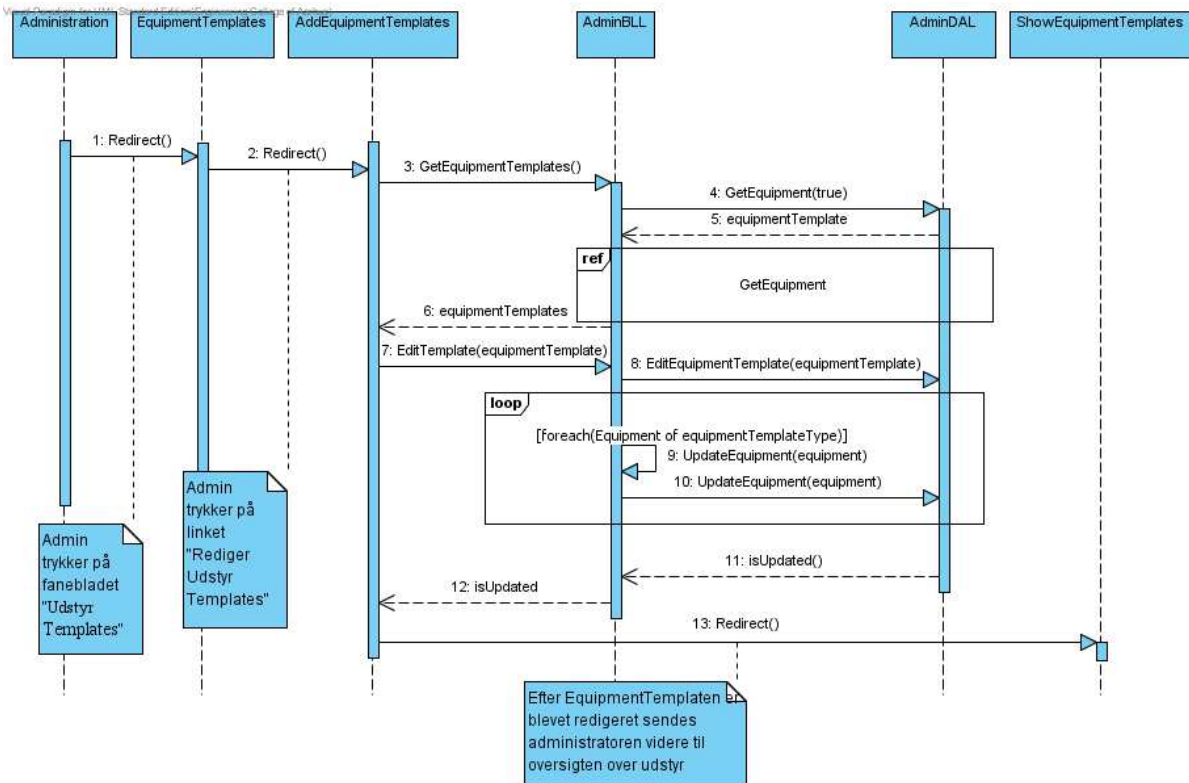
Figur 29: Sekvensdiagram for use case realisering af use case 5 - normalforløb.

5.3.1.5.2 Alternativt forløb – Tilføj en udstyr template:



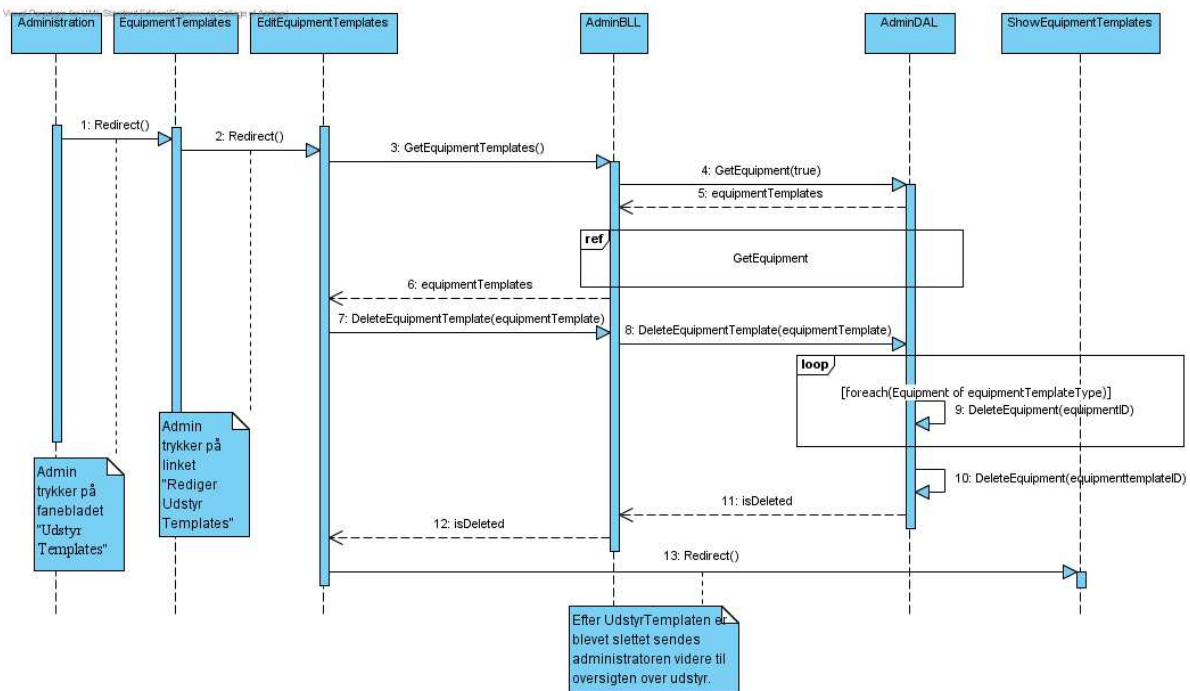
Figur 30: Sekvensdiagram for use case realisering af use case 5 – Alternativforløb(Tilføj).

5.3.1.5.3 Alternativt forløb – Rediger en udstyr template:



Figur 31: Sekvensdiagram for use case realisering af use case 5 – Alternativforløb(Rediger).

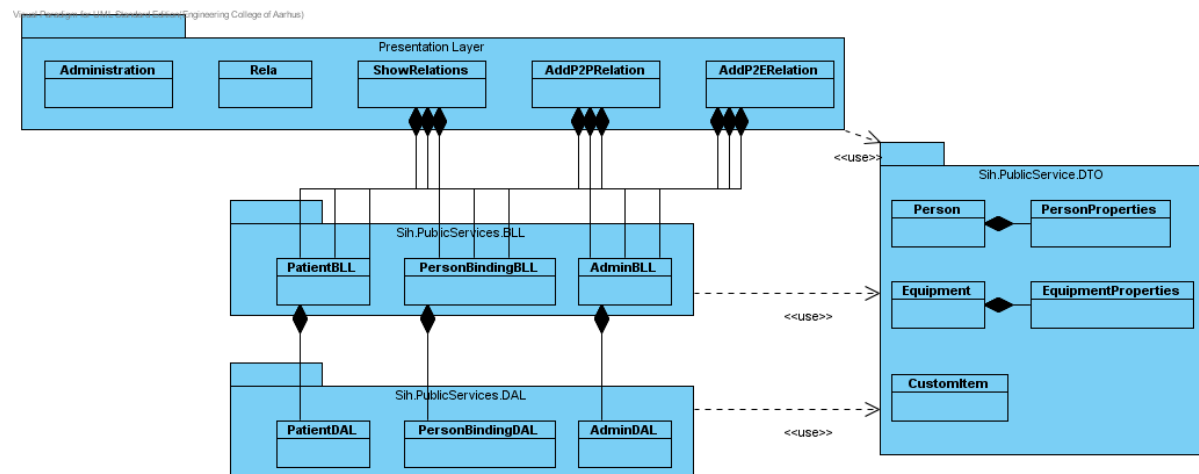
5.3.1.5.4 Alternativt forløb – Slet en udstyr template:



Figur 32: Sekvensdiagram for use case realisering af use case 5 – Alternativforløb(Slet).

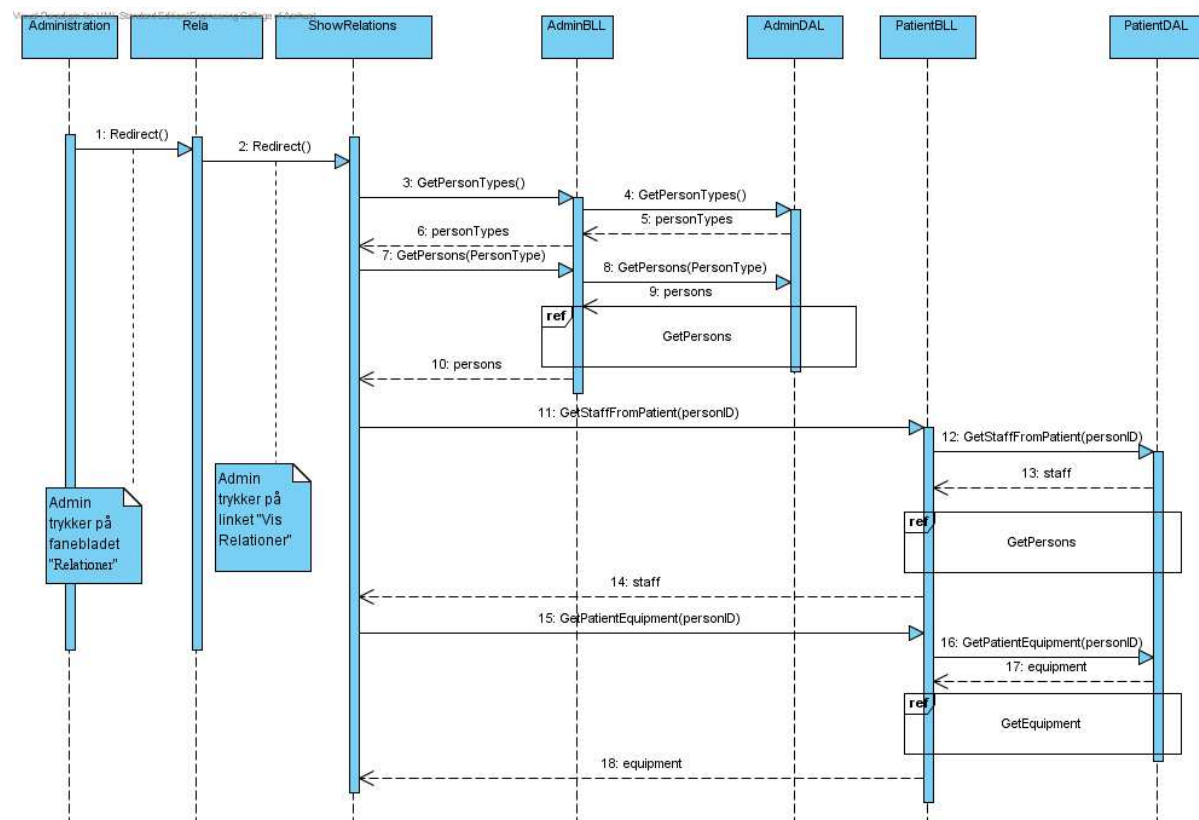
5.3.1.6 Use case 6. Håndter Relationer

Denne use case indeholder en asp.net side der kalder ned i business logik laget for at hente, tilføje eller slette relationer mellem personer eller mellem patienter og udstyr.



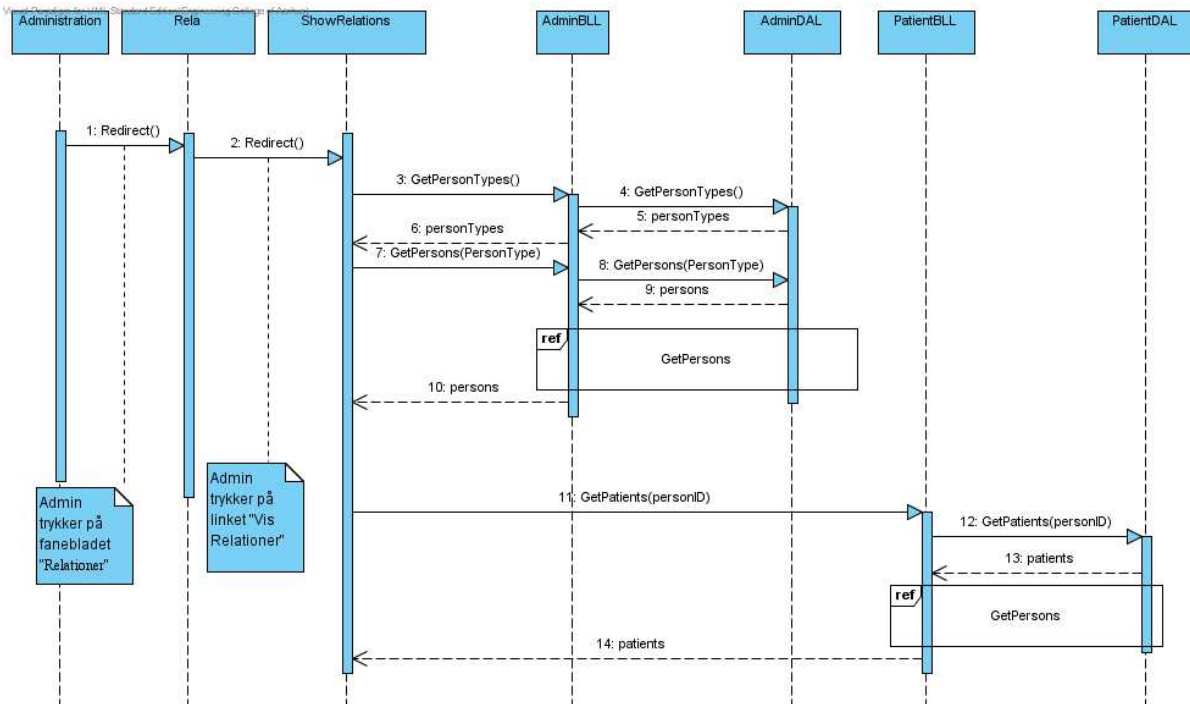
Figur 33: Klassediagram for use case realisering af use case 6

5.3.1.6.1 Normalforløb for use case 6 (patient relationer):



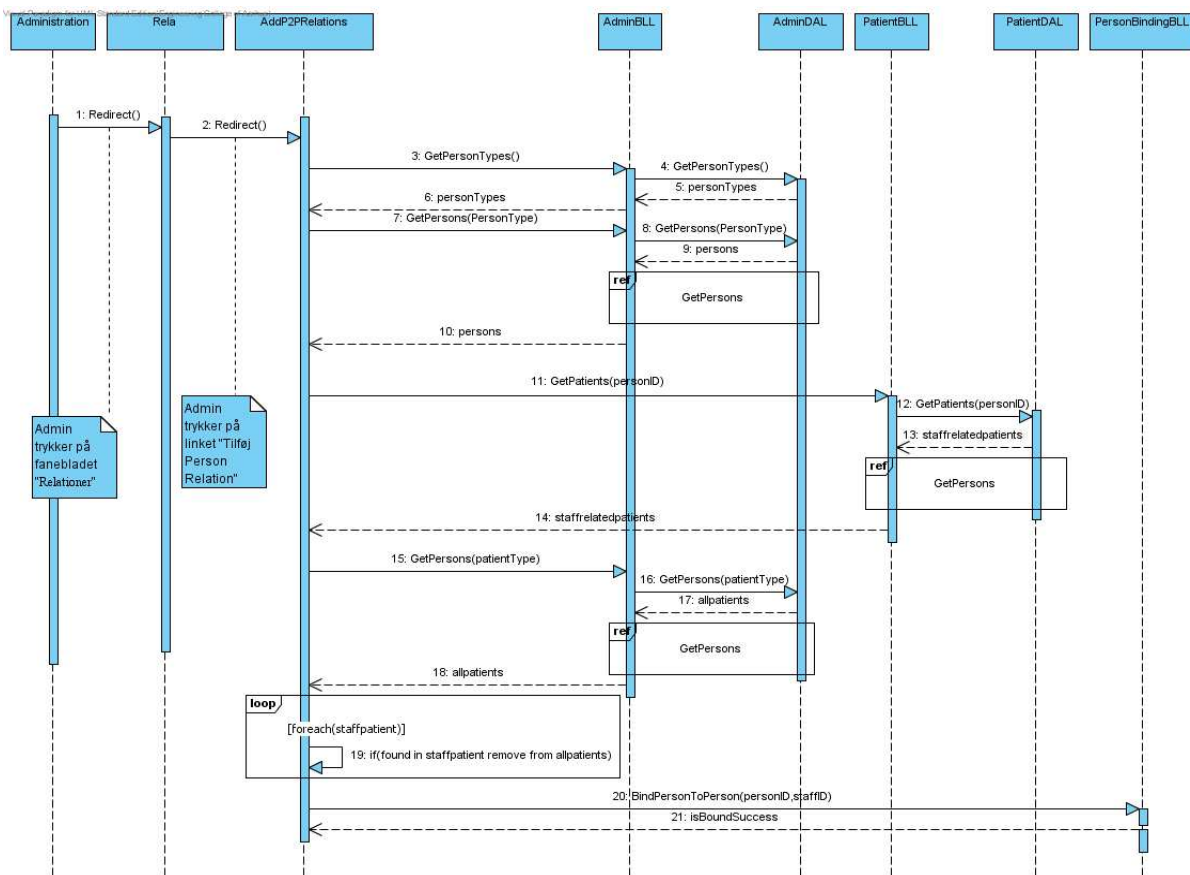
Figur 34: Sekvensdiagram for use case realisering af use case 6 – normalforløb for patient.

5.3.1.6.2 Normalforløb for use case 6 (staff relationer):



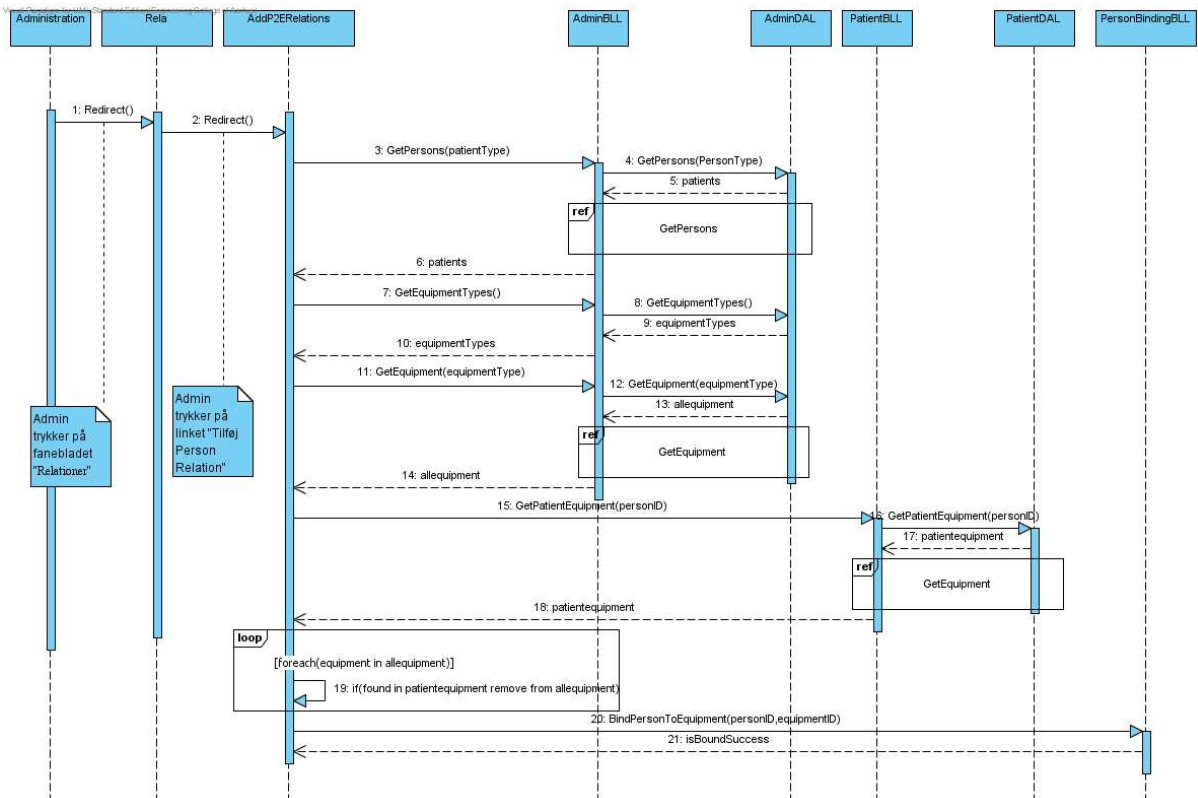
Figur 35: Sekvensdiagram for use case realisering af use case 6 – Normalforløb for staff.

5.3.1.6.3 Alternativt forløb use case 6 (bind person til person):



Figur 36: Sekvensdiagram for use case realisering af use case 6 – Alternativforløb(Bind person to person).

5.3.1.6.4 Alternativt forløb use case 6 (bind person til et udstyr):

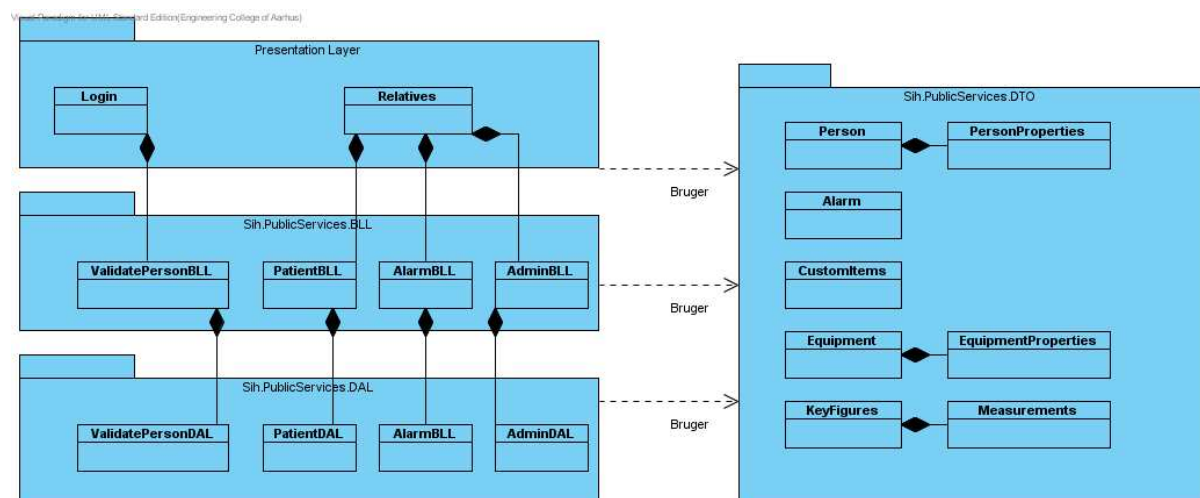


Figur 37: Sekvensdiagram for use case realisering af use case 6 – Alternativforløb(Bind person to equipment).

5.3.2 Subsystem – Relatives

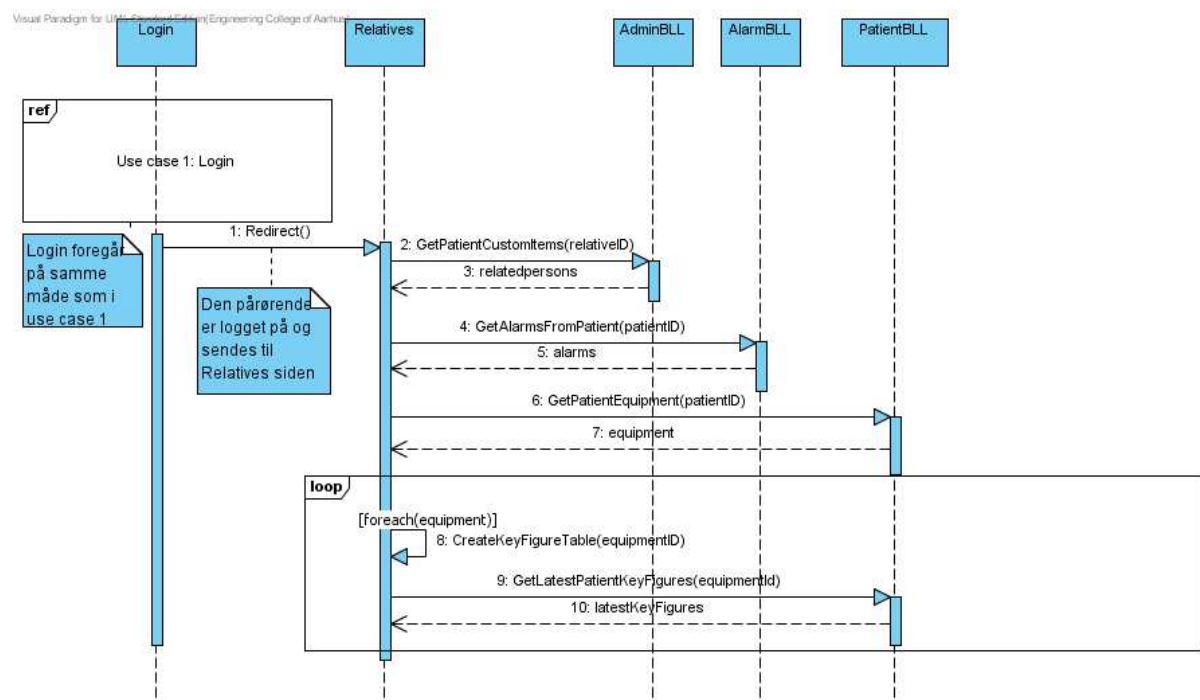
5.3.2.1 Use case 7. Relatives

Denne use case indeholder en asp.net side der kalder ned i business logik laget for at hente patienter tilknyttet en pårørende og data for denne.



Figur 38: Klassediagram for use case realisering af use case 7

5.3.2.1.1 Normalforløb for use case 7:



Figur 39: Sekvensdiagram for use case realisering af use case 7.

5.3.3 Subsystem – PatientData Plugins

Dette subsystem er realiseret som en DLL med 7 usercontrols der til sammen realiserer de use cases der er under dette subsystem. Der kan læses mere om hvilke interfaces der skal overholdes og hvordan et plugin udvikles i Systemarkitektur dokument for SIH Presentation Framework.

5.3.3.1 Valg af patient

Fælles for alle UserControls i dette plugin og i plugin'et udviklet til håndtering af påmindelser er, at der skal kunne vælges en patient. Hvis en patient allerede er valgt i en anden UserControl indenfor denne plugin skal denne patient fortsat være valgt. Undtagelsen for denne regel er når den valgte person ikke har tilkoblet det udstyr der skal vises data for.

5.3.3.1.1 User story for valg af patient

Fru Hansen får målt blodsukker og blodtryk, men hun får ikke mål blodgas da hun ingen problemer har med dette.

Fru Hansens læge benytter SIH Presentation Frameworket med dette plugin og går ind under "se seneste blodtryksmålinger". Her vælges Fru Hansen. (Se sekvensdiagram for "Vælg Patient – Patient ikke valgt").

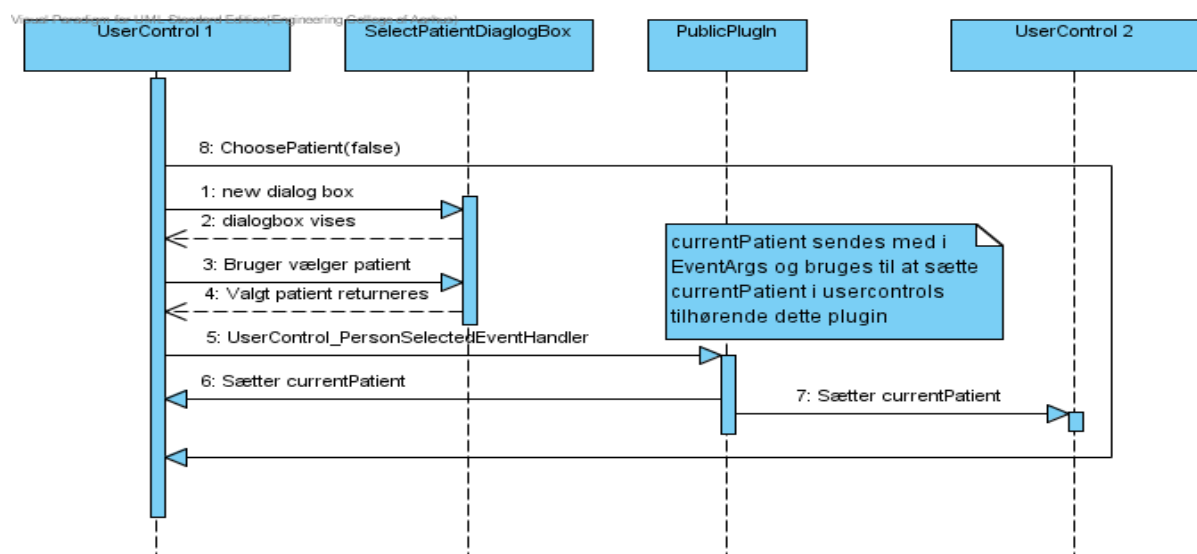
Fru Hansens målinger vises og nu ønsker lægen at se Blodsukker målingerne.

Lægen vælger derfor "se seneste blodsukkermålinger" og nu skal der ikke igen vælges patient for det er allerede sket. (Se sekvensdiagram for "Vælg patient – Patient allerede valgt").

Lægen kommer nu i tanke om at han også skal have set på nogle blodgas målinger for Hr. Jensen og lægen går derefter ind under "se seneste blodgasmålinger". Fru Hansen er stadig valgt men hun har ikke en måler tilknyttet og derfor vises en liste hvor Hr. Jensen er en valgmulighed. (Se sekvensdiagram for "Vælg patient – vælg ny patient").

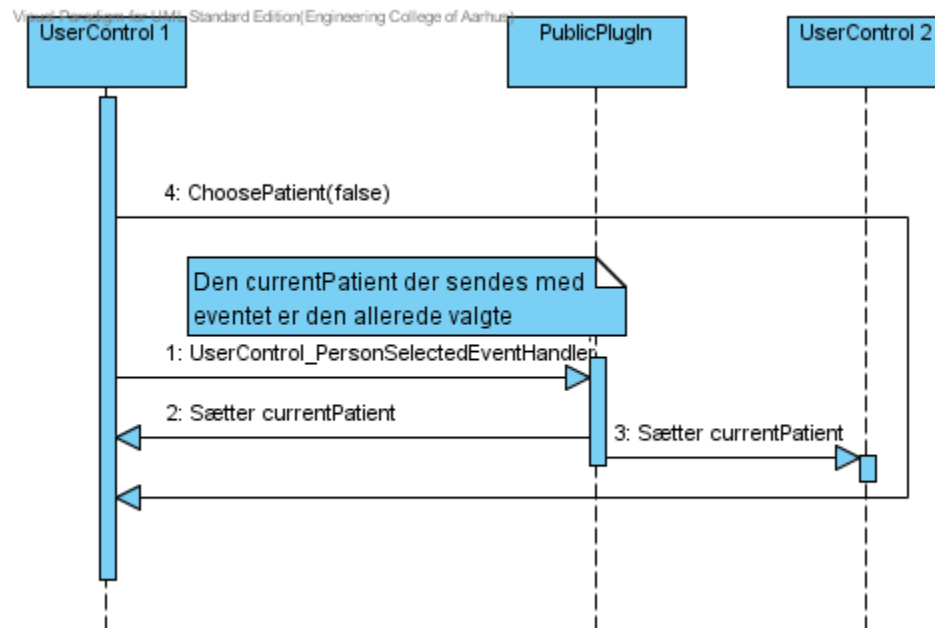
Valg af ny patient kan også ske indefra enhver UserControl.

5.3.3.1.2 Vælg patient – Patient ikke valgt



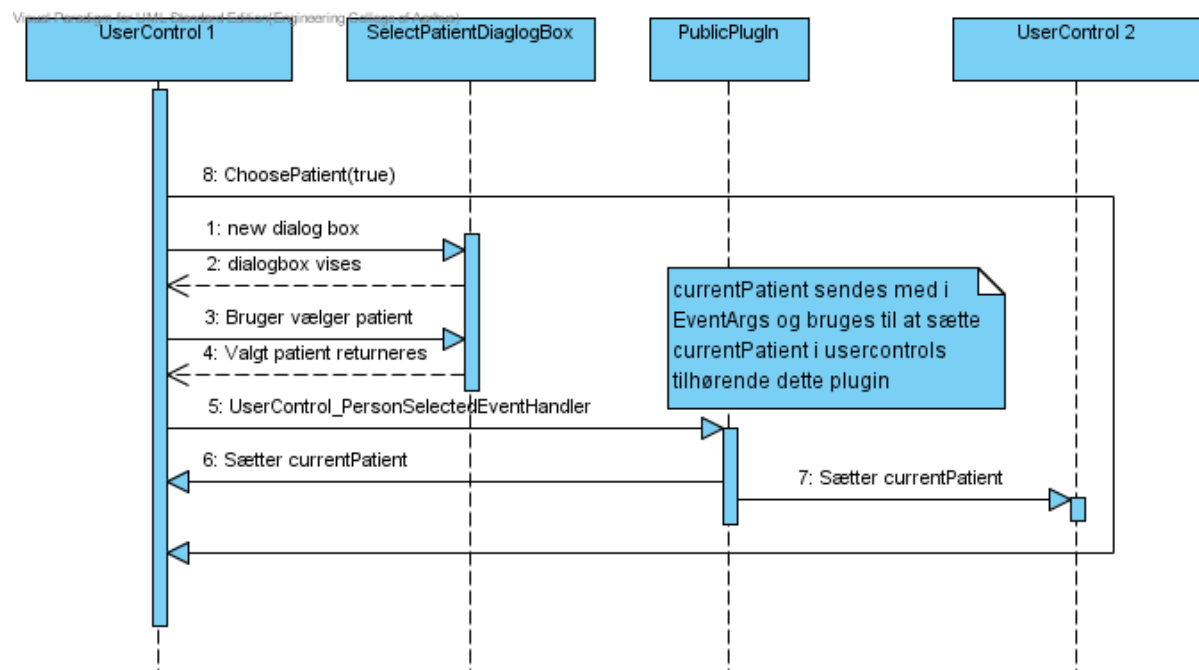
Figur 40: Sekvensdiagram for patient ikke valgt.

5.3.3.1.3 Vælg patient – patient allerede valgt



Figur 41: Sekvensdiagram for patient allerede valgt.

5.3.3.1.4 Vælg patient – vælg ny patient



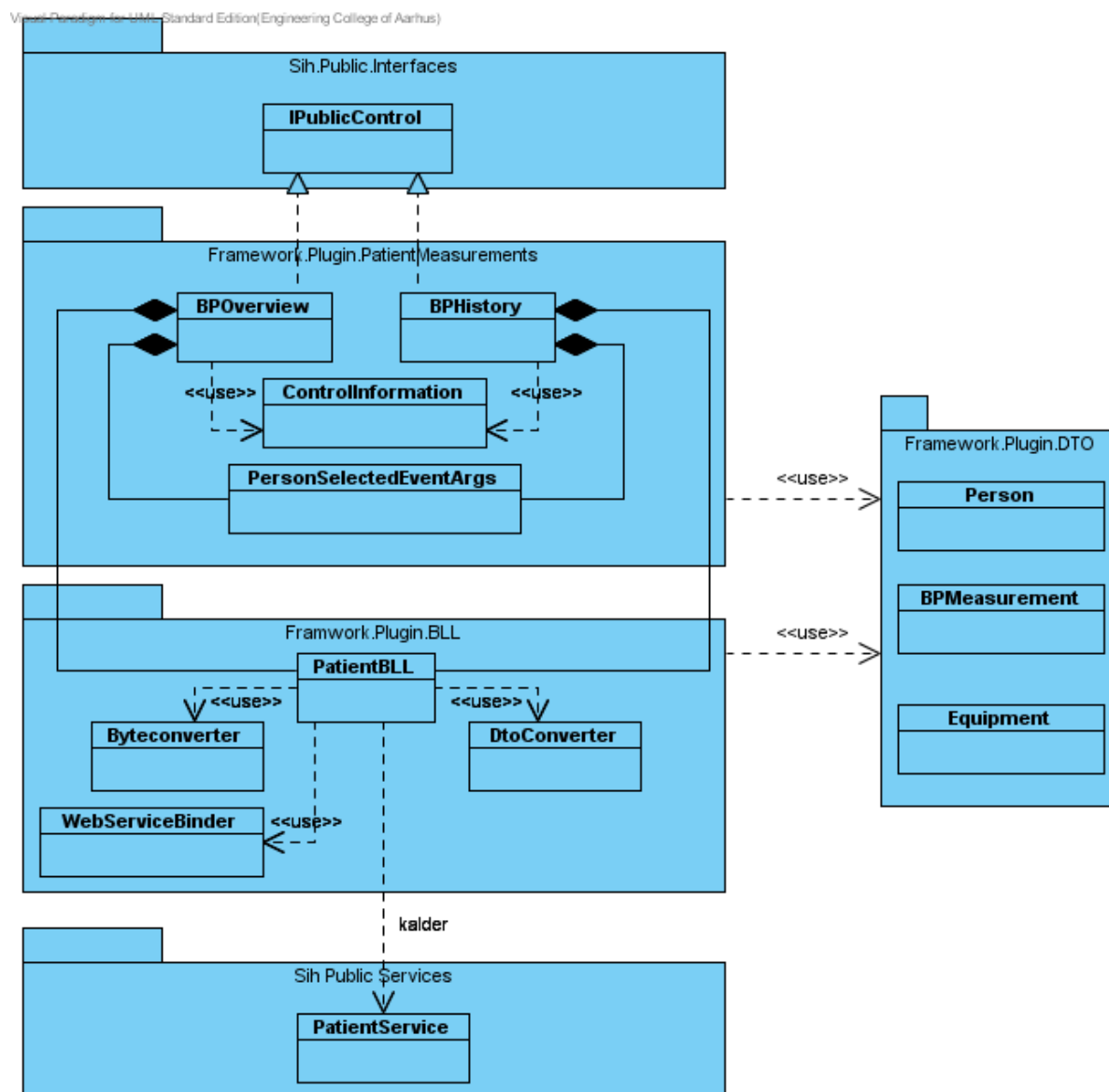
Figur 42: Sekvensdiagram for vælg ny patient.

5.3.3.2 Use Case 8: Login

Denne use case er realiseret i SIH Presentation Frameworket.

5.3.3.3 Use Case 9. Se Blodtryksdata

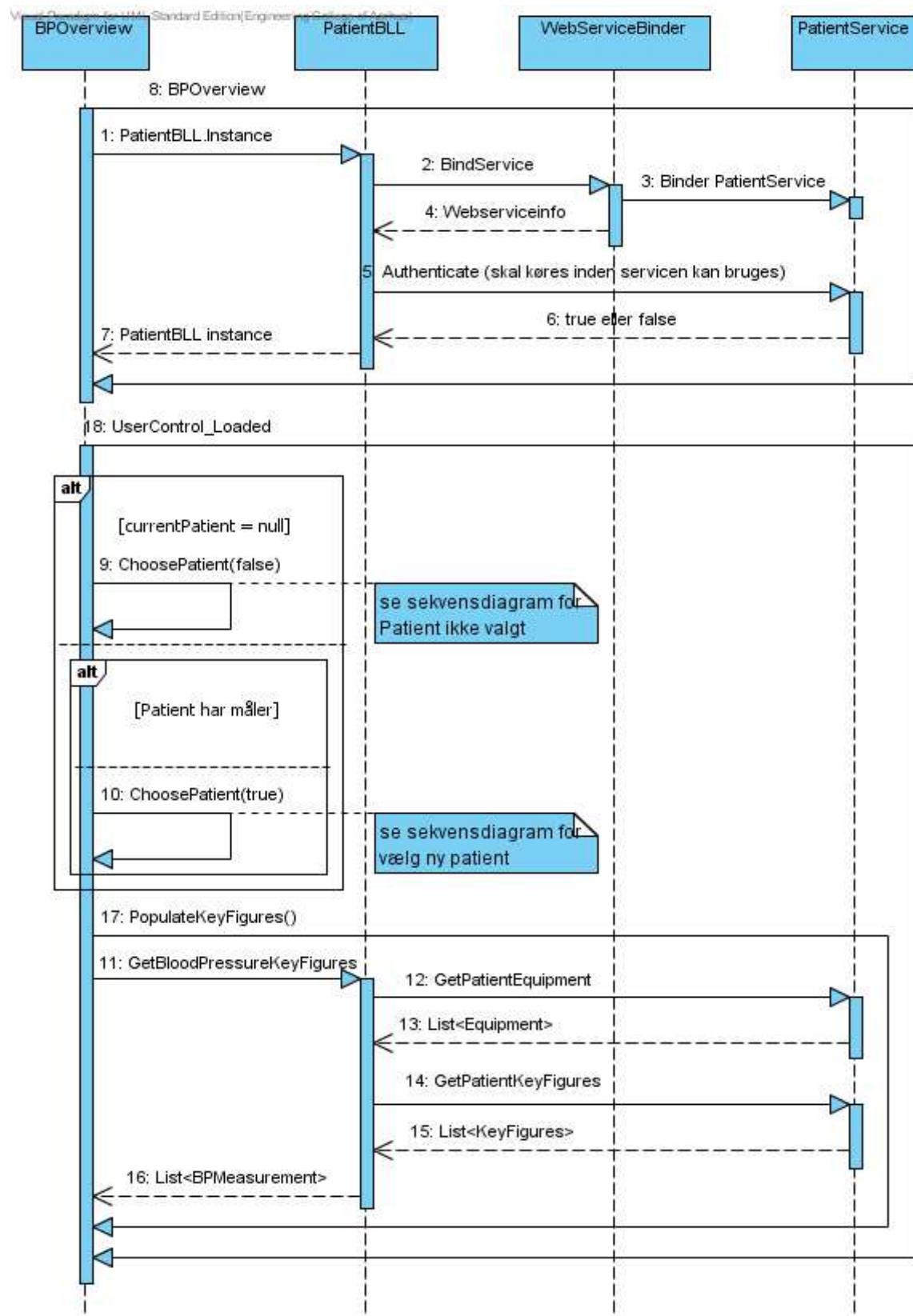
Denne use case er tilgængelig for praktiserende læger, hospitalslæger og sygeplejersker. Use casens hovedklasser er BPOverview og BPHistory som kan ses i sammenhæng med resterende nedenfor.



Figur 43: Klassediagram for use case realisering af use case 9

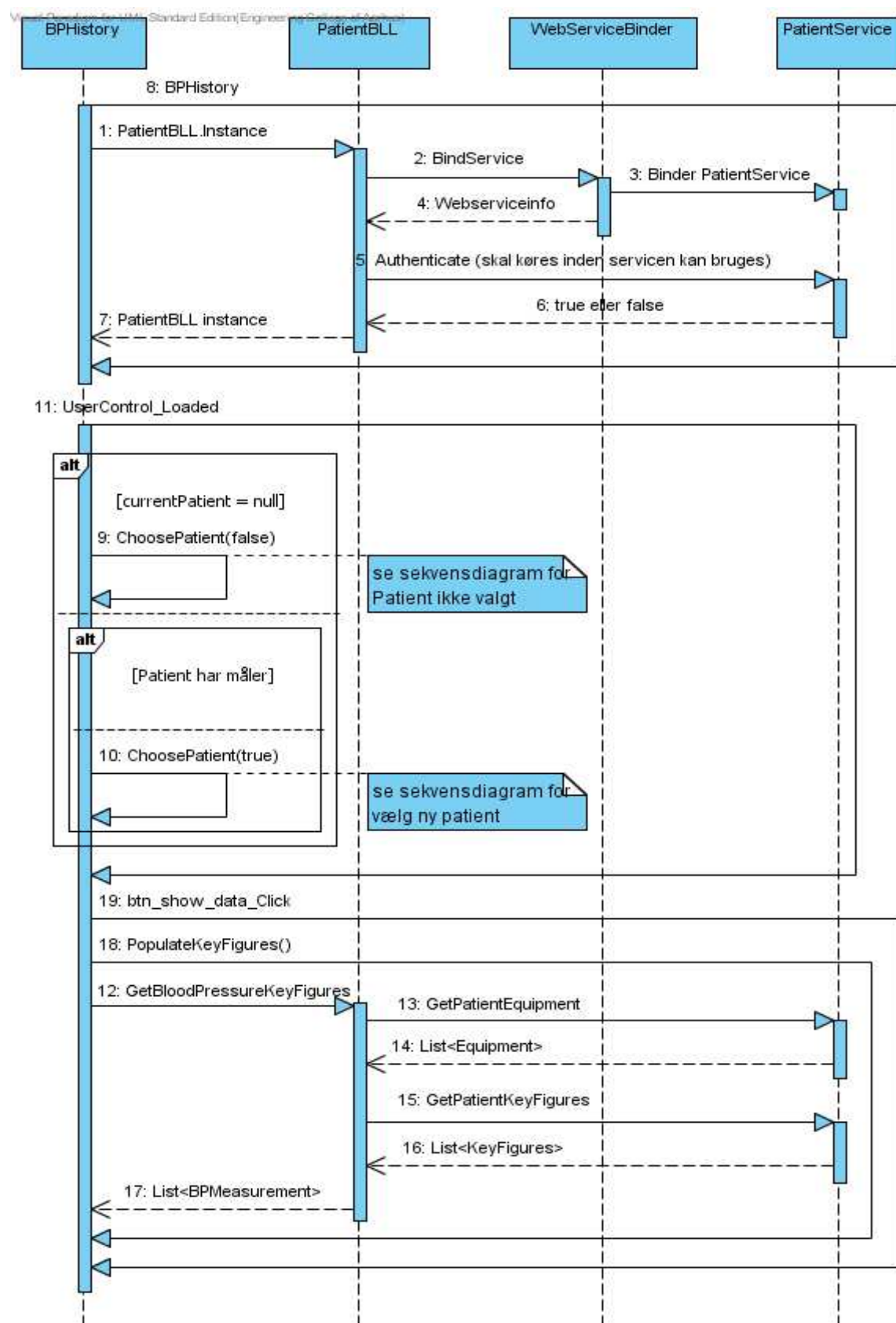
På klassediagrammet ovenfor er der ikke vist hvilke klasser der har fat i hvilke DTO'er. Overblikket over klasserne bliver forringet væsentligt ved at gøre dette. BPOverview og BPHistory benytter Person og BPMeasurement. PatientBLL og DtoConverter benytter alle DTO'er.

5.3.3.3.1 Normalforløb: "Se seneste blodtryksmålinger"



Figur 44: Sekvensdiagram for "se seneste blodtryksmålinger"

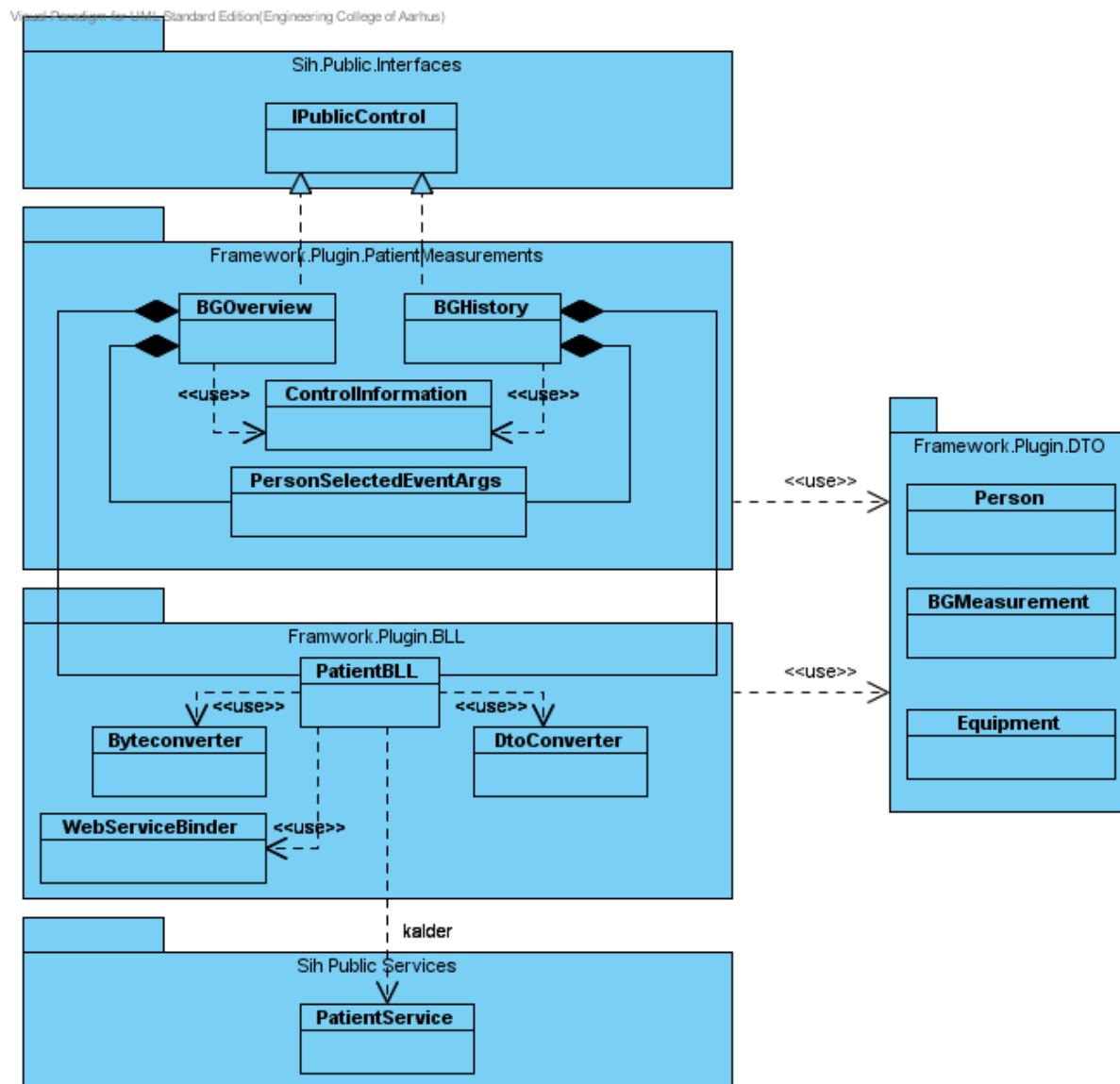
5.3.3.3.2 Alternativt forløb: "se historiske data"



Figur 45: Sekvensdiagram for "se historiske data"

5.3.3.4 Use Case 10. Se Blodgasdata

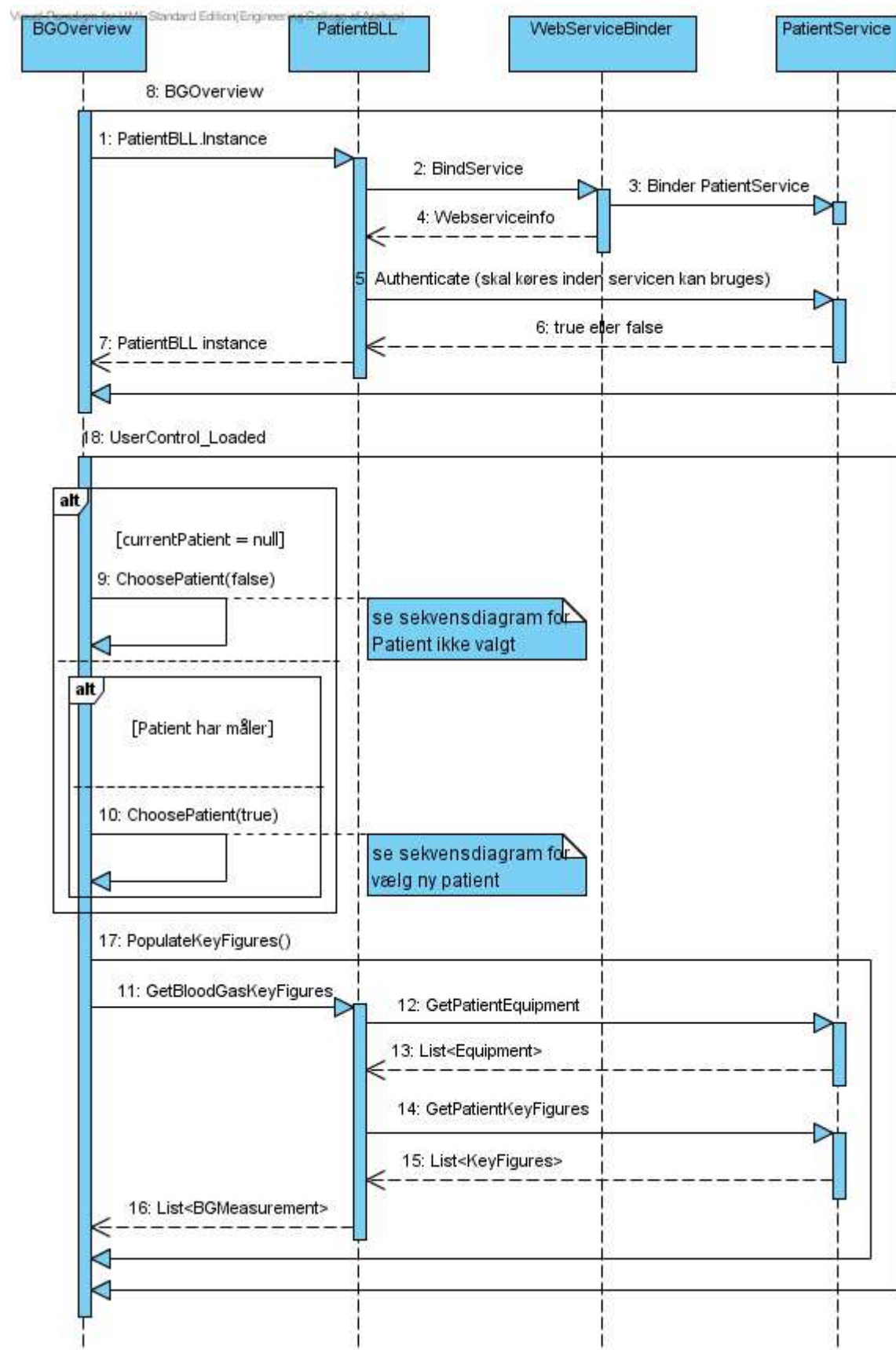
Denne use case er tilgængelig for praktiserende læger, hospitalslæger og sygeplejersker. Use casens hovedklasser er BGOverview og BGHistory som kan ses i sammenhæng med resterende nedenfor.



Figur 46: Klassediagram for use case realisering af use case 10

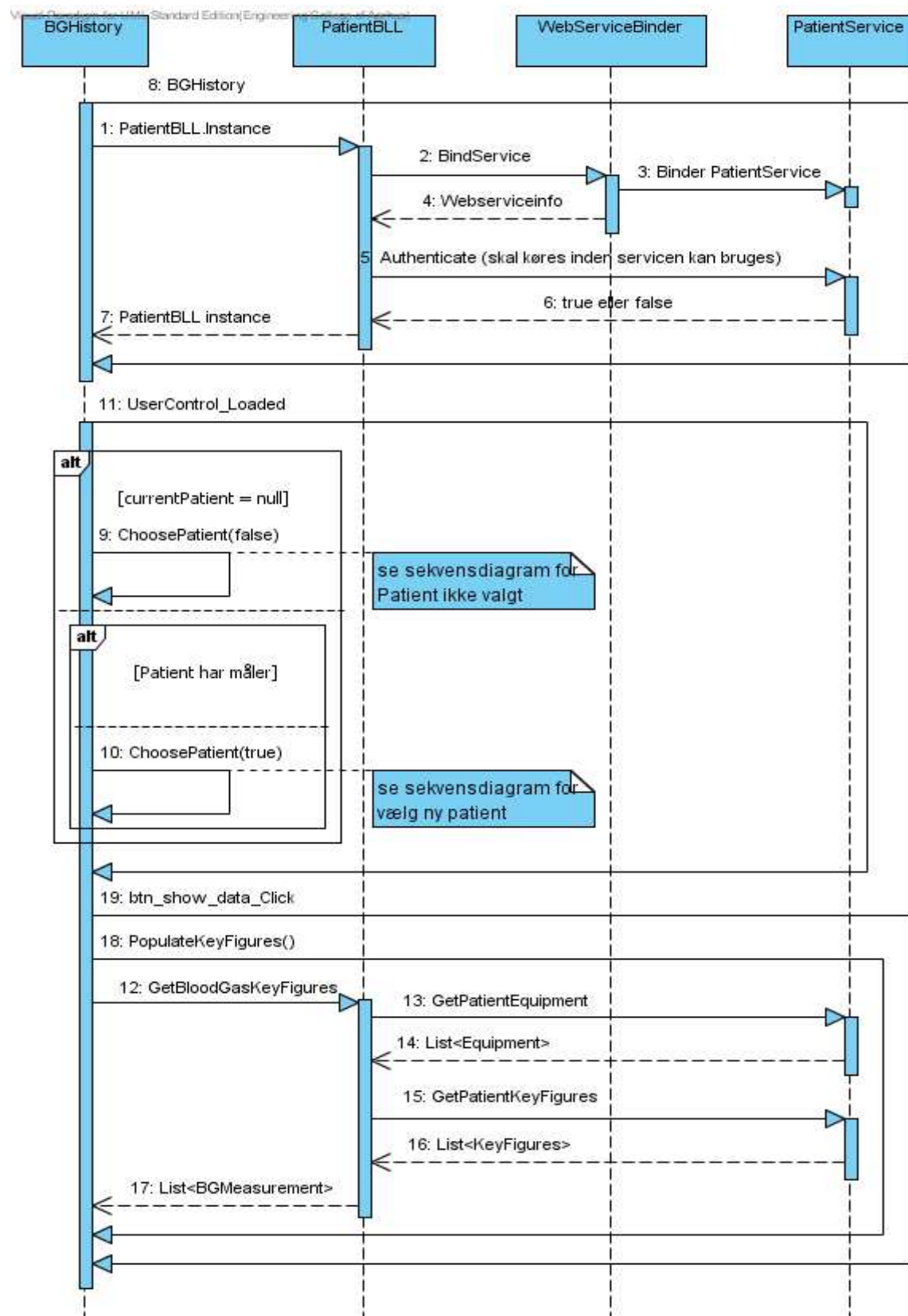
På klassediagrammet ovenfor er der ikke vist hvilke klasser der har fat i hvilke DTO'er. Overblikket over klasserne bliver forringet væsentligt ved at gøre dette. BGOverview og BGHistory benytter Person og BGMeasurement. PatientBLL og DtoConverter benytter alle DTO'er.

5.3.3.4.1 Normalforløb: "Se seneste blodgasmålinger"



Figur 47: Sekvensdiagram for "se seneste blodgasmålinger"

5.3.3.4.2 Alternativt forløb: "se historiske data"

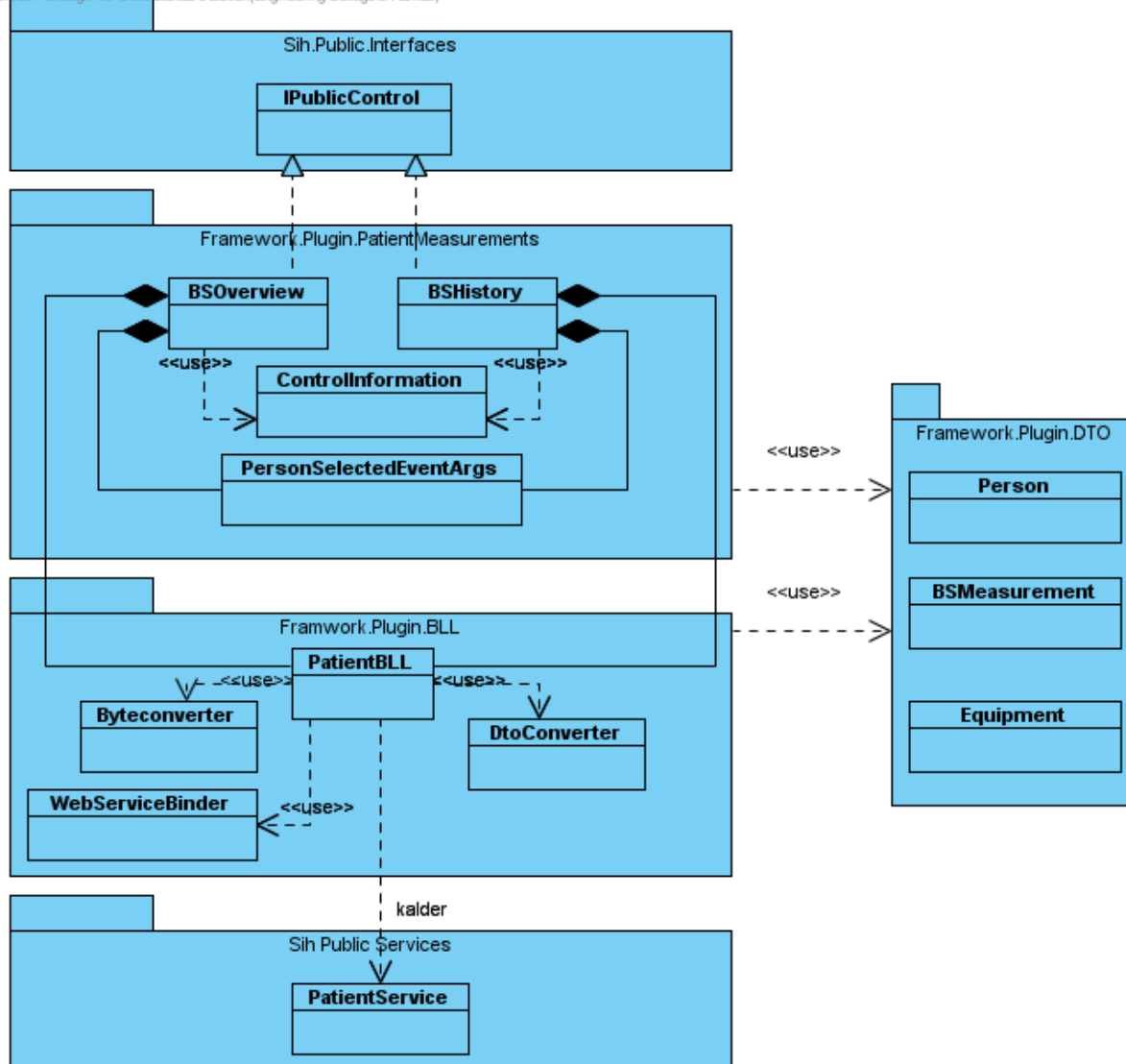


Figur 48: Sekvensdiagram for "se historiske data"

5.3.3.5 Use Case 11. Se Blodsukkerdata

Denne use case er tilgængelig for praktiserende læger, hospitalslæger og sygeplejersker. Use casens hovedklasser er BSOversview og BSHistory som kan ses i sammenhæng med resterende nedenfor.

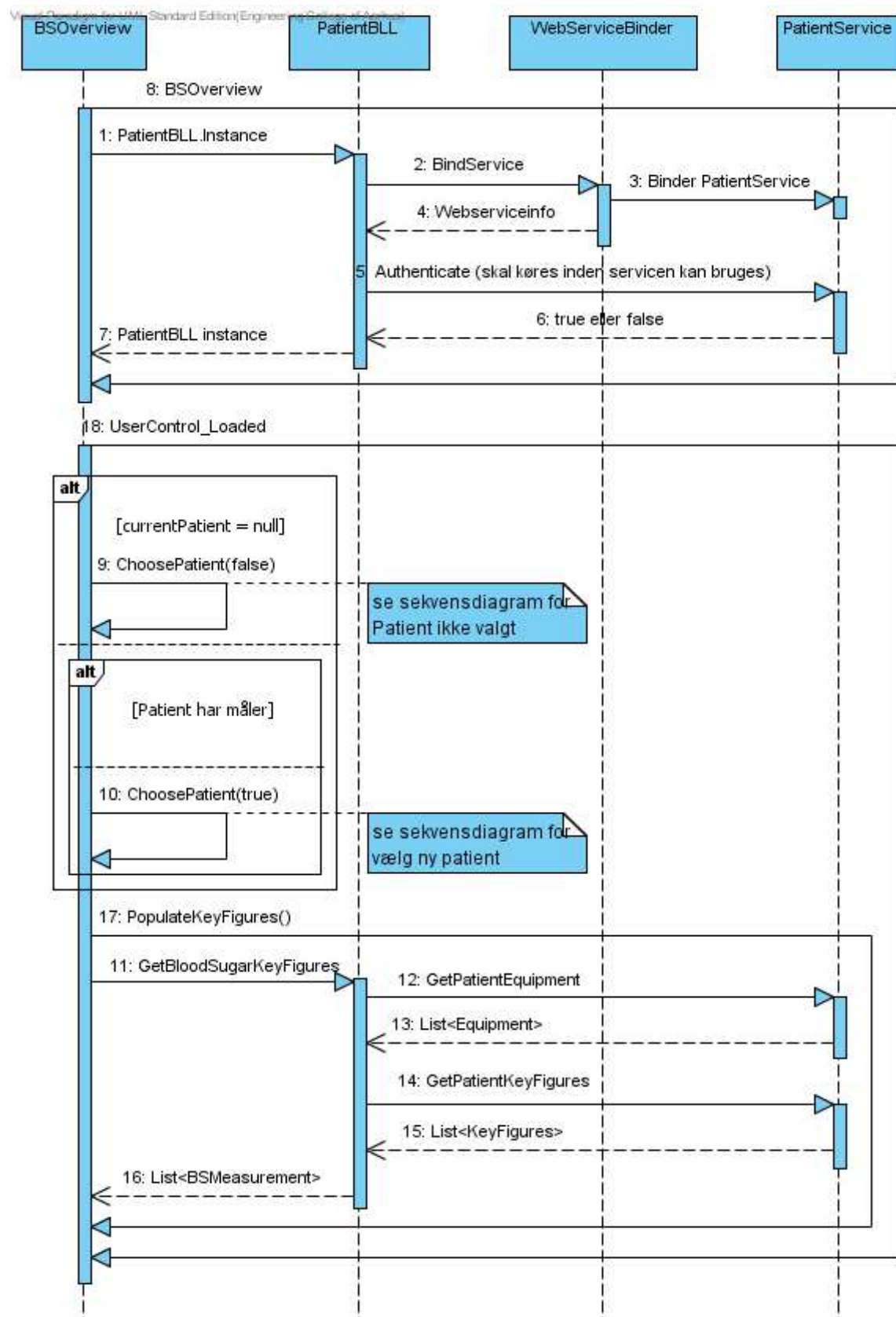
Visual Paradigm for UML - Standard Edition (Engineering College of Aarhus)



Figur 49: Klassediagram for use case realisering af use case 11

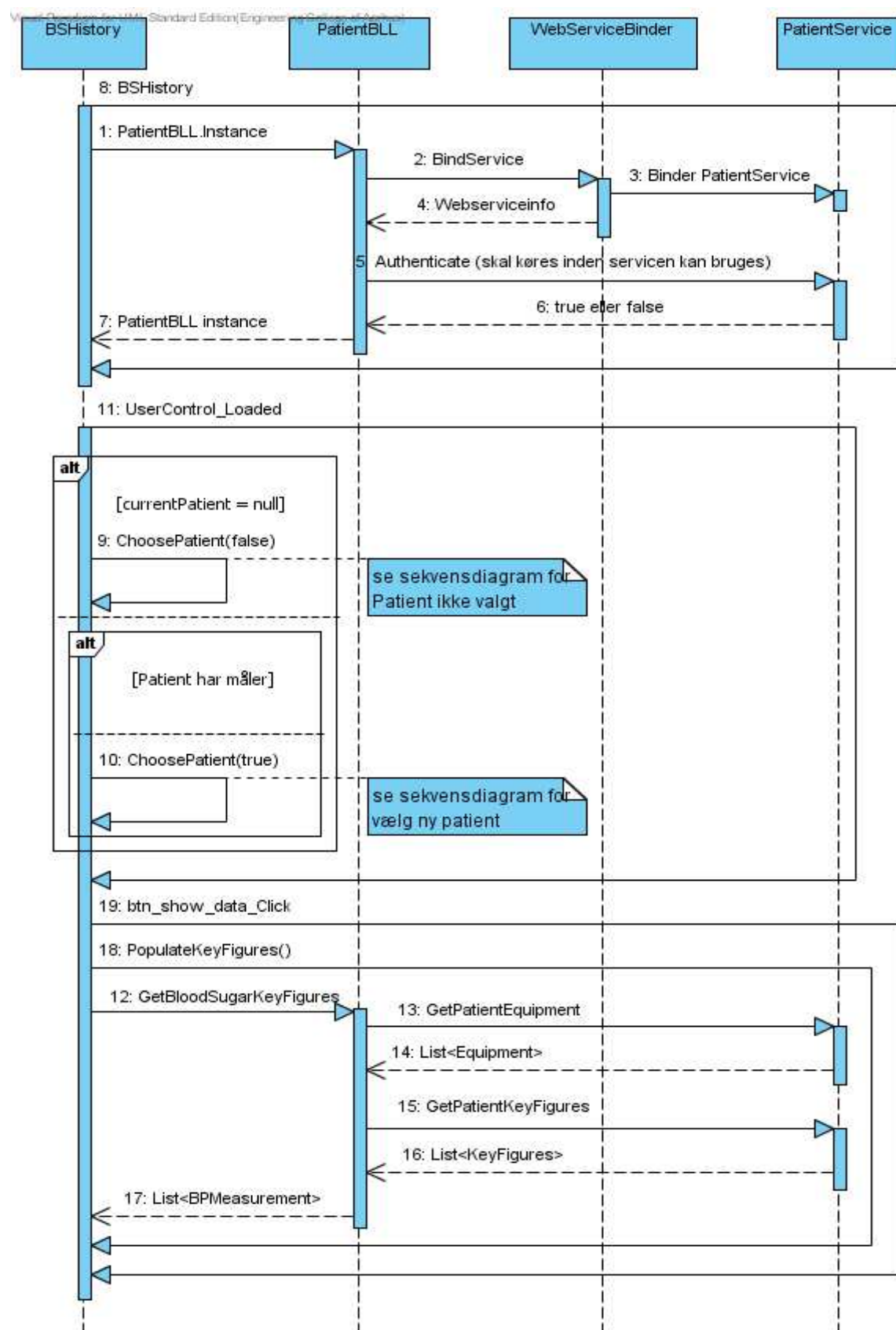
På klassediagrammet ovenfor er der ikke vist hvilke klasser der har fat i hvilke DTO'er. Overblikket over klasserne bliver forringet væsentligt ved at gøre dette. BSOversview og BSHistory benytter Person og BSMeasurement. PatientBLL og DtoConverter benytter alle DTO'er.

5.3.3.5.1 Normalforløb: "Se seneste blodsukkermålinger"



Figur 50: Sekvensdiagram for "se seneste blodsukkermålinger"

5.3.3.5.2 Alternativt forløb: "se historiske data"

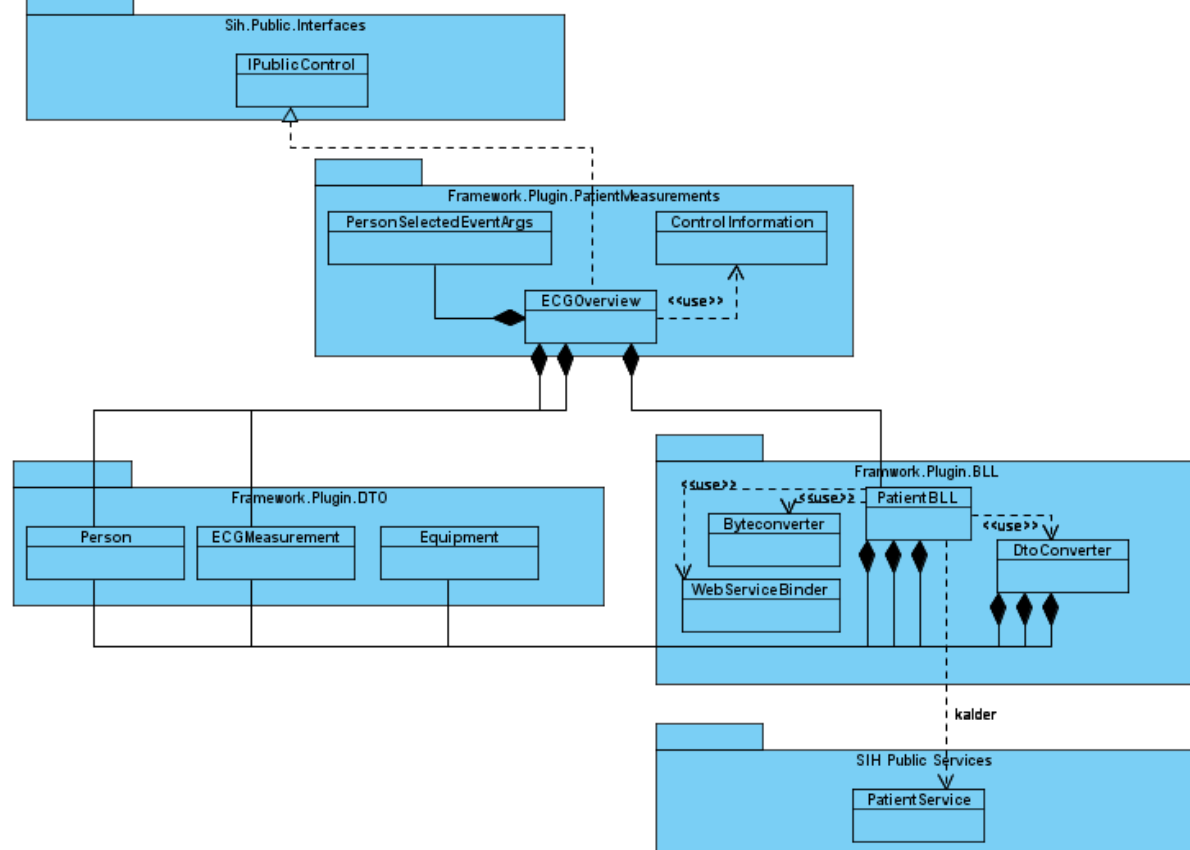


Figur 51: Sekvensdiagram for "se historiske data"

5.3.3.6 Use Case 12. Se EKG data

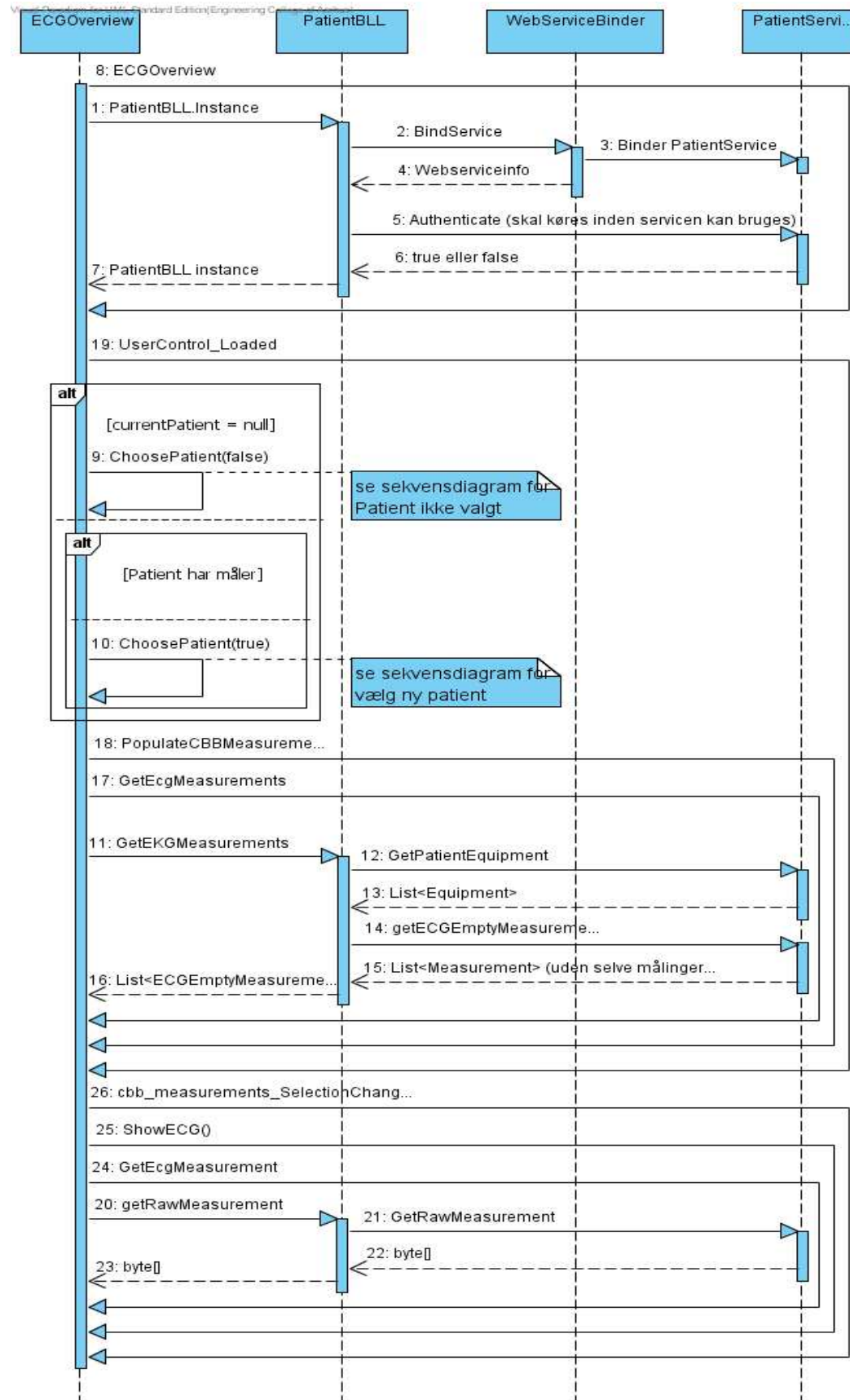
Denne use case er tilgængelig for praktiserende læger og hospitalslæger.
Use casens hovedklasse er ECGOverview som kan ses i sammenhæng med resterende nedenfor.

Visual Paradigm for UML Standard Edition (Engineering College of Aarhus)



Figur 52: Klassediagram for use case realisering af use case 12

5.3.3.6.1 Normalforløb: "Se EKG målinger"

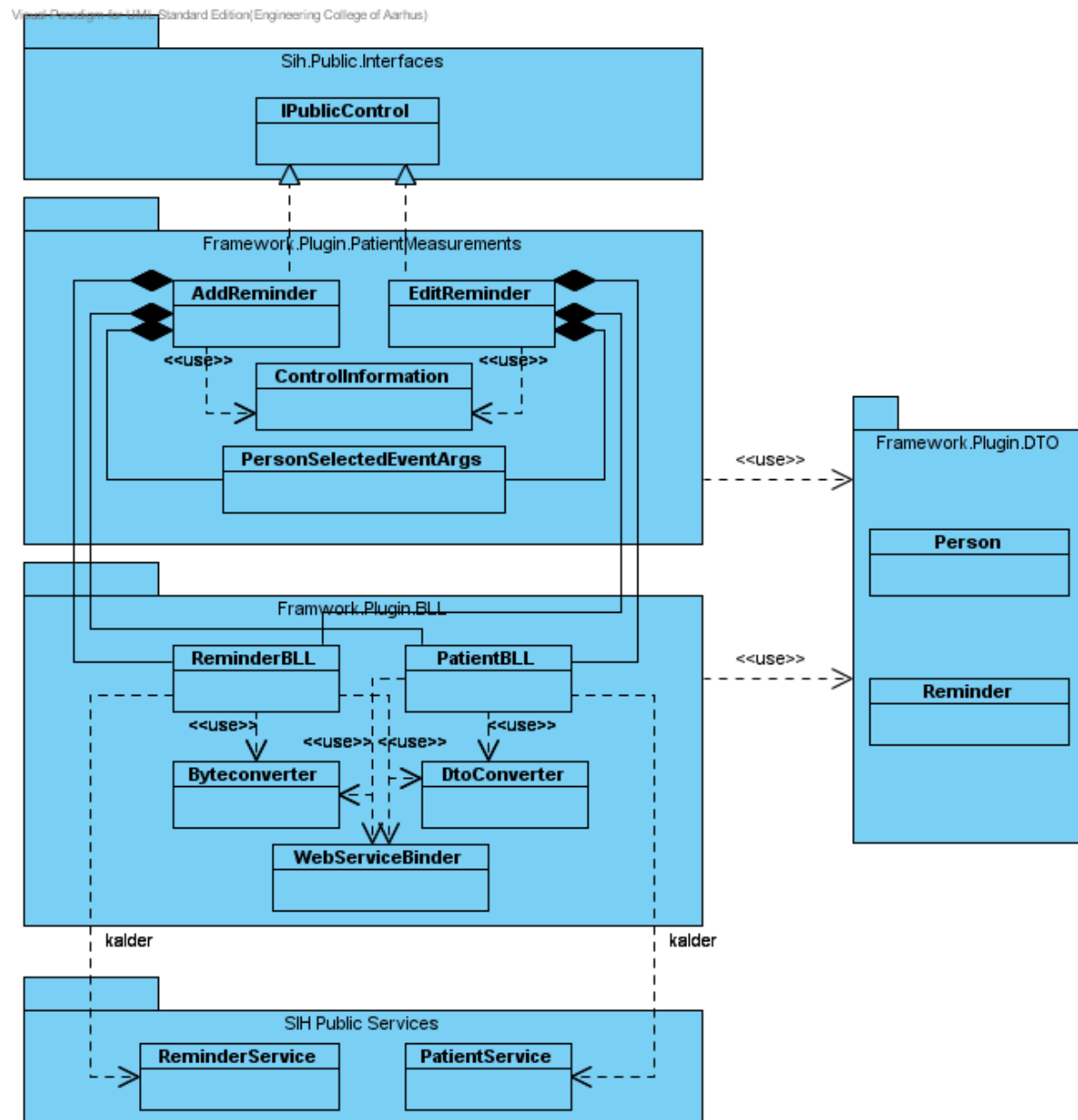


Figur 53: Sekvensdiagram for "se EKG målinger"

5.3.4 Subsystem – Reminder Plugin

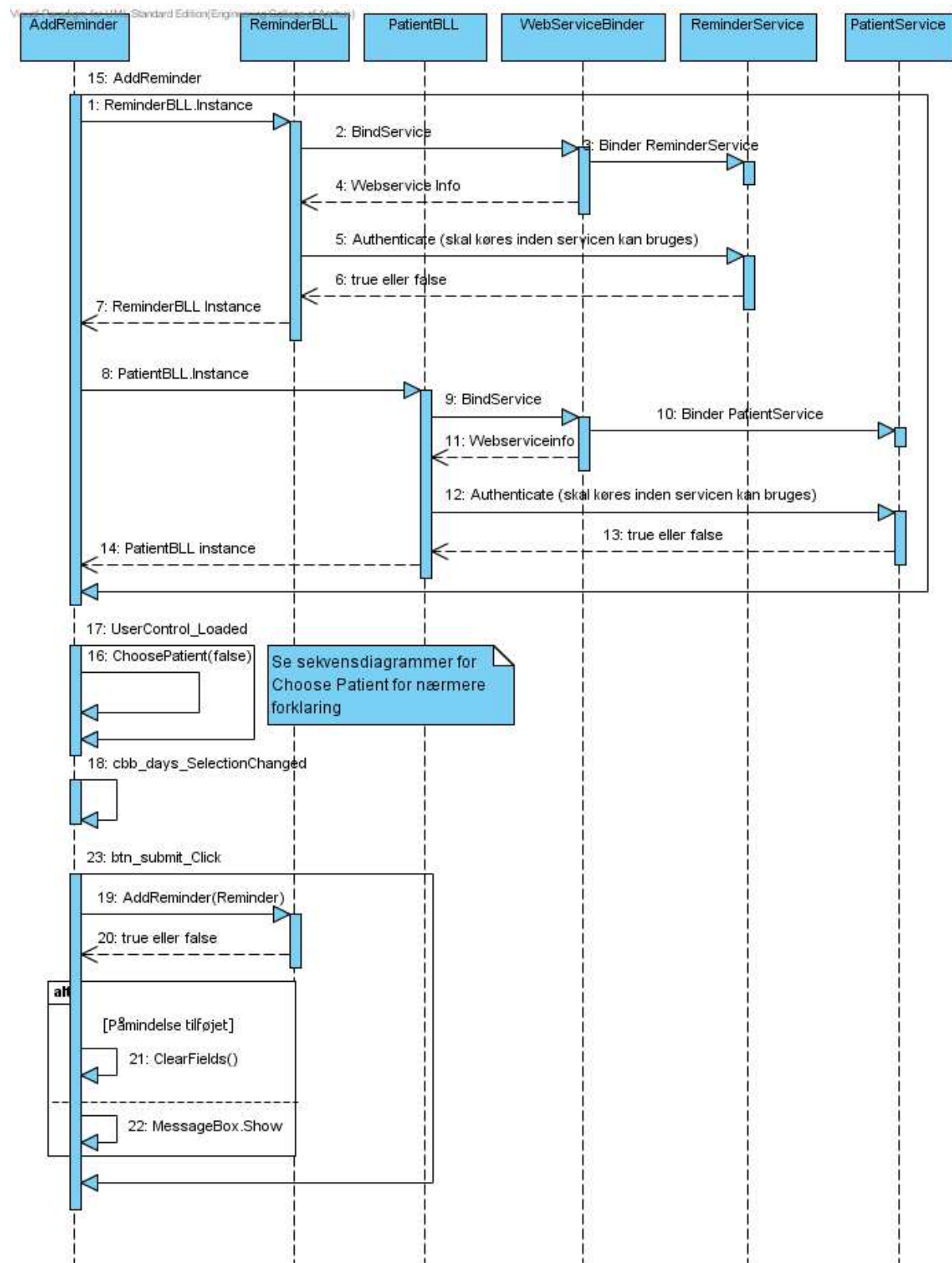
5.3.4.1 Use Case 13

Denne use case er tilgængelig for praktiserende læger og hospitalslæger. Use Casens hovedklasser er AddReminder og EditReminder som ses i sammenhæng herunder.



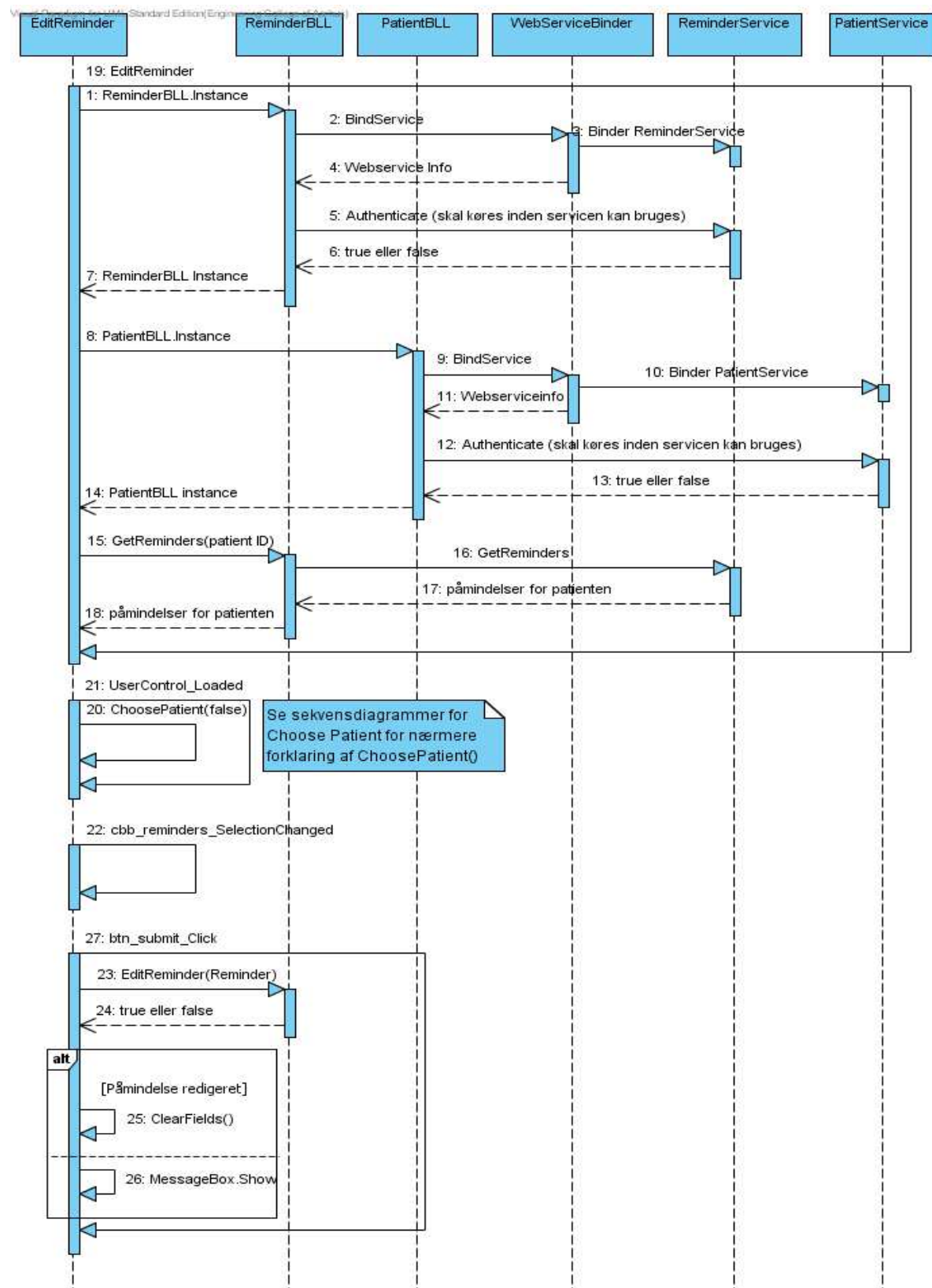
Figur 54: Klassediagram for realisering af use case 13

5.3.4.1.1 Håndter påmindelse normaltforløb (Tilføj påmindelse)



Figur 55: Sekvensdiagram for tilføjelse af en påmindelse til systemet.

5.3.4.1.2 Håndter påmindelse alternativt forløb – Rediger påmindelse

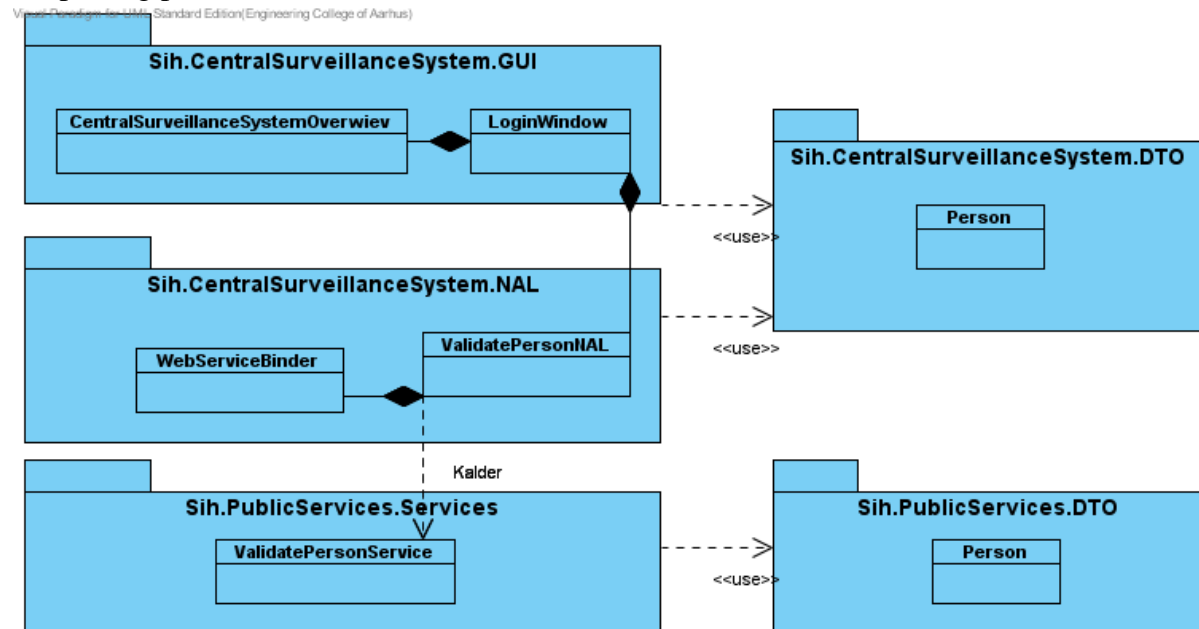


Figur 56: Sekvensdiagram for redigering af en påmindelse til systemet.

5.3.1 Subsystem – Central Surveillance System

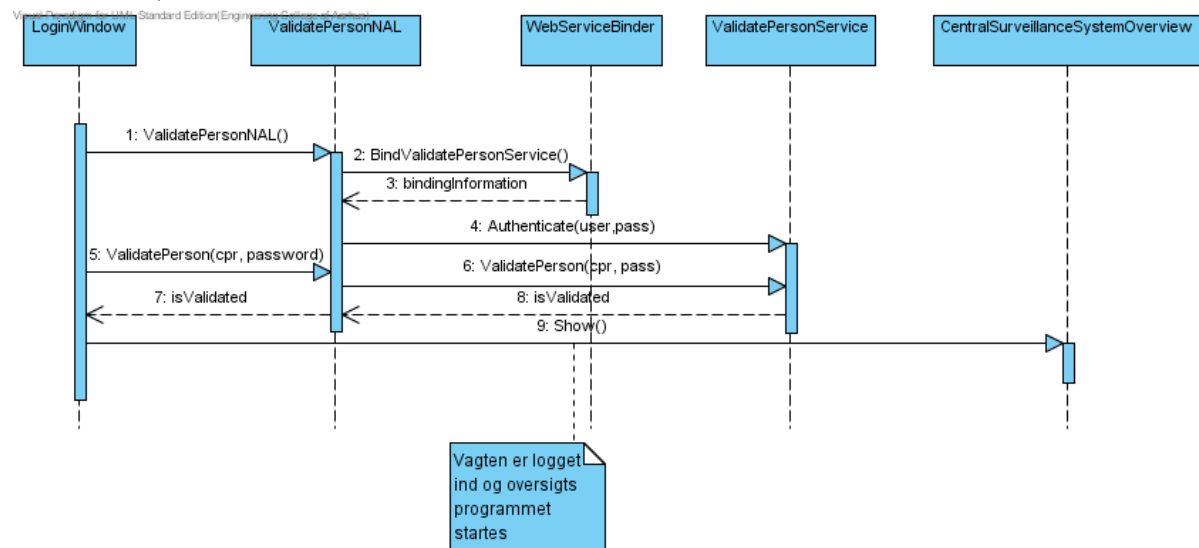
5.3.1.1 Use case 14. Login

Denne use case indeholder et WPF vindue der kalder en service for at validere et brugerinput af cprnr og password.



Figur 57: Klassediagram for use case realisering af use case 14

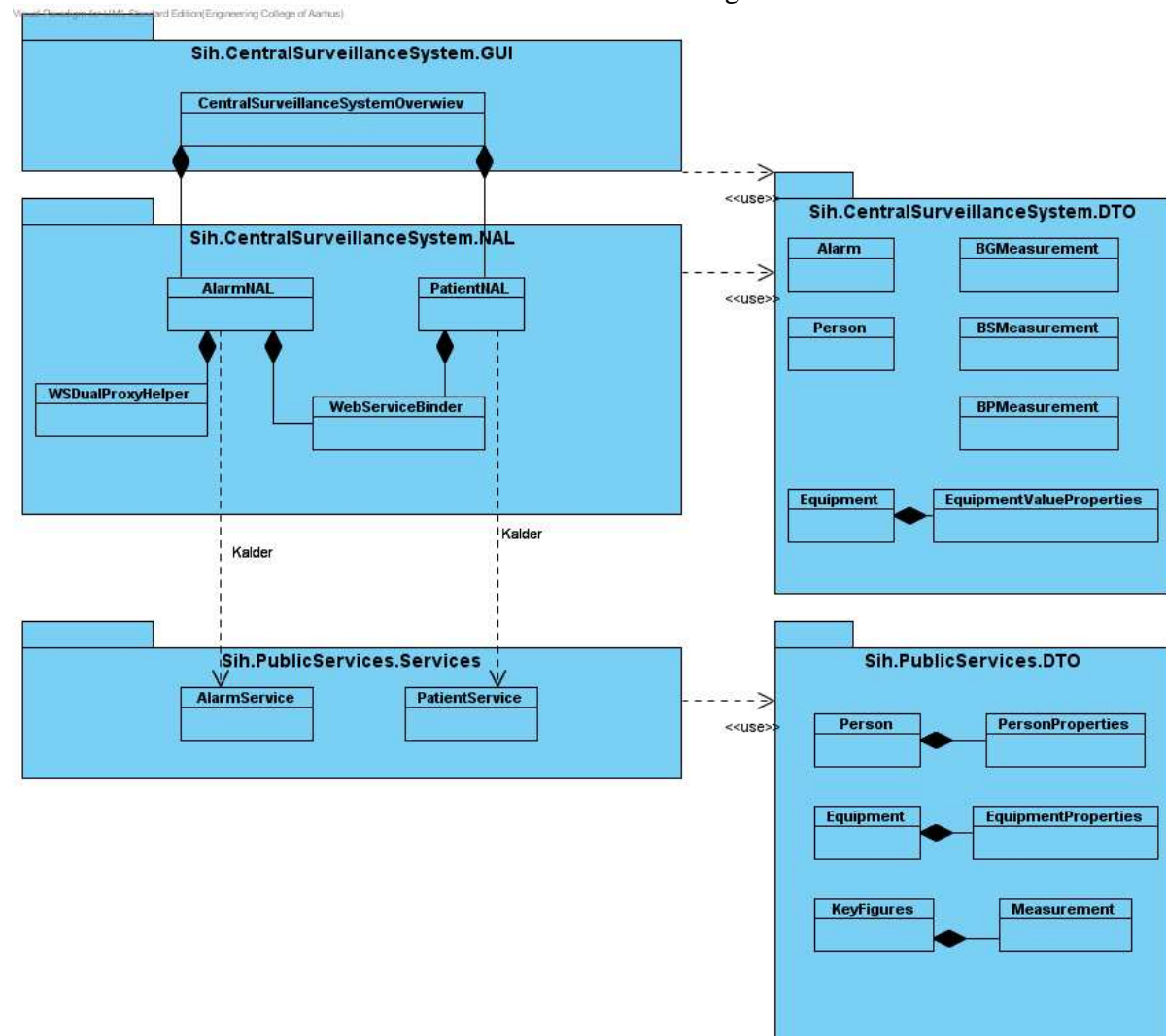
Normalforløb for use case 14:



Figur 58: Sekvensdiagram for use case realisering af use case 14.

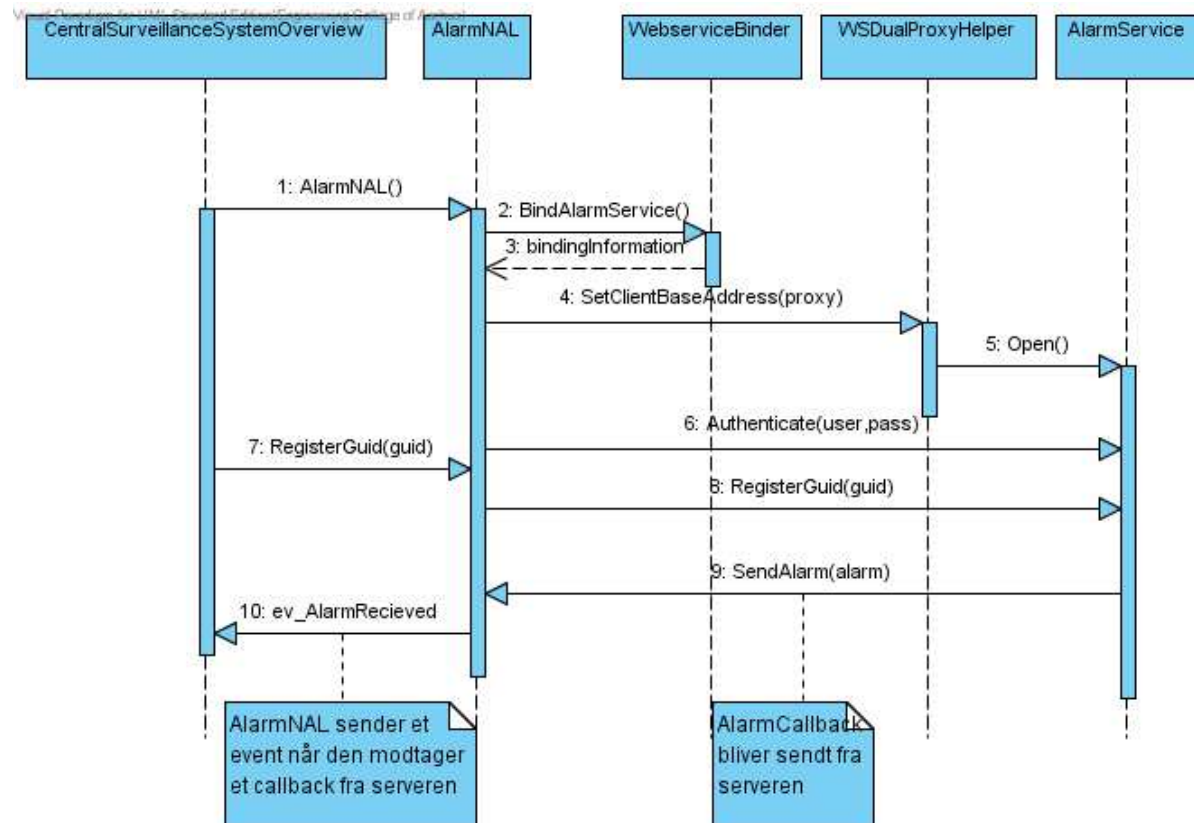
5.3.1.2 Use case 15. Modtag Alarm

Denne use case indeholder et WPF vindue der kan modtage en alarm callback fra serveren.



Figur 59: Klassediagram for use case realiserings af use case 15

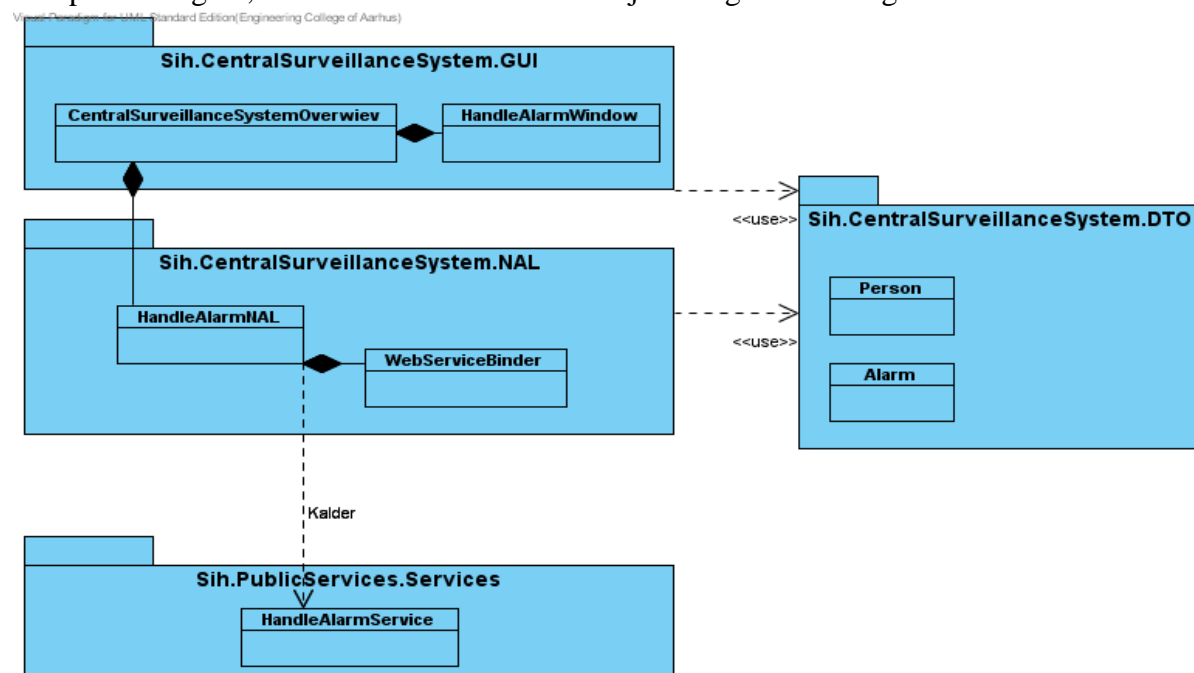
Normalforløb for use case 15:



Figur 60: Sekvensdiagram for use case realisering af use case 15- normalforløb.

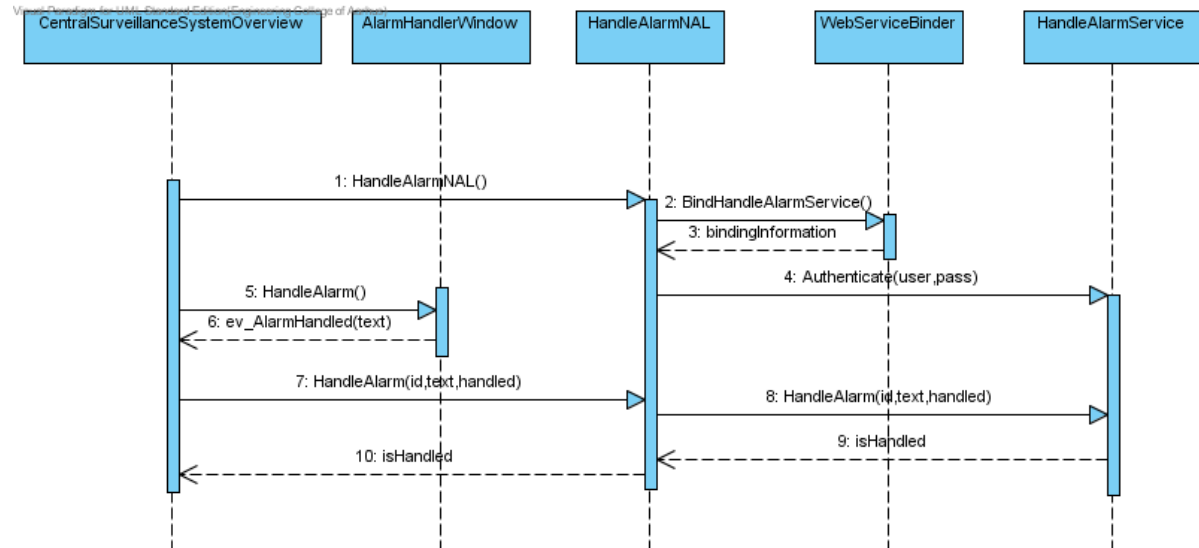
5.3.1.3 Use case 16. Håndter Alarm

Denne use case består af alarm oversigtsvinduet samt et ekstra wpf vindue der kan modtage et input fra vagten, der bliver koblet til alarm objektet og sendt tilbage til serveren.



Figur 61: Klassediagram for use case realisering af use case 16

Program afvikling foregår således:



Figur 62: Sekvensdiagram for use case realisering af use case 16.

5.4 Klassebeskrivelser

Herunde følger en kort beskrivelse til de forskellige klasser der indgår i de forskellige subsystemer. Beskrivelser af klasserne på serveren (på nærASP.NETdelen) kan findes i Systemarkitektur dokumentet for SIH Public Services[3].

Beskrivelser af de enkelte funktioner kan læses som summaries i koden.

I dette afsnit er skrifttypen ændret for at øge læsbarheden.

5.4.1 Subsystem: Central Administration System

5.4.1.1 Login

På WEB-siden Login kan en administrator (eller pårørende) logge sig på systemet. Siden validerer brugeren ved hjælp af kald ned i business logikken. Brugeren bliver derefter gemt på session.

5.4.1.2 AddEquipment

På WEB-siden AddEquipment kan en administrator oprette et nyt stykke udstyr af en type som han selv kan vælge. Efter han har indtastet data for udstyret bliver det sendt til business logikken hvor det bliver valideret.

5.4.1.3 EditEquipment

På WEB-siden EditEquipment kan en administrator redigere et stykke udstyr, som han selv vælger. Efter han har indtastet data for udstyret bliver det sendt til business logikken hvor det bliver valideret. Han kan også vælge at slette udstyret.

5.4.1.4 ShowEquipment

På WEB-siden ShowEquipment kan en administrator få vist oversigten over alt udstyr af en given type i systemet. Han kan derefter trykke på et link for det enkelte stykke udstyr hvorved han bliver sendt videre til EditEquipment siden.

5.4.1.5 AddPersons

På WEB-siden AddPersons kan en administrator oprette en ny person af en type som han selv kan vælge. Efter han har indtastet data for personen bliver den sendt til business logikken hvor det bliver valideret.

5.4.1.6 EditPersons

På WEB-siden EditPersons kan en administrator redigere en person, som han selv vælger. Efter han har indtastet data for personen bliver den sendt til business logikken hvor det bliver valideret. Han kan også vælge at slette personen.

5.4.1.7 ShowPersons

På WEB-siden ShowPersons kan en administrator få vist oversigten over alle personer af en given type i systemet. Han kan derefter trykke på et link for den enkelte person hvorved han bliver sendt videre til EditPerson siden.

5.4.1.8 AddP2ERelation

På WEB-siden AddP2ERelation kan en administrator tilføje en relation mellem en patient og et udstyr.

5.4.1.9 AddP2PRelation

På WEB-siden AddP2PRelation kan en administrator tilføje en relation mellem en patient og en anden person (læge, sygeplejerske, pårørende).

5.4.1.10 ShowRelation

På WEB-siden ShowRelations kan en administrator se en oversigt over alle relationer for en valgt person.

5.4.1.11 AddEquipmentTemplate

På WEB-siden AddEquipmentTemplate kan en administrator oprette en ny udstyrstemplate.

5.4.1.12 EditEquipmentTemplate

På WEB-siden EditEquipmentTemplate kan en administrator redigere i en eksisterende udstyrstemplate.

5.4.1.13 ShowEquipmentTemplate

På WEB-siden ShowEquipmentTemplate kan en administrator få vist en oversigt over alle udstyrs templates i systemet.

5.4.1.14 AddPersonTemplate

På WEB-siden AddPersonTemplate kan en administrator oprette en ny persontemplate.

5.4.1.15 EditPersonTemplate

På WEB-siden EditPersonTemplate kan en administrator redigere i en eksisterende persontemplate.

5.4.1.16 ShowPersonTemplate

På WEB-siden ShowPersonTemplate kan en administrator få vist en oversigt over alle persontemplates i systemet.

5.4.2 Subsystem: Relatives

5.4.2.1 Login

Samme klasse som for Subsystem: Central Administration System

5.4.2.2 Relatives

På WEB-siden Relatives kan en pårørende få vist en oversigt over nøgledata for en given patient relateret til den pårørende.

5.4.1 Subsystem: PatientDataPlugin

Framework.Plugin.Plugin.BLL

5.4.1.1 ByteConverter

Klasse til at konvertere frem og tilbage mellem bytearrays og strings, bruges i forbindelse med measurements.

5.4.1.2 DtoConverter

Klasse der bruges til at mappe PlugIn specifikke DTO'er til WCF DTO'er og vice versa.

5.4.1.3 PatientBLL

Business logik for håndtering af patienter og målinger. Det er også denne klasse der forbinder til en WCF servicen PatientService, og håndterer kald til denne.

5.4.1.4 ReminderBLL

Business logik for håndtering af påmindelser. Det er også denne klasse der forbinder til en WCF servicen ReminderService, og håndterer kald til denne.

5.4.1.5 WebServiceBinder

Klasse der er lavet med det formål at kunne binde en WCF service uden at benytte en app.config fil.

Framework.Plugin.Plugin_DTO

5.4.1.6 BGMeasurement

DTO for blodgasmåling i PlugIn frameworket

5.4.1.7 BPMeasurement

DTO for blodtryksmåling i PlugIn frameworket

5.4.1.8 BSMeasurement

DTO for Blodsuktermåling i PlugIn frameworket

5.4.1.9 ECGMeasurement

DTO for EKG-måling i PlugIn frameworket

5.4.1.10 Person

DTO for en Person i PlugIn frameworket

5.4.1.11 PulseMeasurement

DTO for Pulsmåling i PlugIn frameworket

5.4.1.12 Reminder

DTO for Reminder i PlugIn frameworket

Framework.Plugin.PatientMeasurements

5.4.1.13 BGHistory

Code-behind klasse for interaktions logik til BGHistory wpf UserControl'en

5.4.1.14 BGOOverview

Code-behind klasse for interaktions logik til BGOOverview wpf UserControl'en

5.4.1.15 BPHistory

Code-behind klasse for interaktions logik til BPHistory wpf UserControl'en

5.4.1.16 BPOOverview

Code-behind klasse for interaktions logik til BPOOverview wpf UserControl'en

5.4.1.17 BSHistory

Code-behind klasse for interaktions logik til BSHistory wpf UserControl'en

5.4.1.18 BSOOverview

Code-behind klasse for interaktions logik til BSOOverview wpf UserControl'en

5.4.1.19 ControlInformation

Klasse der implementerer IPublicControlInformation, hvilket er nødvendigt for at plugin'et kan køre på frameworket.

5.4.1.20 ECGOverview

Code-behind klasse for interaktions logik til ECGOverview wpf UserControl'en

5.4.1.21 PersonSelectedEventArgs

Eventarg for event til at vælge personer

5.4.1.22 PluginInformation

Klasse der implementerer `IPublicPluginInformation`, hvilket er nødvendigt for at plugin'et kan køre på frameworket.

5.4.1.23 PublicPlugIn

Klasse der implementerer `IPublicPlugin`, hvilket er nødvendigt for at plugin'et kan køre på frameworket.

5.4.1.24 PulsHistory

Code-behind klasse for interaktions logik til PulsHistory wpf UserControl'en

5.4.1.25 PulsOverview

Code-behind klasse for interaktions logik til PulsOverview wpf UserControl'en

5.4.2 Subsystem: ReminderPlugin Framework.Plugin.PluginReminder

5.4.2.1 AddReminder

Code-behind klasse for interaktions logik til AddReminder wpf UserControl'en

5.4.2.2 ControlInformation

Klasse der implementerer `IPublicControlInformation`, hvilket er nødvendigt for at plugin'et kan køre på frameworket.

5.4.2.3 EditReminder

Code-behind klasse for interaktions logik til EditReminder wpf UserControl'en

5.4.2.4 PersonSelectedEventArgs

Eventarg for event til at vælge personer

5.4.2.5 PluginInformation

Klasse der implementerer `IPublicPluginInformation`, hvilket er nødvendigt for at plugin'et kan køre på frameworket.

5.4.2.6 PublicPlugIn

Klasse der implementerer `IPublicPlugin`, hvilket er nødvendigt for at plugin'et kan køre på frameworket.

5.4.3 Subsystem: Central Surveillance System Sih.CentralSurveillanceSystem.GUI

5.4.3.1 AlarmHandlerWindow

Code-behind klasse for interaktions logik til AlarmHandlerWindow WPF vinduet, der bruges til modtagelse af vagt-input i forbindelse med håndtering af alarm.

5.4.3.2 CentralSurveillanceSystemOverview

Code-behind klasse for interaktions logik til CentralSurveillanceSystemOverview WPF vinduet, der er selve overvågnings systemet der viser alarmer og patient data.

5.4.3.3 LoginWindow

Code-behind klasse for interaktions logik til LoginWindow WPF vinduet, der bruges til modtagelse af vagt-input i forbindelse med login på systemet.

Sih.CentralSurveillanceSystem.DTO

5.4.3.4 Alarm

DTO for Alarmer i overvågningssystemet

5.4.3.5 BGMeasurement

DTO for blodgasmålinger i overvågningssystemet

5.4.3.6 BPMeasurement

DTO for blodtryksmålinger i overvågningssystemet

5.4.3.7 BSMeasurement

DTO for blodsuktermålinger i overvågningssystemet

5.4.3.8 Equipment

EquipmentValueProperties

5.4.3.9 Person

DTO for en Person i overvågningssystemet

5.4.3.10 PulseMeasurement

DTO for Pulsmåling i PlugIn frameworket

Sih.CentralSurveillanceSystem.NAL

5.4.3.11 AlarmNAL

Klasse til at håndtere kald til serveren. AlarmNAL opretter forbindelse til WCF-servicen AlarmService, og håndtere alle kald til denne, inklusive alarm callbacks.

5.4.3.12 ByteConverter

Klasse til at konvertere frem og tilbage mellem bytearray's og strings, bruges i forbindelse med measurements.

5.4.3.13 DtoConverter

Klasse der bruges til at mappe PlugIn specifikke DTO'er til WCF DTO'er og vice versa.

5.4.3.14 HandleAlarmNAL

Klasse til at håndtere kald til serveren. HandleAlarmNAL opretter forbindelse til WCF-servicen HandleAlarmService, og håndtere alle kald til denne.

5.4.3.15 PatientNAL

Klasse til at håndtere kald til serveren. PatientNAL opretter forbindelse til WCF-servicen PatientService, og håndtere alle kald til denne.

5.4.3.16 ValidatePersonNAL

Klasse til at håndtere kald til serveren. ValidatePersonNAL opretter forbindelse til WCF-servicen ValidatePersonService, og håndtere alle kald til denne.

5.4.3.17 WebServiceBinder

Klasse der er lavet med det formål at kunne binde en WCF service uden at benytte en app.config fil.

5.4.3.18 WSDualProxyHelper

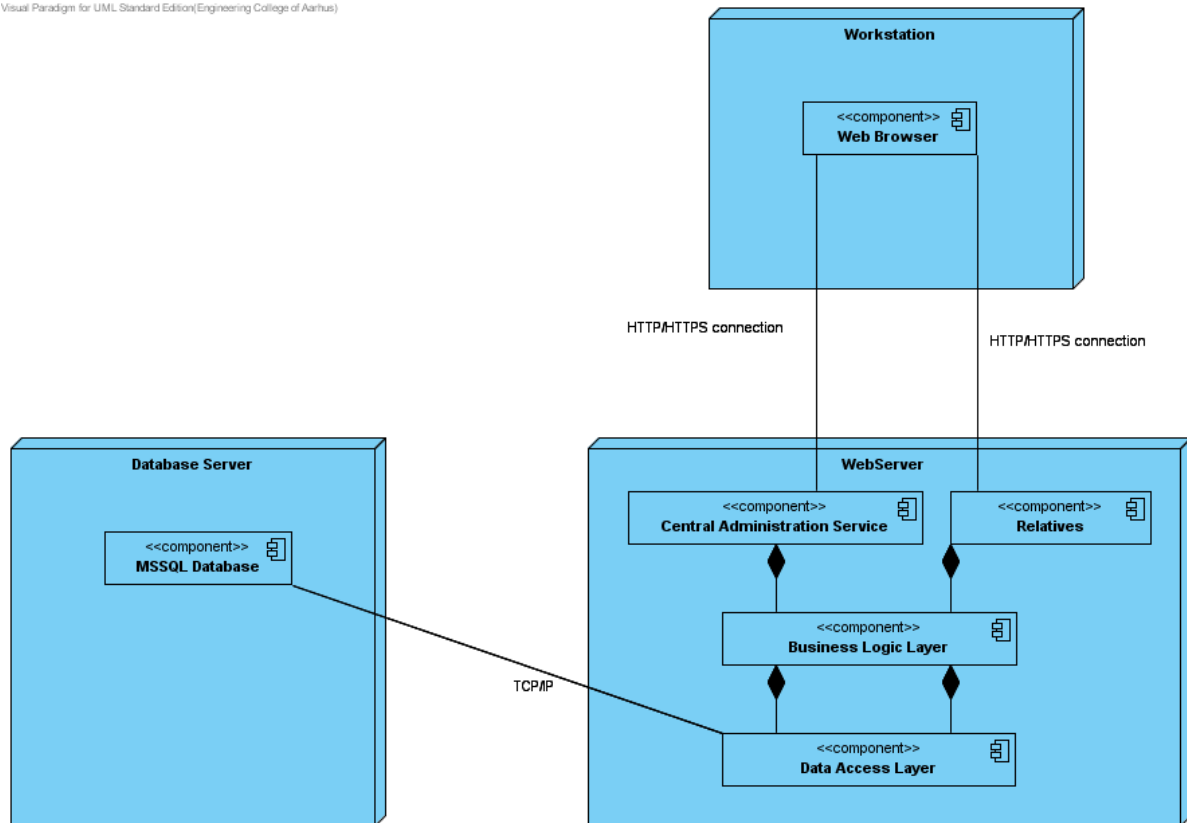
Hjælpeklasse til at finde en ledig port som callbacks fra AlarmService skal gå ind på.

6. DEPLOYMENT VIEW

6.1 Oversight over systemkonfiguration

6.1.1 Deployment af ASP.NET applikationer

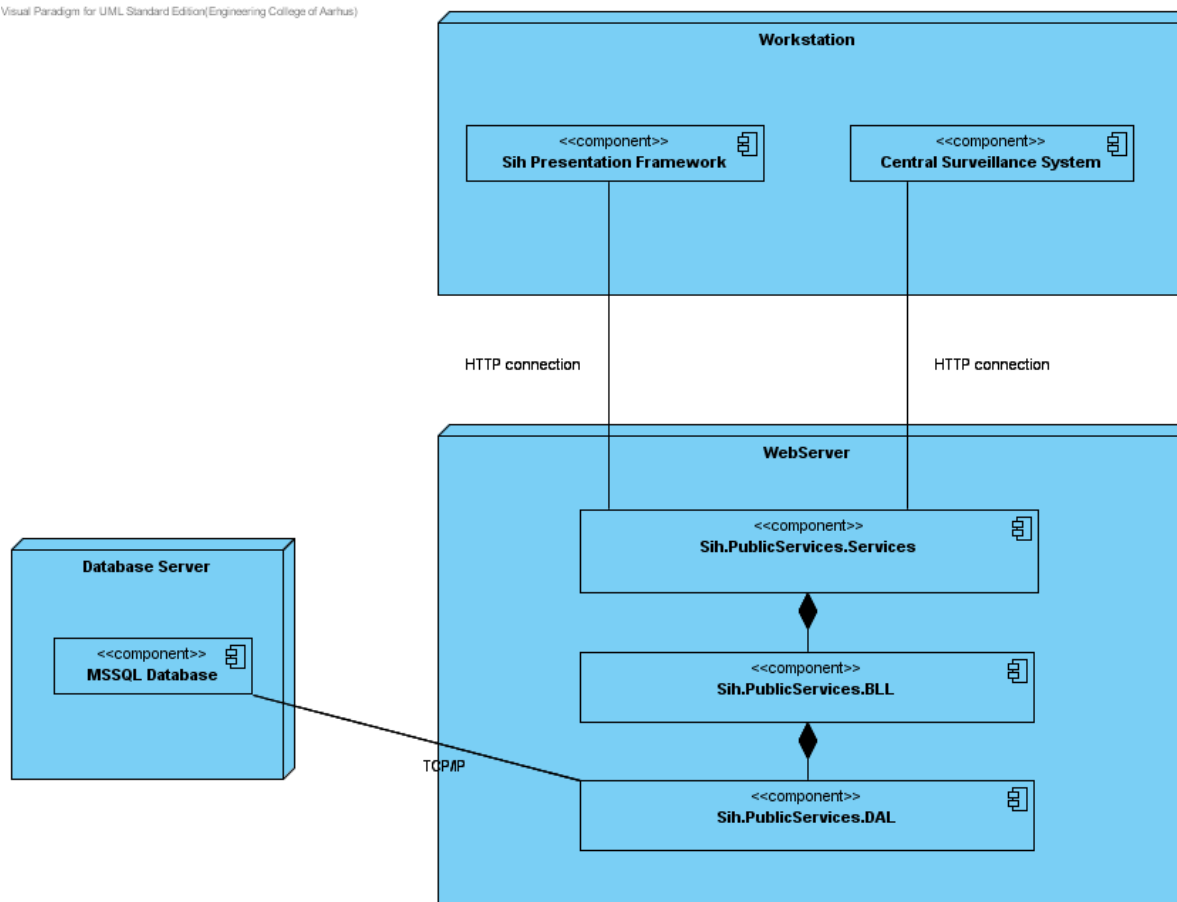
Visual Paradigm for UML Standard Edition(Engineering College of Aarhus)



Figur 63: Deploymentdiagram for HealthCare@Public Web Applikationer.

6.1.1 Deployment af Central Survailliance System og Plugins

Visual Paradigm for UML Standard Edition(Engineering College of Aarhus)



Figur 64: Deploymentdiagram for HealthCare@Public Plugins + Central Surveillance System

6.2 Systemkonfiguration

6.3 Node-beskrivelser

6.3.1 Node Database

Alt statisk data bliver mappet over i en MSSQL 2005 database. Se database design for HealthCare@Central, for information om databasens opbygning.

6.3.2 Node Framework

H@P benytter sig af .NET 3.5 frameworket, og dette er en forudsætning for at H@P kan køre.

6.3.3 Node SQLcmd beskrivelse

Kommunikation link mellem databasen og resten af systemet, hvor alle SQL kommandoer bliver eksekveret fra og data bliver mappe over i de rigtige tabeller

7. Data View

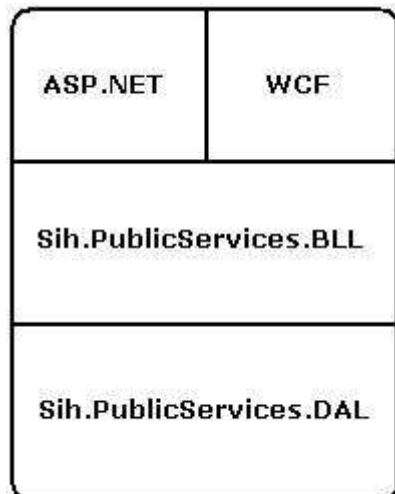
Alt statisk data bliver mappet over i en MSSQL 2005 database som er stillet til rådighed af skolen.

Selve databasens struktur og udvikling er dokumenteret i dokumentet "Database design for HealthCare@Central.

8. Generelle designbeslutninger

8.1 HealthCare@Central struktur

Det blev tidligt i processen valgt at til ASP.NET applikationerne ikke skulle bruges WCF services. Dog er det meget de samme funktioner der skal bruges i disse applikationer som det er der skal tilbydes via WCF services. Derfor blev det valgt at lade disse to applikationer ligge side om side med underliggende lag.



Figur 65: Applikationsstruktur for HealthCare@Central.

Grundet denne struktur er der en højere hastighed på den centrale administration og på den applikation der gør det muligt for en pårørende at logge ind og se data.

8.2 WPF og Plugin

Til præsentation af data for læger og sygeplejersker blev der indgået aftale omkring en plugin struktur. BF gruppen udviklede SIH Presentation Framework der gør brug af de centrale WCF services.

Ud fra de Interfaces der er udbudt af SIH Presentation Framework er påmindelsesmodulet såvel som Se patient data udviklet som plugins.

8.3 Databasestrukturen

Den centrale database placeret på HealthCare@Central database serveren er designet ud fra generiske principper der gør det muligt at tilføje person- og udstyrstyper uden at skulle ændre i database strukturen.

8.4 Implementeringsværktøjer

Følgende værktøjer er benyttet:

Til implementeringen:

- Microsoft Visual Studio 2008
- Microsoft SQL Studio 2005.

Til dokumentation:

- Microsoft Word
- Microsoft Visio 2003 og 2007
- Visual Paradigm for UML 2.0 Enterprise edition

Til Tests:

- Nunit

Andre:

- Tortoise SVN til versionsstyring.

9. STØRRELSE OG YDELSE

HealtCare@Public er implementeret i to forskellige miljøer med to forskellige krav til både afviklingsmiljø og ydelse.

9.1 ASP.NET applikationerne

Skal afvikles på en Windows server med .NET framework 3.5 og IIS 6.0 eller højere installeret.

9.2 Plugins

Skal afvikles på en Windows PC med Windows XP eller nyere version.
Programmet kræver .NET 3.5 Frameworket installeret samt adgang til internettet.

9.3 Central Surveillance System

Skal afvikles på en Windows PC med Windows XP eller nyere version.
Programmet kræver .NET 3.5 Frameworket installeret samt adgang til internettet.

10. KVALITET

10.1 ASP.NET applikationerne

ASP.NET applikationerne udnytter til fulde den generiske database struktur der er beskrevet i Database design for HealthCare@Central.

Der er en god håndtering af exceptions og der har efter endt udvikling ikke været nedetid på disse applikationer.

10.2 Plugins

Desværre har det været svært at opnå en meget høj pålidelighed i de plugins der er udviklet til SIH Presentation Framework. Grunden til dette er at alle har mulighed for at udvikle et plugin med uhåndterede exceptions til dette framework og derved lægge hele frameworket ned.

Til gengæld er det meget åbent for videreudvikling og der er i høj grad udviklet en struktur der gør det let at tilføje mere funktionalitet.

10.3 Central Surveillance System

Pålideligheden af det centrale overvågningssystem er forholdsvis lav da det er en prototype. Kravene har derfor været at få det til at fungere så det kan bruges i en demonstration.