

Undervisningsvejledning til data- og kommunikationsuddannelsen

I dokumentet fremgår målpinde til de 12 programmeringsfag, der blev indført fra juli 2017 og de 27 nye fag, der blev indført fra august 2018 samt en vejledende undervisningsvejledning til disse.

Programmeringsfagene er ikke ændret siden juli 2017. Dog er objektorienteret programmering forlænget med ½ uge, hvorfor faget har fået et nyt nummer.

Fag

Grundlæggende programmering	3
Objektorienteret programmering	4
Databaseprogrammering	5
GUI-programmering	7
Clientsideprogrammering	8
Serversideprogrammering	10
App programmering I	11
App programmering II	12
App programmering III	14
Linux rettet mod server og embedded	15
Versionering og dokumentation	16
Softwaretest og -sikkerhed	17
Netværk I	19
IT Service Management I	22
Computerteknologi	23
Serverteknologi – Databaseserver	24
Serverteknologi	26
Serveradministration og sikkerhed	27
Serverteknologi – Linux	28
Serverteknologi – Web-server	29

Netværk II	31
Serverautomatisering I	33
Praktisk fejlfinding	35
Kundeservice.....	36
Grundlæggende databaseprogrammering.....	37
Netværk III.....	38
Netværkssikkerhed	40
Serverteknologi – Cluster.....	42
Virtualisering	43
Serverautomatisering II.....	44
Netværksteknologi I.....	45
Netværksteknologi II.....	47
Netværksteknologi III.....	49
Backupteknologi	51
Cloudteknologi.....	52
Deployment Service	53
IT Service Management II.....	55
Udvikling	56
Systemudvikling og projektstyring	57

Grundlæggende programmering

v. 1,0

Nummer: 16471-2
Varighed: 10 dage.

Målpinde:

1. Eleven kan anvende et programmeringssprog til at udarbejde mindre konsolprogrammer, herunder fejlfinde og teste dem.
2. Eleven kan vælge en hensigtsmæssig datatype til en opgave.
3. Eleven kan oprette og gøre brug af både enkelt- og multidimensionelle arrays.
4. Eleven kan erklære og anvende simple metoder.
5. Eleven kan diagrammere sit arbejde.
6. Eleven kan benytte forskellige kontrolstrukturer.
7. Eleven kan erklære og bruge variable samt konstanter.
8. Eleven kan redegøre for brugen af kodekommentar.
9. Eleven kan versionsstyre sit arbejde.
10. Eleven kan benytte et debugging-værktøj.
11. Eleven kan vælge hensigtsmæssige datatyper til variable og metoder.
12. Eleven kan erklære og oprette metoder i et objekt med og uden returværdi og parametre, samt gøre brug af disse.

Vejledning:

Eleverne skal møde begrebet coding convention, og at disse har betydning i en samarbejds sammenhæng. Her kan benyttes eksempler hvor eleverne skal opdage hvilken betydning det har for læsbarheden (af andres kildetekst). Eksempelvis ved, at blive præsenteret for uformateret/ustruktureret kildetekst, og blive bedt om, at redegøre for dets virkemåde.

Eleverne skal løse opgaver, hvor de skal træffe valg af datatyper og kontrolstrukturer (do while, while, for, foreach, ...). De skal præsenteres for opgaver der ligger op til brug af et væsentligt udsnit af disse. Eleverne skal erkende, at datatyper kan afhænge af den valgte platform (eg. Int er 16 bit / 32 bit). De skal forstå 2-ers komplement. De skal løse opgaver hvor de møder forskellen imellem e.g. float og double.

Eleverne skal arbejde med valg af datatype ud fra en resourcemæssig betragtning. Eksempelvis lagring af video materiale. Eleven præsenteres for lignende opgaver hvor et valg skal træffes (f.eks. en række tal, hvor den optimale datatype tildeles)

Eleven skal opbygge rutine i, at benytte IDE'et til understøttelse af programmerings processen. Herunder Code-completion, analyse af compiler fejlmeddelelser og debugging funktioner som breakpoints, stepping og watches.

I faget kan benyttes opgaver af forskellig størrelse og hyppighed, alt sammen for, at understøtte en opbygning af rutine det, at skrive programmer. Underviseren kan også give eleverne adgang til et større kode eksempel, der bedre motiverer eleverne, og blot mangler en delmængde af kilden, som eleverne så skal færdiggøre.

Objektorienteret programmering

v. 2,0

Nummer: 16880-2
Varighed: 17½ dage.

Målpinde:

1. Eleven kan anvende et objektorienteret programmeringssprog til at udarbejde konsolprogrammer, der indeholder flere klasser og er i overensstemmelse med OOP konceptet.
2. Eleven har en grundlæggende viden om det valgte programmeringssprog/framework.
3. Eleven kan definere og designe egne klasser.
4. Eleven kan erklære og instantiere objekter.
5. Eleven kan redegøre for typer af collections og kan udpege hensigtsmæssigt i forhold til et behov.
6. Eleven kan anvende en given kodestandard for det pågældende sprog.
7. Eleven kan håndtere "exception handling".
8. Eleven kan redegøre for OOP konceptet såsom indkapsling, polymorfi og arv.
9. Eleven kan udarbejde en applikation som gør brug af OOP konceptet.
10. Eleven kan implementere abstrakte klasser og metoder.
11. Eleven kan skelne mellem override og overload af metoder.
12. Eleven kan begrunde valget af "access modifiers"/virkefelter.
13. Eleven kan oprette og implementere et selvudviklet interface.
14. Eleven kan benytte funktion pointer/callback.
15. Eleven kan udarbejde UML klassesdiagrammer.
16. Eleven kan designe en simpel domænemodel baseret på best practice.
17. Eleven kan redegøre for betydningen af løs kobling og afhængigheder mellem moduler.
18. Eleven kan udføre asynkron programmering med threads, herunder anvende forskellige thread klasser.
19. Eleven kan redegøre for grundlæggende problemstilling med Thread Safety og Atomic State.
20. Eleven kan benytte frameworkets klasser til asynkron programmering, der håndterer problemerne med Thread Safety og synkronisering.
21. Eleven kan oprette en multitrådet applikation samt redegøre for potentielle udfordringer i forhold til tråde, herunder dead locks, live locks og data race.
22. Eleven kan redegøre for mulighederne ved at anvende anonyme metoder og Lambda metoder.

Vejledning:

Målpind nr. 2-10 er til H1. Målpind nr. 11-19 er til H2.

I forhold til arv: hvis man benytter f.eks. C++ at der tales om multipel arv.

Eleven skal afslutningsvis kunne modulere/simulere et kompleks objekt fra den virkelige verden, eks. En vaskemaskine, sodavandsautomat eller instrumentpanel, herunder udarbejdes et klassesdiagram til modelleringen.

Det anbefales at introducere Software Patterns alle relevante steder.

I dette forløb afsluttes med et klasse hierarki, som eleverne skal benytte i det efterfølgende hovedforløb. Dette hierarki kan være et toolkit, eller en objekt model, som eleverne skal gøre brug af på det efterfølgende hovedforløb. Eksempelvis kunne der afsluttende bygges et class library (DLL), der kan benyttes i fremtidige projekter.

Kilden hertil, skal check'es ind i configurations styrings systemet, så det kan check'es ud igen på efterfølgende hovedforløb.

Brugen af configurationsstyring introduceres i dette forløb.

Databaseprogrammering

v. 1,0

Nummer: 16474-3
Varighed: 15 dage.

Målpinde:

1. Eleven kan udarbejde et avanceret databasedesign, og anvende SQL syntaxen og ORM (Object Relational Mapping) til at opbygge databaser, suppleret af en tilhørende dokumentation.
2. Eleven kan anvende SELECT til at foretage forespørgsler op imod en tabel, herunder sortering, gruppering, filtrering, aggregering, funktioner.
3. Eleven kan anvende INSERT, UPDATE, DELETE til at foretage manipulationer af en tabel.
4. Eleven kan oprette, redigere og slette tabeller samt dokumentere vha. E/R-diagram.
5. Eleven kan anvende relationstyperne 1:mange, mange:mange og 1:1 med tilhørende nøgler.
6. Eleven kan implementere konsistenskrav, referenceintegritet, relationer og forskellige constraints.
7. Eleven kan anvende JOIN og SUBQUERIES til at foretage forespørgsler fra flere tabeller.
8. Eleven kan oprette og slette en database.
9. Eleven kan oprette Views og begrunde sikkerhedsaspektet.
10. Eleven kan anvende avancerede SQL-kommandoer som parametre, kontrolstrukturer, rettighedsstyring o.l.
11. Eleven kan redegøre for baggrunden for konsistenskrav, referenceintegritet, relationer og forskellige constraints.
12. Eleven kan oprette, redigere og slette Stored Procedures med et administrations- og databehandlingsformål.
13. Eleven kan oprette og administrere Triggers med henblik på at holde databasen konsistent.
14. Eleven kan træffe korrekte forholdsregler til at imødegå SQL Injection.
15. Eleven kan oprette og administrere indexes så der opnås optimal performance.
16. Eleven kan udføre performance-måling på en stor database.
17. Eleven kan normalisere en database.
18. Eleven har basal viden om ORM (Object Relational Mapping).
19. Eleven kan oprette en domain-model og lade ORM generere databasen og forespørgsler.
20. Eleven kan anvende en færdig database og lade ORM generere domain-modellen.
21. Eleven har et grundlæggende viden om DocumentDatabaser.
22. Eleven kan ud fra et givent performancebehov beskrive fordele og ulemper i forhold til valg mellem relationel database, ORM eller DocumentDatabase.
23. Eleven har en generel viden om mulige sikkerhedsproblematikker omkring databaser, og kan tage højde for dem i forbindelse med opbyggelsen af en database, her f.eks. krypteringen af data og password-beskyttelse af adgangen til databasen.
24. Eleven har en grundlæggende viden om metoder til at teste en nyudviklet database.

Vejledning:

H1 (1 uge) pkt. 2 – 9: Grundlæggende SQL

Omhandler SQL syntaxen og forståelsen af opbygningen af en database, herunder dokumentation.

Gennemføres på færdig database med ca. 5 tabeller. Overalt benyttes SQL-scripts
CASE: Eleven designer, dokumenterer og implementere en mindre database med minimum 5 tabeller (f.eks. et bibliotek) (ingen GUI)

Som database kan benyttes mySql, MS SQL Server eller lignende.

H2 (1 uge) pkt. 10 – 17: Database Design

Omhandler avanceret databasedesign og optimering af performance.

Casen danner grundlag for et interface i form af GUI-programmering (H2)

CASE: Eleven designer, optimerer, dokumenterer og implementerer en større database og der laves performance-målinger. Løsningen skal demonstrere anvendelsen af de tidligere målpinde i faget.

H3 (1 uge) pkt. 18 – 22: Object-Relational Mapping (ORM)

Omhandler ORM (Object Relational Mapping), hvor der genereres dynamisk SQL.

Casen danner grundlag for et interface i form af GUI-programmering (H3)

CASE: Eleven designer en større domain-model (minimum 20 entiteter) med tilhørende entiteter og constraints. ORM frameworket benyttes til at lave diverse queries og manipulationer.

Desuden laves et projekt med en DocumentDatabase, f.eks. MongoDB. Her lægges vægt på forskellen mellem DocumentDatabase og den Relationelle Database, med fordele og ulemper. Dette understøttes med forskellige performance-målinger.

GUI-programmering

v. 1,0

Nummer: 16475-3
Varighed: 12½ dage.**Målpinde:**

1. Eleven kan foretage GUI-programmering rettet mod både clientside-applikationer, og webbaserede applikationer, der bygger på SPA (Single Page Application).
2. Eleven kan implementere en grafisk brugergrænseflade, som understøtter et eller flere funktionelle krav.
3. Eleven kan anvende forskellige typer af layout containers.
4. Eleven kan udvælge og konfigurere de korrekte kontroller til at understøtte et eller flere funktionelle krav.
5. Eleven kan implementere "event-handling".
6. Eleven kan implementere et UI Design pattern.
7. Eleven kan designe, udvikle og anvende "custom-controls".
8. Eleven kan konfigurere Application-klassen og dens lifetime-events.
9. Eleven kan implementere en multitrådet applikation.
10. Eleven kan udføre en simpel usability-test.
11. Eleven kan beskrive arkitekturen for en Single Page Application (SPA) med dens fordele og ulemper.
12. Eleven kan udvikle en SPA client vha. et eller flere SPA frameworks.
13. Eleven kan anvende og konfigurere komponenter, moduler, services osv. i et eller flere frameworks.
14. Eleven kan udvikle en SPA client med flere "sider" vha. Routing.
15. Eleven kan udvikle en SPA client, der vha. HTTP kommunikerer asynkront med en webservice.
16. Eleven kan redegøre for sikkerhedsmæssige udfordringer ved en SPA løsning.
17. Eleven kan debugge en SPA client.

Vejledning:

Faget bygger videre på fagene Objektorienteret programmering, Databaseprogrammering og Clientsideprogrammering.

Faget omhandler to typer GUI:

Den gammeldags Windows brugerflade, som typisk forbindes med et Windows program installeret lokalt på en client-maskine.

En moderne webbrugerflade kaldet SPA (Single Page Application), der afvikles i en browser, men som fungerer så overbevisende at brugeren ikke mærker at der kommunikeres asynkront med en webserver.

H2 (1 uge), pkt. 2 – 10: Windows programmering

Her er tænkt på en Windows applikation, der installeres lokalt. Det kan f.eks. være WPF (Windows Presentation Foundation), Java Swing eller Microsoft UWP.

UI Design pattern, kunne være MVC, MVP, MVVM

Custom-control : Egen udvikling af controls f.eks. ved brug af arv

Localization : Sprogpakker

H3 (1,5 uge) pkt. 11 – 17: Single Page Application (SPA)

Her kan benyttes et JavaScript Framework som Angular (Google), React (Facebook) eller lignende. Denne del af faget kombineres med den del af Serversideprogrammering, hvor man laver Web Services, således at man har et komplet client-server application.

I forbindelse med Angular anbefales det fra starten af at benytte Angular CLI, da det giver den korrekte projektstruktur og løser mange praktiske problemer. Det er også en rigtig god ide at have arbejdet med TypeScript forinden.

Benytter man React gælder det om at have et solidt kendskab til de nyeste udgaver af JavaScript

Clientsideprogrammering

v. 1,0

Nummer: 16476-2
Varighed: 12½ dage.**Målpinde:**

1. Eleven kan foretage grundlæggende web-programmering med HTML, CSS og Bootstrap framework, samt programmering af webapplikationer med JavaScript og jQuery, hvor målet er kommunikation med et backend Web API.
2. Eleven kan forklare Client-Server modellen, herunder HTTP-protokollen.
3. Eleven kan anvende grundlæggende HTML struktur og semantik.
4. Eleven kan anvende lister, links, billeder samt tabeller.
5. Eleven kan anvende HTML formularer.
6. Eleven kan anvende CSS selectors.
7. Eleven kan anvende CSS styling til tekst, links, list og tables m.m.
8. Eleven kan anvende Box modellen.
9. Eleven kan udvikle responsive websider, evt. ved hjælp af et CSS/JavaScript framework.
10. Eleven kan anvende grundlæggende JavaScript kommandoer og strukturer.
11. Eleven kan anvende funktioner og parameteroverførsel.
12. Eleven kan arbejde med JavaScript-objekter.
13. Eleven kan redegøre for væsentlige forskelle mellem JavaScript og et OOP sprog.
14. Eleven kan anvende JavaScript til at give funktionalitet til en webpage.
15. Eleven kan benytte jQuery og forskellige selectors til navigation og manipulation af websidens DOM.
16. Eleven kan benytte jQuery og AJAX til at kommunikere med en backend web Service.
17. Eleven kan beskrive best practise muligheder ved anvendelsen af JavaScript i forhold til både sikkerhed og performance.
18. Eleven har en generel viden om forskellige sikkerhedsproblemer, der kan opstå under afvikling af clientside webapplikationer, som f.eks. cross-site scripting, og kan anvende denne i sit programdesign.
19. Eleven kan gennem test dokumentere funktionaliteten i en udviklet webapplikation.

Vejledning:

Grundlæggende Web-programmering med HTML og CSS, samt Bootstrap framework. Web-applications med JavaScript og jQuery, hvor målet er kommunikation med et backend Web API.

H1 (1 uge) pkt. 2 – 9: Grundlæggende HTML og CSS

Client laver en GET Request, server finder ressourcen frem og sender en Response. Vises i WireShark/Fiddler/DevTools

Tags, Elements og Attributes, Headings, Paragraph. Forskellen mellem Block og Inline elements. Bold/Italic. Strukturen med Head, Body. Udvikling af en simpel HTML page. Semantisk HTML5 markup (nav, header, footer, section, article, aside).

Links, Images, Tables, List

Form og forskellige Form Elements. POST HTTP Response sammlignes med GET HTTP response (QueryString).

Forskellen mellem Inline, Internal og External Style Sheet. Oprettelse og brug af Tag, Class og ID-selectors. Styling af DIV og SPAN. Group Selectors. Websidens DOM. The Cascade Descendent Selectors. Nedarv og håndtering af konflikter. Reset Document. Formatering af Tekst, font, farve, størrelse, formatering af lister og links og tabeller. Background Images. Simpel menu.

Box-modellen med Margin, Padding og Border, herunder Flexbox Floating Elements. CSS Positioning (static, absolut, relativ og fixed).

Her kan benyttes Bootstrap eller måske Googles Material Design. Grid-classes og hvordan de styrer layoutet. Styling af Text, Tables, Images, Jumbotron, PageHeader, Wells og Alerts. Desuden arbejdes med Buttons, Panels og Navigation, samt ikoner med Font Awesome. Andre emner er styling af Forms, Carousel og Modal Plugins.

CASE: Opbyg website med 3 webpages og navigation. Der styles med hhv. CSS og med f.eks. Bootstrap-frameworket.

H2 (1,5 uge) pkt. 10 – 19: Grundlæggende JavaScript og jQuery

CASE: Websiten kan udvides med animation, validering og dynamiske data via AJAX. Diverse open source libraries kan inddrages. Forskellige sikkerhedsproblemer kan indbygges og forskellige løsninger afprøves og undersøges. Hvis der er mere tid, kan man med fordel kigge på TypeScript, som f.eks. benyttes til Angular (SPA websites).

W3Schools har færdige elearning-kurser.

DevTools F12 benyttes overalt hvor det er relevant.

Serversideprogrammering

v. 1,0

Nummer: 16477-3
Varighed: 15 dage.

Målpinde:

1. Eleven kan udvikle serverside webapplikationer, der kan levere HTML-kode til browseren, samt Web API eller webservice, som kan udveksle data med en client-application, f.eks. en browser eller en mobil App.
2. Eleven kan redegøre for forskellige arkitekturer for web Applikationer og web API (web Services), med fordele og ulemper.
3. Eleven kan opbygge og konfigurere en web Application og web API (web service) vha. et framework.
4. Eleven kan benytte validering af brugerinput i en web Applikation.
5. Eleven kan implementere passende ViewModels eller DTO klasser.
6. Eleven kan anvende Unit Test og mocking af objekter.
7. Eleven kan konfigurere routing i en applikation.
8. Eleven kan udvide en applikation med en database, evt. med et ORM-framework.
9. Eleven kan programmere services til brug for en applikation, f.eks. data- og logging-services.
10. Eleven kan benytte en hensigtsmæssig strategi for Exception handling.
11. Eleven kan implementere sikkerhed og brugeradministration i en applikation.
12. Eleven kan udrulle (deploy) en applikation, både On-Premises og Cloud baseret.
13. Eleven kan udføre Parallel Programming.
14. Eleven kan redegøre for fordele/ulemper ved forskellige teknikker inden for Cryptography.
15. Eleven kan anvende Hashing, Symmetric og Asymmetric Encryption.

Vejledning:

Der arbejdes med to typer serversidede webapplikationer: Web Application der leverer en komplet HTML-side til en browser hver gang der ændres state. Web API eller webservice, hvor der kun udveksles data til en client application, som kan være browser, mobil App eller noget helt andet.

H3 (2 uger) pkt. 2 – 15: Web Application og Web API

CASE: Der kan f.eks. benyttes ASP.NET, PHP, JSP MVC frameworket. En Web Application med CRUD kan udbygges med en database/EntityFramework, som bagefter udvides med et antal repositories. Inversion of Control og Dependency Injection er indbygget i frameworket.

Det er vigtigt at benytte IoC (Inversion of Control) og Dependency Injection for at sikre løs kobling mellem lagene.

Samme framework kan også benyttes til at bygge Web API'er, der kan kombineres med faget GUI Programmering. Der kan udvides med asynkrone metoder.

Forslag til projekter: Biograf-system (booking af billetter, oprettelse af film) eller webshop.

H5 (1 uge) pkt. 14 – 15: Kryptering og sikkerhed

CASE: Eleverne kan f.eks. bygge en styring til en vaskehal, hvor flere biler gennemgår forskellige faser af vask på samme tid. Systemet skal kunne vise progressionen og kunne afbrydes. I tilfælde af fejl skal systemet også håndtere nedlukningen korrekt.

Evt. Node.js

Her arbejdes med at kryptere data, både i forbindelse med opbevaring på serveren og i transmissionen mellem server og klient. Så både synkron og asynkron kryptering og SSL er oplagt, samt hashing i forbindelse med opbevaring af passwords og følsomme data på serveren.

Andre vigtige emner kunne være OAuth og OpenIDConnect, som er en meget anvendt standard for hvordan en klient bliver autentikeret og får adgang til ressourcer på serveren. Benyttes meget i forbindelse med WebAPI. Her kan man både benytte AzureAD eller en lokal Identityserver (der findes en opensource Identity-server, der kan anbefales til formålet).

Evt. kan man prøve at introducere Node.js, som er JavaScript der afvikles på serveren.

App programmering I

v. 1,0

Nummer: 16478-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan udvikle en simpel app til mobile enheder.
2. Eleven kan redegøre for filstrukturen i et native app-projekt i udviklingsværktøjet.
3. Eleven kan redegøre for en app's Life Cycle, dvs. hvilke metoder der udføres når en app startes, skjules, vises og afsluttes.
4. Eleven kan programmere en simpel brugerflade, eksempelvis en knap og et tekstfelt.
5. Eleven kan med et grafisk værktøj designe en brugerflade med de mest almindelige elementer.
6. Eleven kan redegøre for og anvende relevante eventhandlers for brugerfladen.
7. Eleven kan via programkode starte andre programmer/processer på enheden f.eks. en browser.
8. Eleven kan benytte dialogboks til beskeder og svar fra brugeren.
9. Eleven kan oprette forskellige brugerflader i samme app, hvor der kan overføres data imellem disse.
10. Eleven kan beskrive de væsentligste forskelle på udviklingsmiljøer til forskellige mobilplatforme.
11. Eleven kan dokumentere og kommentere sin kode.
12. Eleven kan gennem en brugertest dokumentere funktionaliteten og brugeroplevelsen i en udviklet app.

Vejledning:

App Programmering 1 – 3 kunne være et fortløbende projekt, der afsluttes på App 3. Der kan evt. benyttes kommunikation med systemer, der er udviklet i de øvrige programmeringsfag.

Tilføjelser til målpindene.

1. Tanken bag faget er at eleven lærer at programmere til en mobil enhed, f.eks. en Android enhed, og får rutine i at arbejde med strukturen i koden på den platform, som eleven arbejder med.
2. Eksempelvis hvordan mappestrukturen i projektet er opbygget under udviklingsfasen, og hvordan den ser ud i den installerbare app.
3. Hvilke metoder udføres når en app startes, skjules, vises og afsluttes, og hvad ville være relevant at gøre i disse metoder.
4. Det er meningen at eleven får forståelse for, hvordan man opbygger en grafisk brugerflade uden anvendelse af grafisk designværktøj...
5. ... og med anvendelse af grafisk designværktøj. Hvordan evt. XML-kode oversættes til programkode.
6. Eventhandlere kan evt. tilknyttes i designdelen eller tilknyttes i den egentlige programkode. Begge metoder skal bruges. Det er også muligt at oprette eventhandlers på forskellig måde i koden (Android Java: Anonyme klasser, eksterne klasser, indre klasser)
7. En meget simpel måde på at starte en anden app, f.eks. en browser, hvor der overføres URL på den side man vil se.
8. En dialogboks vises som en pop-up, hvor der kan indtastes en tekst, vælges Ja eller Nej osv. Det kan også være et hint (Android Toast) der vises.
9. En anden brugerflade skal kaldes sådan, at når den afsluttes, returneres brugeren til den oprindelige side. Der skal overføres data i begge retninger. F.eks. overfør navn og telefonnummer til den kaldte nye side, ret disse, tryk ok og returner de rettede oplysninger til hovedsiden.
10. Eleven ved at der er forskellige udviklingsmiljøer og at der kodes i forskellige sprog til Android og iOS
11. Der kan f.eks. bruges JavaDoc, som genererer HTML ud fra kommentarer.

App programmering II

v. 1,0

Nummer: 16479-3
Varighed: 5 dage.

Målpinde:

1. Eleven kan udvikle en appløsning, der kan afvikles på forskellige mobile platformsteknologier.
2. Eleven kan installere, konfigurere og anvende et cross-platformsudviklingsmiljø.
3. Eleven kan programmere en appløsning med menuer.
4. Eleven kan programmere en appløsning, der kan vise dynamisk oprettede lister på skærmen og reagere på brugerens valg.
5. Eleven kan programmere en appløsning, der kan gemme persistent data lokalt på enheden.
6. Eleven kan programmere en appløsning, der kan anvende enhedens sensorer f.eks. gyroskop, accelerometer, temperatur osv.
7. Eleven kan tilpasse en responsiv brugerflade, der kan orientere sig efter landskab eller portræt.
8. Eleven kan programmere en appløsning, der kan anvende touch-input fra enhedens skærm, og ved berøring flytte et grafisk objekt rundt på skærmen.
9. Eleven kan programmatisk få et grafisk element til at bevæge sig kontinuerligt på skærmen.
10. Eleven kan udvikle et program, hvor sensorer, touch-skærmen og objekter, der bevæger sig, fungerer sammen på en enkel måde.
11. Eleven kan benytte f.eks. Repository Pattern og Dependency Injection til minimering af afhængigheder mellem moduler.
12. Eleven kan benytte Inversion of Control Pattern (f.eks. Dependency Injection).
13. Eleven kan programmere en brugergrænseflade med asynkrone kald.
14. Eleven kan programmere asynkrone server services i forbindelse med web API.
15. Eleven kan gennem en brugertest dokumentere funktionaliteten og brugeroplevelsen i en udviklet appløsning.
16. Eleven kan redegøre for mulige sikkerhedsproblemer, der skal tages højde for i forbindelse med udvikling af appløsninger.

Vejledning:

App Programmering 1 – 3 kunne være et fortløbende projekt, der afsluttes på App 3. Der kan evt. benyttes kommunikation med systemer, der er udviklet i de øvrige programmeringsfag.

Tilføjelser til målpindene.

1. En app, der kan afvikles på flere platforme og med en vis kompleksitet af brugerflade, grafik, sensorer og berøring.
2. Dette kan være en omfattende øvelse. Udvikling i Xamarin giver mulighed for i C# at benytte MVVM-lagdelingen. Ved udvikling til flere platforme skal overvejes native/ikke-native (HTML)
3. Med menuer menes fortrinsvis statiske drop-down/pop-up menuer, hvor der sker et generelt valg.
4. Med lister i brugerfladen menes, at vise en række af data, f.eks. navne på personer, hvor der kan vælges en enkelt. Items i en sådan liste kan være mere avanceret, f.eks. med billede, tekst og checkbox, hvor det er muligt vælge flere items på en gang i listen.
5. Persistent data kan være lokal database (SQLite), filer eller andre specifikke muligheder på enheden, (f.eks. Android Shared Preferences).
6. - Der kan f.eks. styres en kugle på skærmen ved at lægge enheden fladt ned og så lade kuglen løbe nedad når enheden hældes.
- Der kan ved at ryste enheden styres en udvikling, f.eks. at en blomst springer ud.
7. Ideen er at have ens brugerflader henholdsvis tilpasset portræt og landskab, og vælge den aktuelle når enheden drejes.
8. Der kan registreres om berøringen sker på et objekt, hvordan det beregnes, om det sker indenfor f.eks. en cirkel. Ved efterfølgende flytning af berøringen, kan objektet flyttes. Tænkes gjort med egentlig grafik, (hvor paint-metoden er overridet).

9. Til automatisk at flytte et objekt kontinuerligt, kan bruges en uendelig løkke i en ny tråd, en timer, eller at paint-metoden udfører et forsinket kald af sig selv.
10. Et simpelt "spil", hvor punkt 8 og 9 benyttes, så der både sker noget automatisk og brugeren kan gøre noget. Der skal benyttes en form for check af kollisioner.

App programmering III

v. 1,0

Nummer: 16480-3
Varighed: 5 dage.

Målpinde:

1. Eleven kan udvikle en appløsning, som manipulerer eksterne data, herunder f.eks. fjernstyrer ting over internettet, og husker opsætninger imellem flere udførsler af den pågældende app.
2. Eleven kan selvstændigt udvikle en appløsning.
3. Eleven kan vælge det bedst egnede udviklingsværktøj til en given opgave, under hensyntagen til crossplatform-kompatibilitet og performance.
4. Eleven kan programmere en appløsning, der kan kommunikere med eksterne datakilder, så som databaseserver, web API o.l.
5. Eleven kan programmere en appløsning, der kan kommunikere med andre apps på enheden ved f.eks. at opdatere en kontakt, sende en SMS eller hente GPS-data.
6. Eleven kan programmere en appløsning med threads med det formål at udføre parallelle opgaver.
7. Eleven kan optimere koden i appløsningen med henblik på optimal udnyttelse af enhedens ressourcer.
8. Eleven kan redegøre for mulige sikkerhedsproblemer når en app skal kommunikere med eksterne datakilder og tage højde for disse i forbindelse med udvikling af en appløsning ved f.eks. at kryptere.
9. Eleven kan gennem en test dokumentere funktionaliteten i en udviklet appløsning.

Vejledning:

App Programmering 1 – 3 kunne være et fortløbende projekt, der afsluttes på App 3. Der kan evt. benyttes kommunikation med systemer, der er udviklet i de øvrige programmeringsfag.

Tilføjelser til målpindene.

1. Eleven skal lave en app, der er mere avanceret end de apps, eleven tidligere har udviklet i de forgående app programmerings fag.
2. Her menes der at eleven selv bør kunne udvikle sit projekt, uden hjælp fra underviseren. Det betyder naturligvis ikke at eleven ikke kan spørge om vejledning, men kan stå på egne ben det meste af tiden.
3. Eleven kan selv tage stilling til, og argumentere for, hvilke(t) OS projektet skal udvikles til, og hvilket udviklingsværktøj eleven vil bruge.
4. Eleven kan udvikle et projekt, der kan kommunikere og veksle data med eksterne enheder, så som en databaser server, cloud-services, web-api eller lignende.
5. Eleven kan indhente data fra eksempelvis enhedens GPS, kontaktbog, SMS tjeneste, Wi-Fi system, eller andre midler som enheden har tilgængeligt.
6. Eleven kan oprette nye tråde i programmet, således at der kan udføres en eller flere separate opgaver parallelt.
7. Eleven kan strukturere sin kode, sådan at den ikke benytter unødvendige ressourcer fra enhedens hardware. Eksempelvis, hvordan man kan genbruge objekter eller variabler i programmet, eller kan spare på enhedens batteri.

Linux rettet mod server og embedded

v. 1,0

Nummer: 16481-2
Varighed: 5 dage.

Målpinde:

1. Eleven har en generel viden om Linux miljøet, og kan anvende denne viden i forbindelse med opgaver rettet mod Linux baserede systemer.
2. Eleven kan navigere i et Linux shell miljø og anvende almindeligt forekommende Linux kommandoer.
3. Eleven har en grundlæggende viden om Linux distributioner og Linux open source miljøet.
4. Eleven har en grundlæggende viden om Linux opbygning og Linux kernen.
5. Eleven kan anvende Linux programmeringsmiljøet.
6. Eleven kan anvende værktøjer og programmeringsmodellen til Linux.
7. Eleven kan anvende programmeringsmodellen til forskellige Linux varianter, herunder Linux server og embedded Linux.
8. Eleven kan anvende Linux dokumentationen til selvstændigt at finde nødvendig information.
9. Eleven kan dokumentere egne løsninger i Linux dokumentationssystemet.

Vejledning:

Overvejelser om faget.

- Faget bør lægges tidligt i forløbet, så eleverne har mulighed for at arbejde på både Linux og Windows platformen.
- Tanken bag dette fag er at eleven får kendskab til Linux programmerings miljøet og kan anvende denne viden i forskellige sammenhænge hvor Linux anvendes. For eksempel til Server eller embeddede Linux systemer.
- Dette fag ligger på H5 og vi formoder at eleven har haft alle fastlagte programmeringsfag til og med H4.
- Vi har lagt vægt på der ikke er nævnt specifikke programmeringssprog eller metoder.
- Programmering af Linux drivere og programmering til kernekald er næsten helt uafhængig af valg af programmeringssprog.

Versionering og dokumentation

v. 1,0

Nummer: 16483-2
Varighed: 2½ dage.

Målpinde:

1. Eleven kan redegøre for hvorfor og hvordan man benytter et versionsstyringsværktøj.
2. Eleven kan redegøre for formålet med udarbejdelse af dokumentation.
3. Eleven kan beherske dokumentation under programudvikling.
4. Eleven kan anvende et versionsstyringsværktøj under programudvikling.
5. Eleven kan begrunde behovet for dokumentation og brug af versionsstyringsværktøj under programudvikling.

Vejledning:

Overvejelser om faget

Faget afvikles bedst, hvis der er kildekode tilstede, som kan versioneres og dokumenteres.

Kildekoden kan være fra skolen eller fra virksomheden, hvis det tillades.

Faget kan med fordel integreres/køre parallelt med et af programmeringsfagene:

Grundlæggende programmering eller Objektorienteret programmering

Eleven præsenteres for et udokumenteret / dokumenteret program (der virker), så forskellen træder tydeligt frem. Diskutere forskellen, evt. gennemgang af programmet før og efter.

Faget bør være medtænkes i Programmeringsmetodik, så eleverne hurtigt kan arbejde med en udviklers vigtigste værktøjer

Softwaretest og –sikkerhed

v. 1,0

Nummer: 16484-2
Varighed: 7½ dage.

Målpinde:

1. Eleven har en generel viden om metoder og løsninger til at fejl- og funktionsteste en udviklet softwareløsning, samt om løsninger, der kan forbedre sikkerheden på en web-enabled applikation.
2. Eleven kan tilrettelægge en testplan, som indeholder test til funktionelle krav.
3. Eleven kan anvende forskellige relevante testmetodikker.
4. Eleven kan vurdere, hvornår en software-release er tilstrækkeligt testet.
5. Eleven kan gennemføre en scriptet test på en app- eller webplatform.
6. Eleven kan integrere automatiserede tests i software udviklingscyklussen.
7. Eleven kan gennemføre en tryktest, hvor performance og krav til hardware medtages.
8. Eleven kan aflæse performancetabeller og identificere flaskehalse i et kompliceret system.
9. Eleven kan redegøre for UX-tests (User Experience), som evaluerer kundeoplevelsen.
10. Eleven har en generel viden om data-kryptering, og kan anvende krypteringsformer som f.eks. SSL i forbindelse med udvikling af webbaserede applikationer.
11. Eleven kan indhente viden omkring softwaresikkerhedsstandarder og kendte sikkerhedshuller i eksisterende software.
12. Eleven har en generel viden om authentication og kan udvikle programmer, der anvender authentication.
13. Eleven har en generel viden om Hashing-kodning, og kan anvende kodningsformer som f.eks. MD5 i forbindelse med udvikling af webbaserede applikationer.
14. Eleven har en generel viden om forskellige former for hacking angreb, som f.eks. SQL injection og XSS (Cross-Site Scripting), og kan anvende denne viden til at sikre egne udviklede webbaserede applikationer.
15. Eleven kan anvende forskellige website security tools til at teste sikkerheden på udviklet software.

Vejledning:

Faget bør ligge tidligt i forløbet.

Softwaretest bør være repræsenteret indgå Programmeringsmetodik.

Faget indeholder to emner.

Softwaretest: målpind 2-9, ½ uge.

Der skal arbejdes med softwaretest i de programmeringssprog, som eleverne møder.

Ad 2) I udarbejdelsen af testplanen bør der lægges vægt på at planen er let at vedligeholde og at testplanen kan udføres af en anden/tredje part.

Ad 3) Testmetodikkerne kan f.eks. være grænseværdianalyse, divide and conquer, test cases vs. test scenarier og testplaner til forskellige formål (release test, acceptance test etc.).

Ad 4) Test af ikke funktionelle krav - hvilke "bløde" krav bør der stilles til den testede software og hvordan testes de - f.eks. startup tider, typiske behandlingstid, ram forbrug etc.

Ad 5) Der kan tages udgangspunkt i gratis software som Selenium/Appium (web & Apps) eller testcomplete som værktøj til test af windows software eller sauce labs som multiplatform test værktøj til websider.

Ad 6) Eksempel på hvorledes automatiseret test kan integreres i softwarens bygge cyklus, således at testen køres hver gang der f.eks. tilføjes ny kode til softwaren (continuous integration)

Ad 7) Tryktest af systemer - hvorledes testes for rå performance og hvor meget kræver det af den hardware der skal teste systemet.

Ad 8) Eksempler kan laves i Apache jMeter. Performance test er der hvor de helt store penge ligger - det er en kunststart!

Ad 9) Der findes værktøjer, som benyttes når softwaren er release't og man vil vide hvordan kunderne oplever den (crashes, performance etc.) gennem systemer som f.eks. Raygun eller Fabric.

Softwaresikkerhed: målpind 10-15, 1 uge.

Der anvendes 1 uge på sikkerhed med fokus på serverside- og app-programmering og evt. med gennemgang af tidligere programmer med sikkerhed for øje.

Forslag til relevant indhold:

- Til introduktion kan f.eks. bruges "Cæsar kryptering" til at illustrere de grundlæggende ideer i hemmeligholdelse og til at bringe terminologien på plads.
- Single factor/two-factor/multiple factor authentication (viden, ejerskab, iboende)
- Grundlæggende gennemgang af idéen bag et public-key kryptosystem.
- Angreb på kryptosystemer (Ciphertext Only Attack, Known Plaintext Attack, Chosen Plaintext Attack, Brute Force Attack, Man In The Middle Attack etc.)
- Introduktion og illustration af gængse sårbarheder i software, såsom SQL-injection, denial-of-service angreb, buffer overflows mm.
- Sikkerhed tænkes ind i design og test af software
- CERT Top 10 Secure Coding Practices
- Eksempler på organisationer: CERT, OWASP, NIST
- Eksempel på standard: ISO 27034 (det væsentlige er her eleven får en forståelse af at den slags standarder eksisterer – ikke det detaljerede indhold)
- Brug af security bulletins
- Se f.eks. <http://www.creativebloq.com/web-design/website-security-tips-protect-your-site-7122853>

Forslag til øvelser:

- Implementer en simpel krypteringsalgoritme, f.eks. en substitutionsalgoritme. Lad eleverne prøve at (skrive et program til at) bryde hinandens krypterede beskeder.
- Brug af et værktøj som f.eks. Wireshark bruges til at illustrere hvor let det i praksis er at opsnappe data sendt i klartekst mellem f.eks. en web server og en web browser.
- Etabler på forhånd et simpelt setup med en web server og en simpel klient. Som er sårbar overfor "SQL injection". Lad eleven indføre inputvalidering, så sårbarheden fjernes.

Lad eleven bladre f.eks. en CERT security bulletin igennem for at opnå en fornemmelse af hvilke information en sådan indeholder

Netværk I

v. 1,0

Nummer: 16852-2
Varighed: 10 dage.

Målpinde:

1. Eleven kan selvstændigt eller i samarbejde med andre elever løse en arbejdsopgave omhandlende en routet netværksløsning.
2. Eleven kan beregne hvilken vej trafikken vil tage igennem netværket, baseret på output fra routertabeller i IPv4 og/eller IPv6.
3. Eleven kan på baggrund af en subnet/wildcard maske beregne et IP-range.
4. Eleven kan forklare hvordan en switch fungerer og anvendes i et firmanetværk.
5. Eleven kan konfigurere basale sikkerhedsfunktioner på en router/switch f.eks. SSH og passwords.
6. Eleven kan konfigurere en switch med VLANs, port-security og trunking.
7. Eleven kan implementere Inter-VLAN routing vha. en router eller en L3 Switch.
8. Eleven kan implementere statisk routing mellem to routere på IPv4 og/eller IPv6.
9. Eleven kan forklare funktionen af og konfigurere en default route.
10. Eleven kan konfigurere og idriftsætte en dynamisk routerprotokol (f.eks. RIP).
11. Eleven kan forklare forskellen på dynamiske routerprotokoller (IGP/EGP – Distance Vector/Link State).
12. Eleven kan konfigurere DHCP server på en router.
13. Eleven kan konfigurere og anvende standard Access-lister (ACL) i forbindelse med Network Address Translation (NAT) og simpel IP trafikregulering.
14. Eleven kan implementere NAT på en router.

Vejledning:

Niveau og indhold svarer til CCNA R&S Routing and Switching Essentials v6 eller nyere.

Med udgangspunkt i Netacad's CCNA - R&S Routing and Switching Essentials v6.0 kursus:

Teoribog CCNA R&S Routing and Switching Essentials v6 Companion Guide.”

Planen for gennemlæsning af bogen se dagsplanen.

Det kan forventes at du hver dag har hjemmeopgaver der skal løses.

Læseskema Netværk I (CCNA R&S RSE)

De angivne øvelser er minimum – Men mangler de ekstra opgaver så hav nogle ekstra i baghånden.

Arbejdsbeskrivelse:

Råd til eleven:

1. 1. **Forud** for emne – dvs. *inden du møder* til undervisning - Se eventuelle youtube videoer som handler om det emne du skal have næste dag.
www.youtube.com/NetworkMadeEasy
2. Lav projektopgave/fejlfinding på fysisk udstyr (gruppearbejde GNS3/PT kan evt. bruges når du ikke er i klassen. Se evt. videoer der kan hjælpe dig med hvordan GNS3 installeres.
3. Læs de angivne sider i bogen.
4. **Tag kapiteltests første gang** (gruppearbejde) undlad at svare på spørgsmål du intet ved om
Se hvilke emner du svarede forkert på i kapiteltesterne og læs (max 30 min.) Tag kapiteltest igen (hjemmearbejde - kan undlades hvis 100% første gang)
5. Gennemgår de spørgsmål i kapiteltesten der volder problemer. Hvis der er spørgsmål/emner i kapiteltesten der er svære at forstå så kig nærmere på problemstillingen sammen.

Dag	Emne	Øvelse m.m.
1	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner	Lab 1.1.4.6 - Configuring Basic Router Settings with IOS CLI

Dag	Emne	Øvelse m.m.
	8.00 - 13.15 "Routing Concepts + Static Routing"	
2	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 08.00 - 15.00 "Dynamic Routing" Opsamling ch 1+2 test	Lab 2.2.2.5 - Configuring IPv4 Static and Default Routes Lab 2.2.4.5 - Configuring IPv6 Static and Default Routes Lab 2.3.2.4 - Troubleshooting IPv4 and IPv6 Static Routes
3	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 08.00 - 15.00 Opsamling ch 3 test	Lab 3.2.1.9 - Configuring Basic RIPv2
4	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 09.45 - 15:00 "Switched Networks + Switch Configuration"	Lab 5.1.1.6 - Configuring Basic Switch Settings Lab 5.2.2.9 - Configuring Switch Security Features
5	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 - 12.30 "VLANs" Opsamling ch 4+5 test	Lab 6.2.2.5 - Configuring VLANs and Trunking Lab 6.2.3.9 - Troubleshooting VLAN Configurations Lab 6.3.2.4 - Configuring Per-Interface Inter-VLAN Routing Lab 6.3.3.7 - Configuring 802.1Q Trunk-Based Inter-VLAN Routing
	kapiteltestene åbnes hele weekenden	
6	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 - 13.15 "Access Control Lists" Opsamling ch 6 test	Lab 7.2.2.6 - Configuring and Modifying Standard IPv4 ACLs Lab 7.2.3.4 - Configuring and Verifying VTY Restrictions
7	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 - 15.00 "DHCP + NAT for IPv4" Opsamling ch 7 test	Lab 8.1.2.4 - Configuring Basic DHCPv4 on a Routers Lab 8.1.2.5 - Configuring Basic DHCPv4 on a Switch Lab 8.1.4.4 - Troubleshooting DHCPv4 Lab 8.2.3.5 - Configuring Stateless and Stateful DHCPv6 Lab 8.2.4.4 - Troubleshooting DHCPv6 Lab 9.2.2.6 - Configuring Dynamic and Static NAT Lab 9.2.3.7 - Configuring Port Address Translation (PAT) Lab 9.3.1.5 - Troubleshooting NAT Configurations
8	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 - 15.00 "Device Discovery, Management, and Maintenance" Opsamling ch 8+9 test	Lab 10.1.2.5 - Configure CDP and LLDP Lab 10.2.3.6 - Configuring Syslog and NTP Lab 10.3.1.9 - Managing Router Configuration Files with Terminal Emulation Software Lab 10.3.1.10 - Managing Device Configuration Files Using TFTP, Flash, and USB

Dag	Emne	Øvelse m.m.
		Lab 10.3.1.10 - Managing Device Configuration Files Using TFTP, Flash, and USB
9	8.00 – 15.00 Praktisk prøve	
10	8.00 – 13.15 Teoretisk prøve, oprydning, feedback og karaktergivning	

IT Service Management I

v. 1,0

Nummer: 16853-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan redegøre for de almindelige roller, der findes i og omkring 1st level support, især bruger, kunde, 1st line analytiker og 2nd level support.
2. Eleven kan udarbejde en rollebeskrivelse for 1st line analytiker, der indeholder de primære arbejdsopgaver, ansvar, beføjelser, nødvendige kompetencer og personlige egenskaber.
3. Eleven kan redegøre for brugerens og kundens forventninger til 1st level support, herunder beskrive og afstemme forventninger såvel uformelt som formelt i form af aftaler.
4. Eleven kan selvstændigt løse sager under hensyntagen til formelle eller uformelle forventninger, og tage ansvar for brugerens situation og oplevelse – også når en sag skal eskaleres eller håndteres af andre.
5. Eleven kan med udgangspunkt i brugerens beskrivelse skelne mellem service requests og fejl samt registrerer disse i relevante sagsstyringssystemer herunder udvælge eller angive en sigende kategori og prioritere sagen ud fra dens alvor og konsekvenser for forretningen,
6. Eleven kan selvstændigt vurdere, hvornår en sag er løst tilfredsstillende og kan lukkes, og hvornår sagen skal sendes videre for yderligere behandling.
7. Eleven kan selvstændigt løse service request opgaver (herunder brugeradministration) til brugernes tilfredsstillelse ved at udvælge, forstå og følge den relevante procedure
8. Eleven kan nedbryde og beskrive en service request opgave (herunder brugeradministration) i en struktureret trinvis procedure, som andre kan følge
9. Eleven kan udarbejde et katalog over service requests inklusiv beskrivelser, betingelser, serviceniveauer og bestillingsprocedurer.
10. Eleven kan indsamle, vurdere, sortere og registrere relevante symptomer og informationer ved fejl og henvendelser om hjælp
11. Eleven behersker minimum to af følgende problemløsningsmetoder: 5-Whys, Ishikawa, Kepner-Tregoe, Swarming, Pareto Analysis, Brain-storming og Affinity Mapping
12. Eleven kan udarbejde problembeskrivelser, der tydeligt angiver de berørte objekter med tilhørende afvigelser samt indsamlede symptomer og gennemførte problemløsningsaktiviteter
13. Eleven kan finde, vurdere og anvende eksisterende viden og dokumentation til problemløsning og support

Vejledning:

Faget omfatter kun aktiviteter, der løses af 1st level support.

Det er en forudsætning for faget, at eleven har gennemført faget Kundeservice.

Computerteknologi

v. 1,0

Nummer: 16854-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan ud fra en konkret kravsspecifikation dimensionere en arbejdsstation eller server.
2. Eleven kan installere og administrere en virtualiseringsplatform.
3. Eleven kan tilgå og udføre konfigurationsændringer på virtualiseringsplatforme.
4. Eleven kan konfigurere en arbejdsstation eller server med en hardware RAID controller.

Vejledning:

Ad 1) Ud fra brugerkrav til f.eks. en CAD/CAM arbejdsstation, databaseserver eller

lignende skal eleven kunne sammensætte (købe) hardware, der matcher kravene

Ad 2) Eleverne skal stifte bekendtskab med de servere som benyttes til virtualiseringsplatforme, så de er i stand til at administrere hardwaredelen, herunder fjernsupport som f.eks. ILO fra HP eller lignende

Ad 3) Her tænkes på at eleven skal kunne installere og konfigurere et virtualiseringsoperativsystem f.eks. ESXI eller HyperV på en barebone server

AD 4) Eleven skal have hands on erfaring så de kan konfigurere en RAID Controller, herunder opsætning af forskellige RAID former, genetablere defekte RAID konstruktioner, og evt. konfigurere Hot Standby diske til RAID

Serverteknologi – Databaseserver

v. 1,1

Nummer: 16856-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan, på et niveau svarende til MCP, "Administering a SQL Database Infrastructure", redegøre for arkitekturen for en SQL-Server 2016 eller nyere og herunder SQL-serverens egne datatabeller.
2. Eleven kan planlægge installationen af en SQL server, samt foretage installationen.
3. Eleven kan oprette en database, samt administrere segmenter og andre databaseobjekter, tablespaces og datafiler.
4. Eleven kan oprette brugere, administrere brugerkonti, -rettigheder og -ressourcer, samt bruge SQL-serverens Authentication Modes.
5. Eleven kan administrere tilladelser på kommandoer, Views, Stored Procedure og Funktioner på SQL-serveren.
6. Eleven kan konfigurere og benytte administrationsværktøjer til database-serveren, herunder automatisere administrative jobs og administrere database-serveren både med de grafiske administrationsværktøjer og med SQL-kommandoer.
7. Eleven kan udføre database backup, samt implementere en backup strategi, herunder udføre databaserestore og -recovery.
8. Eleven kan foretage optimering af performance på SQL-serveren bl.a. ved brug af forskellige former for indeksering, Stored Procedure og Statistics, samt brug af Performance Monitoring.
9. Eleven kan konfigurere en forbindelse mellem en client og en databaseserver.
10. Eleven kan flytte data til og fra SQL-serveren ved brug af SQL Server Integration Services (SSIS).
11. Eleven kan redegøre for metoder til at genetablere ødelagte databaser, og kan foretage fejlsøgning på problemer i database systemet, herunder fejlsøgning i SQL-serveren ved hjælp af indbyggede Stored Procedures.

Vejledning:

Undervisningsmaterialet er baseret på MCP-undervisningen: "Administering a SQL Database Infrastructure", arkitekturen for en SQL-Server 2016 eller nyere og herunder SQL-serverens egne datatabeller.

Det anbefales, at man gennemgår emner og stiller opgaver i nævnte rækkefølge:

- a. Installation af SQL Server 2016, hvor eleverne skal tage stilling til alle trin i installations Wizarden. Dog kan visse kritiske valg anbefales. Læg op til eleverne, at de tekniske fejl der opstår undervejs løses ved hjælp af Google og samarbejde. Alle efterfølgende opgaver løses på denne server.
- b. Opret en database med én eller nogle få tilhørende tabeller. Brug eventuelt Import Data wizard. Her stilles opgaver vedrørende Database Properties, herunder segmenter, tablespaces og datafiler. Den database der her oprettes med tilhørende tabeller bør oprettes med data og tabeller der kan anvendes i de næste opgaver.
- c. Gennemgang af Login/User problematikker – herunder Authentication versus Authorization. Logins og Users er ikke med i backup af en database, så her kan PowerShell evt. tages ind og bruges til dokumentation af oprettede Logins og Users. Users oprettes for databasen fra den tidligere opgave.
- d. En gennemgang af de forskellige typer af backup (Full, Differential og Transaction Log). Hvis Backup Wizard bruges kan en række parametre for backup præsenteres. Brug muligheden for at få vist T-SQL statements der genereres i wizarden til at præsentere SQL for backup. Backup af databasen fra de tidligere opgaver.
- e. En gennemgang af Restore og Recover. Hvis Restore Wizard bruges kan en række parametre for backup præsenteres. Brug muligheden for at få vist T-SQL statements der genereres i wizarden til at præsentere SQL for restore.

- f. Der oprettes en MaintenancePlan med anvendelse af wizard. Den baseres på den backup der blev taget tidligere, og kan give anledning til at se på Restore igen.
- g. Gennemgå de forskellige muligheder for optimering og hjælp til overvågning (indeksering, Stored Procedure, Statistics samt Performance Monitoring). Et eksempel på indeksering skal udføres på en tabel med ½-1 million rækker og søgning før og efter oprettelse af indexes.

Målpinde 9 og 11 er ikke behandlet her

Serverteknologi

v. 1,1

Nummer: 16857-2
Varighed: 7½ dage.

Målpinde:

1. Eleven kan installere og konfigurere et serversystem og anvende serversystemets netværksservices og -funktioner, herunder DHCP, DNS og Firewall/Routing.
2. Eleven kan delegere zoner mellem flere DNS-servere.
3. Eleven kan konfigurere og opsætte serveroperativsystemet i forhold til fjernadgang og VPN.
4. Eleven kan konfigurere netværkssikkerheden ved anvendelse af systemets sikkerheds features som f.eks., Public Key Infrastructure (PKI), Internet Protocol Security (IPSec), Radius og lignende.
5. Eleven kan arbejde med serverens storage management-system, herunder eksempelvis disk quota, distribuerede filsystemer.
6. Eleven kan konfigurere og administrere printning, herunder printkø, prioritering mv
7. Eleven kan håndtere brugernes tilladelser i filsystemet.
8. Eleven kan anvende operativsystemets indbyggede værktøj til administration, backup, overvågning, optimering og automatisk opdatering af systemet.
9. Eleven kan foretage fejlsøgning og fejlretning på servere og services på det valgte serversetup.
10. Eleven kan redegøre for centrale begreber vedrørende cloudbaseret serverdrift.

Vejledning:

Dette fag er tænkt som en videreføring af faget Serveradministration og sikkerhed, hvor fokus var på Microsoft teknologier. I dette fag er der mulighed for at inddrage andre teknologier.

- I dette fag kan der opbygges et setup der simulerer flere lokationer til opbygning af remote access, via IPSec eller andet.
- Der kan arbejdes med PKI herunder Certificate Authorities og certifikater
- I den valgte server platform arbejdes der med disksystemet og opbygningen af redundante og dynamiske disksystemer.
- Der arbejdes med administration af disksystem, tilladelser og daglig administration af dette.
- Eleverne introduceres til Cloudteknologier og hvis det er muligt inkluderes en anvendelse af disse.

Der arbejdes med grundlæggende fejlfinding på det opsatte system lokalt og via fjernadgang.

Serveradministration og sikkerhed

v. 1,1

Nummer: 16858-2
Varighed: 10 dage.

Målpinde:

1. Eleven kan installere og konfigurere et client-serversystem.
2. Eleven kan installere og konfigurere Active Directory domain service herunder DHCP og DNS.
3. Eleven kan konfigurere domæner og domæneskove og redegøre for trusts og sites.
4. Eleven kan oprette og administrere bruger- og gruppekonti, samt styre tilladelser til forskellige ressourcer.
5. Eleven kan anvende de forskellige typer af objekter indeholdt i Active Directory, og dermed håndtere tilladelserne til Domænets ressourcer.
6. Eleven kan konfigurere Backup, Recovery og replikering af Active Directory.
7. Eleven kan anvende de indbyggede værktøjer til serveradministration.
8. Eleven kan optimere serversikkerheden på Active Directory og applikationer.
9. Eleven kan anvende lokal sikkerhed på et netværk, herunder omdøbe konti, ændre konto-egenskaber, administration af brugerkonti herunder resette password og slette konti.
10. Eleven kan anvende serveroperativsystemets indbyggede sikkerhedsværktøjer, herunder implementering af lokal password politik samt implementering af lockout politik.
11. Eleven kan optimere serverens netværkssikkerhed ved anvendelse af Windows Firewall.
12. Eleven kan administrere en Member Server, lokale sikkerheds- og gruppepolitikker, herunder implementering, auditering og auditpolitik, sikkerhedslog, krypteret filesystem og anvendelse af hjælpeværktøjer til sikkerhedskonfiguration.
13. Eleven kan administrere og konfigurere Domain Controllerens lokale sikkerhed, gruppe, domain og Site politikker og eleven har forståelse for rangfølgen af implementerede politikker.

Vejledning:

Dette fag bør placeres før Serverteknologi og er baseret på Microsoft-plattformen. Dette fag er sammenfatning og opdatering af de 2 tidligere fag 6222 Operativsystemer samt 6544 Sikkerhed II.

Det kan være en ide at opbygge en case opgave med en mindre virksomhed bestående af eksempelvis: 4 stk. Windows Server 2016 maskiner samt 2 stk. Windows 10 Education eller Pro klienter.

I denne case vil det være fornuftigt at der kan opbygges et redundant system med minimum 2 stk. Domain Controller, DHCP og DNS server
Der indgår oprettelse af organisational units (OU), brugere og GPO'er iht. delopgaverne i den valgte case.

Hvis man vælger at benytte Microsoft Windows Server 2016, kan man introducere de indbyggede sikkerhedsfilosofier fra Microsoft og denne platform, herunder

- Windows Update Services WSUS
- Bitlocker
- Reduced attack surface, herunder Nano og Core versioner
- Assume Breach
- Udvidelser til GPO'er
- Windows defender
- Windows Firewall
- Baseline Security og introduktion til Security Compliance Toolkit (SCT)

Der arbejdes med password- og lockoutpolitikker både lokalt på klienterne og via GPO i Active Directory.

Serverteknologi – Linux

v. 1,1

Nummer: 16859-2
Varighed: 10 dage.

Målpinde:

1. Eleven kan installere, anvende og administrere et Linux system.
2. Eleven kan vælge, installere og afinstallere moduler.
3. Eleven kan forklare, opsætte og gendanne software RAID.
4. Eleven kan opsætte netværksinterface (statisk og dynamisk).
5. Eleven kan, via Shell kommandoer, oprette brugere og grupper med tilhørende tilladelser.
6. Eleven kan servicere og vedligeholde systemet direkte fra Shell.
7. Eleven kan opsætte og anvende Telnet, SSH og FTP klient/server.
8. Eleven kan opsætte og anvende grundlæggende DHCP, Web server samt redegøre for DNS, mailserver inkl. webmail.
9. Eleven kan redegøre for Pre- og Post-routing, og med vejledning opsætte NAT på en Linux-server.
10. Eleven kan opsætte en simpel firewall via terminal.
11. Eleven kan anvende Shell til at udarbejde Shell Scripts.
12. Eleven kan opsætte og konfigurere et grafisk brugerinterface.
13. Eleven kan fejlfinde på et Linux system.
14. Eleven kan dokumentere udført arbejde i forbindelse med installation og fejlfinding.

Vejledning:

Faget kan ligge på H2 forinden Web-server og gerne efter Windows server kurset. Casen til Ekspertniveau er fin, men på rutineniveau skal der foregå undervisning.

Den indledende opsætningen af serveren kan med fordel indeholde opsætning og gendannelse af RAID og opsætning af statisk IP (målpind 4+5).

På mail -serveren oprettes et antal brugere.

Oprettelse af Shell Scripts kan kombineres med oprettelse af brugere eller opsætning af regler i firewall (f.eks. firewall.sh ssh, åbner for ssh i firewallen).

Linux klienter opsættes med dynamisk IP (målpind 5).

Efter opsætning og afprøvning af ovenstående netværk, udskiftes Firewall / router med en grafisk baseret løsning f.eks. PFSense.

Der kan f.eks. suppleres med:

- nmap
- tcpdump
- journalctl.
- SSH logins med kryperings nøgler i stedet for passwords.

Det er vigtigt at eleverne introduceres til Linux og at der er tid og mulighed for at sætte sig ind i et for eleverne helt nyt styrestem.

Her kan man evt. benytte det "vendor-neutrale" kursus Linux Essentials som forlæg for grundlæggende Linux-undervisning.

Til elever som ikke magter ekspert-casen, men mangler udfordring kan man f.eks. bede eleven lave sin egen Linux distribution vha. et open source materiale som hedder "Linux from Scratch".

De svageste elever får en god støtte fra Linux Essentials læringsportalen og de elever som skal udfordres får rig mulighed for at grave sig ned i et nyt styresystem via "Linux from Scratch" og blive gjort klar til mere avancerede emner som Network Service

Linux Essentials materialet er frit tilgængeligt for alle skoler som benytter Cisco Netacad, der ikke er udviklet eller supporteret af Cisco. Det er Linux Professional Institute (lpi.org), en non-profit open source organisation, der også tilbyder international anerkendte certificeringer) som står bag indholdet.

Serverteknologi – Web-server

v. 1,1

Nummer: 16860-2
Varighed: 10 dage.

Målpinde:

1. Eleven kan installere og konfigurere en Web-server og tilhørende plugins i forhold til en given opgave.
2. Eleven kan konfigurere sikkerhed og brugerspecifikke tilladelser.
3. Eleven kan installere supplerende plugins på en Web-server, som f.eks. CGI og Java Servlets.
4. Eleven kan opsætte HTTP og HTTPS.
5. Eleven kan installere og konfigurere værktøjer til administration af hostede løsninger svarende til et web host panel suite, som bl.a. indeholder SQL implementering.
6. Eleven kan opsætte og anvende DNS, Mail server, DHCP, Web-server samt webmail.
7. Eleven kan anvende Pre- og Post-routing, samt opsætte NAT.
8. Eleven kan opsætte Firewalls via terminal.
9. Eleven kan forklare og opsætte traditionelle og transparente Proxy som f.eks. Squid, samt opsætte Firewalls, som f.eks. Iptables, FirewallD eller PFSense, sammen med Squid.

Vejledning:

Faget kan ligge på H2 efter Serverteknologi I – Linux.

Det kan anbefales at bruge en case med klient og server opsætning, hvor serverne er med minimal installation uden GUI. I et virtuelt miljø med en Firewall, bestående af DMZ, LAN og WAN zoner.

DMZ: kan indeholde Linux server med følgende services.:

- DNS.
- FTP-server
- Mail server med tilhørende webmail.
- Web-server med bruger adgangsbegrænsning og tilhørende webmail.
- Telnet og SSH.

LAN: skal indeholde en DHCP-server og mindst én Linux klient, derudover kan der suppleres med yderligere Linux eller Windows- klienter.

Firewall / router: denne opbygges på en Linux distribution, og indeholder følgende services:

- Pre- og Post routing og NAT
- Firewall med Iptables eller FirewallD
- Squid

Eleven lærer at installere og konfigurer en samlet "Web Host Panel Suite" løsning. Eleven lærer at forenkle processen med at installere og konfigurer samt integrere forskellige programmer og scripts, der er nødvendige for at levere web hosting-tjenester.

Eleven lærer at administrere og kontrollere servere og kunde administration via en web-baseret GUI-grænseflade.

Eleven lærer at installere tilgængelige moduler til kontrolpanelet som f.eks:

- Webserver, herunder både HTTP og HTTPS
- DNS-server
- SMTP-server
- POP3/IMAP server
- FTP-server
- SQL-server
- Spamfilter
- Firewall
- Antivirus

Eleven lærer derudover at arbejde med serverlogfiler, diskplads samt foretage besøgsstatistik ved hjælp af web loganalyse programmer.

Web Host Panel Suite kan f.eks. være ISPconfig, Plesk eller cPanel.

Netværk II

v. 1,0

Nummer: 16861-2
Varighed: 10 dage.

Målpinde:

1. Eleven kan selvstændigt eller i samarbejde med andre elever løse en arbejdsopgave omhandlende en switchet LAN-løsning.
2. Eleven kan lave et grundlæggende netværksdesign baseret på en hierarkisk model.
3. Eleven kan konfigurere og fejlfinde hhv. routere og L2/L3 switche.
4. Eleven kan konfigurere og idriftsætte dynamisk routing med Distance Vector og Link State protokoller i IPv4 og/eller IPv6, f.eks. OSPF, EIGRP og RIP.
5. Eleven kan løse almindelige problemer i forbindelse med routing.
6. Eleven kan implementere redundans i et switched/routed netværk.
7. Eleven kan beskrive og forklare funktionen af protokoller som OSPF, EIGRP og STP.
8. Eleven kan lave en basal implementering af First Hop Redundancy (FHRP), HSRP, VRRP og GLBP.

Vejledning:

Niveau og indhold svarer til CCNA R&S Scaling Networks v6 eller nyere.

Med udgangspunkt i Netacad's CCNA- Scaling Networks Scaling Networks v6.0 kursus:

Teoribog CCNA R&S Scaling Networks v6 Companion Guide."

Planen for gennemlæsning af bogen se dagsplanen.

Det kan forventes at du hver dag har hjemmeopgaver der skal løses.

Læseskema Netværk II (CCNA R&S ScaN)

De angivne øvelser er minimum – Men mangler de ekstra opgaver så hav nogle ekstra i baghånden.

Arbejdsbeskrivelse:

Råd til eleven:

1. 1. **Forud** for emne – dvs. *inden du møder* til undervisning - Se eventuelle youtube videoer som handler om det emne du skal have næste dag.
www.youtube.com/NetworkMadeEasy
2. Lav projektopgave/fejlfinding på fysisk udstyr (gruppearbejde GNS3/PT kan evt. bruges når du ikke er i klassen. Se evt. videoer der kan hjælpe dig med hvordan GNS3 installeres.
3. Læs de angivne sider i bogen.
4. **Tag kapiteltests første gang** (gruppearbejde) undlad at svare på spørgsmål du intet ved om

Se hvilke emner du svarede forkert på i kapiteltesterne og læs (max 30 min.) Tag kapiteltest igen (hjemmearbejde - kan undlades hvis 100% første gang)

5. Gennemgår de spørgsmål i kapiteltesten der volder problemer. Hvis der er spørgsmål/emner i kapiteltesten der er svære at forstå så kig nærmere på problemstillingen sammen.

Dag	Emne	Øvelse m.m.
1	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 – 13.15 LAN Design + Scaling VLANs	Lab 2.1.4.5 - Configure Extended VLANs, VTP, and DTP Lab 2.2.2.5 - Troubleshooting Inter-VLAN Routing
2	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 08.00 – 15.00 "STP" Opsamling ch 1+2 test	Lab 3.1.2.12 - Building a Switched Network with Redundant Links Lab 3.3.2.3 - Configuring Rapid PVST+, PortFast, and BPDU Guard

Dag	Emne	Øvelse m.m.
3	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 08.00 – 15.00 "Etherchannel and HSRP" Opsamling ch 3 test	Lab 4.2.1.4 - Configuring EtherChannel Lab 4.2.2.4 - Troubleshooting EtherChannel Lab 4.3.3.4 - Configure HSRP
4	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 09.45 – 15:00 "Dynamic Routing + EIGRP" Opsamling ch 4 test	Lab 6.2.2.5 - Configuring Basic EIGRP for IPv4 Lab 6.4.3.5 - Configuring Basic EIGRP for IPv6
5	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 – 12.30 EIGRP Tuning and Troubleshooting Opsamling ch 5+6 test	Lab 7.1.3.6 - Configuring Advanced EIGRP for IPv4 Features Lab 7.2.3.6 - Troubleshooting Basic EIGRP for IPv4 and IPv6 Lab 7.2.3.7 - Troubleshooting Advanced EIGRP
	kapiteltestene åbnes hele weekenden	
6	8.00 – 13.15 Single-Area OSPF + Multiarea OSPF Opsamling ch 7 test	Lab 8.2.4.5 - Configuring Basic Single-Area OSPFv2 Lab 8.3.3.6 - Configuring Basic Single-Area OSPFv3 Lab 9.2.2.8 - Configuring Multi-area OSPFv2
7	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 – 15.00 + OSPF Tuning and Troubleshooting Opsamling ch 8+9 test	Lab 9.2.2.8 - Configuring Multi-area OSPFv2 Lab 9.2.2.9 - Configuring Multi-area OSPFv3 Lab 10.1.1.13 - Configuring OSPFv2 on a Multiaccess Network Lab 10.1.3.5 - Configuring OSPFv2 Advanced Features Lab 10.2.3.3 - Troubleshooting Basic Single-Area OSPFv2 and OSPFv3 Lab 10.2.3.4 - Troubleshooting Advanced Single-Area OSPFv2 Lab 10.2.4.5 - Troubleshooting Multiarea OSPFv2 and OSPFv3
8	8.00 – 15.00 Opsamling ch 10 test	Case Study eller challenge opgave
9	8.00 – 15.00 Praktisk prøve	
10	8.00 – 13.15 Teoretisk prøve, oprydning, feedback og karaktergivning	

Serverautomatisering I

v. 1,0

Nummer: 16862-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan anvende PowerShell til automatisering og fjernadministration af servere og klienter.
2. Eleven kan implementere sikkerheden korrekt i forbindelse med scripting i Powershell.
3. Eleven kan anvende de grundlæggende Cmdlets og forstår at bruge de indbyggede hjælpefunktioner i Powershell.
4. Eleven kan anvende pipelinen i Powershell.
5. Eleven kan anvende grundlæggende systemkald til WBEM (Web-Based Enterprise Management) funktioner.
6. Eleven kan anvende -whatif, -confirm og -transcript kommandoerne i Powershell.
7. Eleven kan anvende Aliases i Powershell.
8. Eleven kan oprette og bruge variabler i Powershell.
9. Eleven kan anvende datahåndtering op imod en database struktur.

Vejledning:

WBEM er den overordnede betegnelse for teknologier som WMI eller CIM.
Database struktur kan være SQL, Active Directory mm.

Det anbefales, at dette fag er placeret efter fagene Computerteknologi (6225) og Serverteknologi (Ny), samt at eleverne er blevet præsenteret for en WBEM teknologi, samt et af produkterne SQL og Active Directory inden dette fag.
Derudover anbefales det, at faget ligger tidligt i første hovedforløb, dog efter Computerteknologi og Serverteknologi, og at der i fag der kommer efter i videst mulig omfang tænkes scriptprogrammering ind i fagene.

Det anbefales, at man gennemgår emner og stiller opgaver i nævnte rækkefølge:

- a. Start med at introducere de tre kommandoer Get-Command, Get-Help, og Get-Member og brug dem til at forklare noget om opbygningen af kommandoer i PowerShell. (Noun/Verb/Parameters).
- b. Introducer eleverne for programmeringsbegreberne objekter, variable, selektion, iteration og funktioner som elementer i at skrive script filer.
Her kan stilles nogle helt simple opgaver, hvor der skal bruges variable, selektion (if/switch) og iterationer (while,do,for) sammen med PowerShell kommandoer (cmdlet's). Her kan opgaverne stilles så eleverne selv skal finde de udsagnsord og navneord i opgaveteksten, der gør at de med hjælpefunktionerne selv kan finde de cmdlet's de skal bruge, samt de parametre de skal bruge.
Bemærk, at objekter, selektion, iteration og funktioner ikke er nævnt under målpindene. Det anbefales at begreberne introduceres, da de giver en dybere forståelse af arbejdet med at udvikle scrips.
- c. Træk nogle elementer ind fra fagene Computerteknologi (6225) og Serverteknologi (Ny) og giv eleverne en opgave, der udnytter disse elementer til automatisering og fjernadministration af eksempelvis servere.
Her kan de eventuelt bygge videre på en struktur (menu?) fra den foregående opgave, når de skal opbygge en programstruktur til automatisering.
- d. I forbindelse med den sidste opgave kan elementet sikkerhed tages op. I den forbindelse kan der diskuteres rettigheder for den bruger der kører scripts, og scripts der køres uden en aktiv bruger. Er der tilstrækkelige rettigheder til at kunne udføre fjernadministration?
Derudover bør sikkerhed og risiko i forbindelse med eksekvering af scripts andre har skrevet diskuteres.
- e. Anvendelse af begrebet alias i PowerShell præsenteres og der kan gives opgaver med flere typer af alias, simple alias, alias med brug af funktioner og alias med parametre.
Her kan diskuteres det sikkerheds problem, der består i at alias'er ikke altid er tilgængelige når et script køres.

- f. Hvis der er anvendt en WBEM teknologi tidligere i forløbet bør opgaverne omkring grundlæggende system kald knyttes til den pågældende teknologi. Dette punkt og datahåndtering kan med fordel ligge sidst i undervisningen i faget, og dermed bruges til at opøve en elementær rutine i at finde kommandoer og i at strukturere et script program.

Datahåndtering bør relateres til det produkt (eksempelvis AD eller SQL) eleverne er blevet introduceret til tidligere i forløbet. Her vil det være en fordel, hvis eleverne har adgang til produktet samtidig, og derfor har en mulighed for at

Dette punkt og WBEM teknologi kan med fordel ligge sidst i undervisningen i faget, og dermed bruges til at opøve en elementær rutine i at finde kommandoer og i at strukturere et script program.

Praktisk fejlfinding

v. 1,0

Nummer: 16863-3
Varighed: 7½ dage.

Målpinde:

1. Eleven kan udføre struktureret fejlfinding ud fra en konkret fejlbeskrivelse.
2. Eleven kan anvende OSI Modellen i forbindelse med fejlfinding og identificere relevante metoder (Top-Down, Bottom-Up og Divide-and-Conquer).
3. Eleven kan isolere et problem til enhed, protokol eller service.
4. Eleven kan anvende fejlfindingskommandoer op mod en kommandobaseret grænseflade (CMD, CLI).
5. Eleven kan dokumentere sin fejlfindingsproces, udføre fejlretning og forberede eskalering.
6. Eleven kan fejlfinde i et enterprise-miljø.

Vejledning:

Faget bør placeres på H3 før svendepøveprojektet, og indeholder fejlfinding i et enterprise-miljø.

Det vil være godt at have en standardopstilling som eleverne kan øve sig på enten med fysisk udstyr eller i et virtuelt miljø.

Der trænes i at udvælge og anvende korrekt fejlfindingsmetode og udarbejde fejlfindingsplan ud fra en konkret fejlbeskrivelse.

Vægten ligger på praktisk struktureret fejlfinding i et stort firmas infrastruktur. Dvs. både netværks-, server- og klientfejl kan forekomme.

Kundeservice

v. 1,0

Nummer: 16864-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan redegøre for vigtigheden af kundeservice.
2. Eleven kan redegøre for indholdet af begreber vedrørende kundeservice.
3. Eleven kan redegøre for og anvende elementer fra kommunikation og konflikthåndtering.
4. Eleven kan anvende IT-værktøjer og metoder til planlægning, prioritering og dokumentation af servicearbejde.
5. Eleven kan redegøre for hvilke forudsætninger, der skal til for at levere kvalificeret kundeservice.
6. Eleven kan sikre kvaliteten af leveret kundeservice samt foretage evaluering heraf.
7. Eleven kan anvende systematiske spørgeteknikker til behovsafdækning.
8. Eleven kan instruere og vejlede interne og eksterne kunder i IT-services.
9. Eleven kan på en struktureret måde foretage fejlfinding.

Vejledning:

Faget bør placeres på H1. Eleverne skal forstå hvad begrebet Kundeservice dækker og hvilke forudsætninger - eks. viden, råderum, erfaring, "værktøjer", der skal til for at levere god kundeservice. Eleverne skal undervises i værktøjer inden for planlægning og prioritering af opgaver eks. Gantt-kort og ABCD-model. Eleverne skal undervises i kommunikationsmodeller som eks. Shannon og Weaver og lære om konflikttrappen til afspænding af konflikter.

Eleverne skal undervises i kvalitetssikring, spørgeteknik og struktureret fejlfinding, samt planlægge og gennemføre en kort instruktion i anvendelsen af et IT-værktøj. Forløbet kan bygges op om at eleverne skal udarbejde en manual til nye medarbejdere om hvordan man yder kundeservice. Forløbet bør trække på elevernes egne erfaringer med kundeservice for at få så mange erfaringer som muligt i spil i undervisningen.

Grundlæggende databaseprogrammering

v. 1,0

Nummer: 16865-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan anvende SELECT til at foretage forespørgsler op imod en tabel, herunder sortering, gruppering, filtrering, aggregering og funktioner.
2. Eleven kan anvende INSERT, UPDATE, DELETE til at foretage manipulationer af en tabel.
3. Eleven kan oprette, redigere og slette tabeller samt dokumentere vha. E/R-diagram.
4. Eleven kan anvende relationstyperne 1:mange, mange:mange og 1:1 med tilhørende nøgler.
5. Eleven kan implementere konsistenskrav, referenceintegritet, relationer og forskellige constraints.
6. Eleven kan anvende JOIN og SUBQUERIES til at foretage forespørgsler fra flere tabeller.
7. Eleven kan oprette og slette en database.
8. Eleven kan oprette Views og begrunde sikkerhedsaspektet.

Vejledning:

Omhandler SQL-syntaksen og forståelsen af opbygningen af en database, herunder dokumentation.

Gennemføres på færdig database med ca. 5 tabeller. Overalt benyttes SQL-scripts
CASE: Eleven designer, dokumenterer og implementere en mindre database med minimum 5 tabeller (f.eks. et bibliotek) (ingen GUI)

Som database kan benyttes mySql, MS SQL Server eller lignende.

Netværk III

v. 1,0

Nummer: 16866-2
Varighed: 10 dage.

Målpinde:

1. Eleven kan selvstændigt eller i samarbejde med andre elever løse en arbejdsopgave omhandlende en WAN-løsning.
2. Eleven kan konfigurere og fejlfinde struktureret på enheder i et enterprise-netværk.
3. Eleven kan løse almindelige problemer med Data Link protokoller.
4. Eleven kan implementere Remote Access opkoblinger og vælge bredbåndsteknologier passende til firmabrug.
5. Eleven kan implementere en simpel EGP (Exterior Gateway Protokol).
6. Eleven kan implementere en almindelig tunnel mellem 2 netværksenheder.
7. Eleven kan udvælge og etablere passende sikkerhedsforanstaltninger i forbindelse med WAN løsninger.
8. Eleven kan på baggrund af en præcis kravspecifikation udvikle og implementere avancerede Access-lister i IPv4 og/eller IPv6 miljøer.
9. Eleven kan beskrive QoS, IoT, Cloud Computing og virtualisering.

Vejledning:

Niveau og indhold svarer til CCNA R&S Connecting Networks v6 eller nyere.

Med udgangspunkt i Netacads CCNA- R&S Connecting Networks v6.0 kursus:

Teoribog CCNA R&S Connecting Networks v6 Companion Guide.”

Planen for gennemlæsning af bogen se dagsplanen.

Det kan forventes at du hver dag har hjemmeopgaver der skal løses.

Læseskema Netværk III (CCNA R&S CN)

De angivne øvelser er minimum – Men mangler de ekstra opgaver så hav nogle ekstra i baghånden.

Arbejdsbeskrivelse:

Råd til eleven:

1. **Forud** for emne – dvs. *inden du møder* til undervisning - Se eventuelle youtube videoer som handler om det emne du skal have næste dag. www.youtube.com/NetworkMadeEasy
2. Lav projektopgave/fejlfinding på fysisk udstyr (gruppearbejde GNS3/PT kan evt. bruges når du ikke er i klassen. Se evt. videoer der kan hjælpe dig med hvordan GNS3 installeres.
3. Læs de angivne sider i bogen.
4. **Tag kapiteltests første gang** (gruppearbejde) undlad at svare på spørgsmål du intet ved om

Se hvilke emner du svarede forkert på i kapiteltesterne og læs (max 30 min.) Tag kapiteltest igen (hjemmearbejde - kan undlades hvis 100% første gang)

5. Gennemgår de spørgsmål i kapiteltesten der volder problemer. Hvis der er spørgsmål/emner i kapiteltesten der er svære at forstå så kig nærmere på problemstillingen sammen.

Dag	Emne	Øvelse m.m.
1	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 – 13.15 WAN Concepts + Point-to-Point Connections	
2	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 08.00 – 15.00 Branch Connections Opsamling ch 1+2 test	Lab 2.3.2.7 - Configuring Basic PPP with Authentication Lab 2.4.1.5 - Troubleshooting Basic PPP with Authentication

Dag	Emne	Øvelse m.m.
3	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 08.00 – 15.00 Opsamling ch 3 test	Lab 3.2.2.7 - Configuring a Router as a PPPoE Client for DSL Connectivity Lab 3.2.2.8 - Troubleshoot PPPoE Lab 3.4.2.6 - Configuring a Point-to-Point GRE VPN Tunnel 3.5.3.5 Lab - Configure and Verify eBGP Lab 3.6.1.3 - Configure a Branch Connection.docx
4	09.45 – 15:00 14.00 Opsamling ch 4 test	
5	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 – 12.30 Access Control Lists	Lab 4.2.2.13 - Configuring and Verifying Extended ACLs Lab 4.3.2.7 - Configuring and Verifying IPv6 ACLs Lab 4.4.2.11 - Troubleshooting ACL Configuration and Placement
	Kapiteltestene åbnes hele weekenden	
6	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 – 13.15 Network Security and Monitoring Opsamling ch 5 test	Lab 5.2.2.6 - Configuring SNMP Lab 5.3.2.3 - Implement Local SPAN Lab 5.3.3.2 - Troubleshoot LAN Traffic Using SPAN
7	Se evt. videoer på NetworkMadeEasy inden du møder omhandlende dagens emner 8.00 – 15.00 Quality of Service + Network Evolution Opsamling ch 6 test	Lab 8.0.1.2 Network Breakdown Instructions Lab 8.2.1.5 - Configure IP SLA ICMP Echo Lab 8.3.1.1 Documentation Development Instructions
8	8.00 – 15.00 Opsamling ch 7+8 test	Case Study eller challenge opgave
9	8.00 – 15.00 Praktisk prøve	
10	8.00 – 13.15 Teoretisk prøve, oprydning, feedback og karaktergivning	

Netværkssikkerhed

v. 1,0

Nummer: 16867-3
Varighed: 10 dage.

Målpinde:

1. Eleven kan redegøre for typiske sårbarheder, der er i sikkerheden på et netværk.
2. Eleven kan konfigurere Lag 2 og 3 enheder inden for et netværksområde, hvor der eksempelvis anvendes Switch sikkerhedsfaciliteter og IOS.
3. Eleven kan konfigurere logging f.eks. Syslog, SNMP client på switche og routere samt implementere NTP i et LAN.
4. Eleven kan konfigurere en logging Syslog server.
5. Eleven kan konfigurere en hardware Firewall til at udføre grundlæggende sikkerhedsoperationer på et netværk.
6. Eleven kan sikre et netværk ved hjælp af Network-Based Intrusion Prevention System.
7. Eleven kan opbygge et IPSec VPN netværk, både site to site og client based.
8. Eleven kan implementere Authentication, Authorization og Accounting løsninger til at sikre adgangen til netværket.
9. Eleven kan beskrive hvilke offentlige sikkerhedsregler og gældende standarder for it-sikkerhed, som f.eks. ISO 27001, der skal tages hensyn til i forbindelse med løsning af it-opgaver.

Vejledning:

Kurset læner sig op ad CCNA Security kurset i Netacad.

Gennem praktiske øvelser og opgaver arbejde med netværks-sikkerhed i et Cisco baseret netværk. Der bygges videre på elevernes viden fra de foregående kurser samt deres praktikperiode.

Efter at have arbejdet videre med emnerne på kurset kan de også tage en CCNA Security certificering, ved at tage en test på et testcenter. Denne certificering er ikke inkluderet i kurset.

Det kommer til dreje sig om at installere, konfigurere og servicere et grundlæggende sikkerheds niveau på LAN/WAN netværk. Eleverne forudsættes at have praktisk og teoretisk kendskab til routere og computernetværk svarende til indholdet af Netværkskurserne på tidligere skoleforløb.

Overordnede emner:

- Konfigurering af Lag 2 og 3 enheder inden for et netværksområde, hvor der eksempelvis anvendes Cisco Switch sikkerhedsfaciliteter og Cisco IOS.
- Konfigurering af en IOS Firewall til at udføre grundlæggende sikkerhedsoperationer på et netværk.
- Sikring af et netværk ved hjælp af Network-Based Intrusion Prevention System.
- Opbygge IPSec baserede VPN forbindelser.
- Implementering af Authentication, Authorization og Accounting løsninger til at sikre adgangen til netværket.

Undervisningsmaterialer: Der kan f.eks. bruges undervisningsmaterialerne fra Netacad samt instruktørens eget materiale. Eleven kan indrulleres i en klasse i Cisco Network Academy hvor der er adgang til al læsematerialet og de praktiske øvelser. Underviserens kan lægge udkast til hvilke noter der er relevante for kurset. Noterne bør kun indeholde kommandosyntakser. De kan være udgangspunkt hvis eleven føler behov for at lave sine egne noter til kurset.

Forløbsbeskrivelse: Der vil være vekselvirkning mellem teori og praktik med fokus på de praktiske færdigheder. Kurset består af ca. 50 % teori og 50 % praktisk arbejde.

Forventninger:

- Det forventes at eleven følger planen for kapitel tests og har lavet alle tests med over 80 procent inden de afsluttende prøver.
- Det forventes at eleven gør sit bedste for at følge planen for praktiske øvelser
- Det forventes at eleven møder rettidigt og ædru til undervisningen

- For at bestå kurset forventes det at eleven deltager aktivt i undervisningen - hver dag! Derudover kan der være hjemmearbejde.

Dagsplan:

Grønne felter = Teoretisk gennemgang

Labs i parentes (optional)

UGE 3					UGE 4				
Mandag	Tirsdag	Onsdag	Torsdag	Fredag	Mandag	Tirsdag	Onsdag	Torsdag	Fredag
Sik. III	Sik. III	Sik. III		Sik. III	Sik. III	Sik. III	Sik. III		Sik. III
Sik. III	Sik. III	Sik. III		Sik. III	Sik. III	Sik. III	Sik. III		Sik. III
Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III
Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III
Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III	Sik. III
Sik. III	Sik. III	Sik. III	Sik. III		Sik. III	Sik. III	Sik. III	Sik. III	
	Sik. III	Sik. III	Sik. III			Sik. III	Sik. III	Sik. III	
	Sik. III	Sik. III	Sik. III			Sik. III	Sik. III	Sik. III	
Kap 1		Kap 4		Kap 8	Kap 9		Kap 11	SBFA 1	
	Kap 2		Kap 5						PT SBFA
			(Kap 7)		Kap 10	Ch.10 Lab A		SBFA 2	
	Kap 3		Kap 6				Ch.11 Lab		Oline-final
	Ch.2 Lab		Ch.5 Lab		Ch.8 Lab	Ch.10 Lab B		SBFA 3	
(Ch.1 Lab A)		Ch.4 Lab		(Ch.7 Lab)	Ch.9 Lab	Ch.10 Lab C			
	Ch.3 Lab		Ch.6 Lab						
(Ch.1 Lab B)						Ch.10 Lab D			

Serverteknologi – Cluster

v. 1,0

Nummer: 16868-3
Varighed: 5 dage.

Målpinde:

1. Eleven kan, ud fra en case-opgave, selvstændigt designe, planlægge, installere, konfigurere og dokumentere en samlet clusterløsning, der omhandler de beskrevne mål, og kan herigennem demonstrere viden, færdigheder og kompetencer, der ligger ud over de i faget beskrevne mål og målniveauer.
2. Eleven kan planlægge, installere og vedligeholde clusters til forskellige formål.
3. Eleven kan vælge den clusterløsning der bedst opfylder case-opgavens kravspecifikation.
4. Eleven kan opsætte og konfigurere et cluster, på både Windows og Linux.
5. Eleven kan oprette web-applikations clusters.
6. Eleven kan administrere clusters, foretage oprettelse af cluster ressourcer samt foretage cluster troubleshooting.
7. Eleven kan installere Network Load Balancing cluster samt varetage troubleshooting.

Vejledning:

En kombination af teori og praksis, den praktiske opsætning kan foregå på en type 2 hypervisor.

F.eks. på VMware Workstation kan installeres en type 1 hypervisor som f.eks. ESXi, Xen, Hyper-V.

Clusterteknologien der anvendes er HA og NLB, både på en Windows Platform og en Linux platform. Der bør implementeres Clustering for almindelige serveropgaver, f.eks. filserver, printserver, DHCP, DNS, SQL, Mail, Firewall og lignende.

Virtualisering

v. 1,0

Nummer: 16869-2
Varighed: 7½ dage.

Målpinde:

1. Eleven kan redegøre for fordele og ulemper ved implementeringen af et Virtuelt Infrastructure.
2. Eleven kan installere og konfigurere en Hypervisor, som f.eks. VMware ESXi eller Microsoft Hyper-V.
3. Eleven kan installere og konfigurere et centralt styringsværktøj til administration af en eller flere Hypervisore, som f.eks. VMware vCenter Server eller Hyper-V Manager.
4. Eleven kan opsætte et virtuelt miljø med redundante netværksforbindelser til eksempelvis Storage, Management og Virtuelle maskiner.
5. Eleven kan konfigurere og implementere Network Storage, som eksempelvis SAN, NAS eller iSCSI, i et givent virtuelt miljø.
6. Eleven kan installere og konfigurere et Virtuelt Cluster.
7. Eleven kan oprette og deploye virtuelle maskiner, manuelt eller fra templates, herunder konvertere fysiske installationer til virtuelle (p2v Consolidation).
8. Eleven kan modificere, administrere og migrere virtuelle maskiner.
9. Eleven kan konfigurere og administrere brugerroller og -rettigheder i et givent virtuelt datacenter.
10. Eleven kan administrere og fordele hardwareressourcer over flere logiske maskiner.
11. Eleven kan redegøre for og implementere live migration på et givent Virtuelt datacenter.
12. Eleven kan administrere og implementere automatisk ressourcestyring i et givent Virtuelt Cluster.
13. Eleven kan redegøre for og implementere High Availability og eventuelt Fault tolerance i et givent Virtuelt Cluster.
14. Eleven kan redegøre for fordele og ulemper ved implementeringen af et Virtuelt Infrastructure.
15. Eleven kan konfigurere en Samba fil og print server i et multibrugermiljø bestående af Microsoft Windows operativsystemer og applikationer kombineret med en UNIX/Linux baseret server.
16. Eleven kan undersøge innovative løsninger inden for virtualiseringsløsninger.

Vejledning:

Eleverne skal i løbet af kurset installere, konfigurere og administrere en komplet virtualiseringsplatform, som den ville se ud i et virkeligt datacenter.

Målepinde 1-15 svarer meget godt til VMware vSphere: Install, Configure, Manage, men noget tilsvarende kan sagtens klares med Hyper-V også.

Af innovative løsninger kunne på nuværende tidspunkt nævnes vSAN fra VMware, Container computing og lignende.

Serverautomatisering II

v. 1,0

Nummer: 16870-3
Varighed: 5 dage.

Målpinde:

1. Eleven kan anvende scriptsprog på tværs af platforme.
2. Eleven kan udvikle egne funktioner i et scriptsprog.
3. Eleven kan dokumentere egne funktioner i et scriptsprog.
4. Eleven kan implementere indbygget brugerhjælp med et scriptsprog.
5. Eleven kan anvende indbyggede programstrukturer i et scriptsprog.
6. Eleven kan anvende biblioteker/moduler i et scriptsprog.

Vejledning:

Tværs-platform scriptsprog kan for eksempel være Pearl, Python, PowerShell, Ruby, TCL, LUA eller andre.

Det anbefales at faget ligger på tredje hovedforløb eller tidligt i fjerde hovedforløb, og at der i fag, der kommer efter dette fag i videst mulig omfang tænkes scriptprogrammering ind i fagene.

Eleverne forventes at have haft faget 'Serverautomatisering I' før dette fag, og de dermed er bekendt med PowerShell scriptsproget.

Det anbefales, at man gennemgår emner og stiller opgaver i nævnte rækkefølge:

- a. Start med introduktion til det valgte scriptsprogs terminolog og grundlæggende begreber. Det kan eventuelt gøres ved at målpindene fra Serverautomatisering I repeteres i det valgte scriptsprog. Den resterende vejledning er med udgangspunkt i PowerShell, og de må derfor transformeres til det valgte scriptsprog.
- b. Introducér eleverne for programmeringsbegreberne objekter, variable, selektion, iteration og funktioner som elementer i at skrive script filer. Her kan stilles nogle passende komplicerede opgaver, hvor der skal bruges variable, selektion (if/switch) og iterationer (while,do,for) sammen med det valgte scriptsprogs kommandoer. Opgaverne bør være forskelligartede og så de tvinger eleverne til at vælge forskellige programstrukturer.
- c. Introducér eleverne for 'Advanced function' begrebet i PowerShell, og her kan så stilles en række relevante opgaver. Der bør vælges en område opgaverne kan relateres til. Det kan eksempelvis være Active Directory eller SQL Server, og gerne et emne der i øvrigt undervises i på samme forløb.
- d. Eleverne introduceres til Get-Help konceptet i PowerShell og der undervises så i, hvordan man indbygger hjælp i de allerede udviklede functions fra punktet herover.

Som afslutning på forløbet PowerShell Modules og den functions fra de to sidste punkter gemmes i et Module og herunder diskuteres de muligheder der er i anvendeligheden af modules.

Netværksteknologi I

v. 1,0

Nummer: 16871-3
Varighed: 12½ dage.

Målpinde:

1. Eleven kan beskrive et entrepris netværk med en enkelt eller flere internetopkoblinger.
2. Eleven kan redegøre for forskelle mellem routingprotokoller.
3. Eleven kan implementere RIPv2 og RIPng.
4. Eleven kan konfigurere, implementere og optimere EIGRP for både IPv4 og IPv6 herunder brugen af address familie.
5. Eleven kan konfigurere, implementere og optimere OSPF for både IPv4 og IPv6 herunder brugen af address familie.
6. Eleven kan redegøre for optimering af performance ud fra router opdateres.
7. Eleven kan implementere path control - Fast Switching (f.eks. CEF), policy-based routing (PBR) og SLA.
8. Eleven skal kunne etablere redistribution mellem de forskellige routing protokoller.
9. Eleven kan redegøre for BGP terminologi, koncept, drift, konfiguration, afprøvning og fejlfinding.
10. Eleven kan sikre managementdelen af routere vha. authentication og andre best practice foranstaltninger.

Vejledning:

Niveau og indhold svarer til CCNP R&S Implementing Cisco IP Routing v7 eller nyere.

Med udgangspunkt i Netacads CCNP-ROUTE v7.1 kursus:

Teoribog Implementing Cisco IP Routing (ROUTE) Foundation Learning Guide: (CCNP ROUTE 300-101).

Planen for gennemlæsning af bogen se dagsplanen.

Det kan forventes at du hver dag har hjemmeopgaver der skal løses.

Læseskema Netværksteknologi I (CCNP-ROUTE)

De angivne øvelser er minimum – Men mangler de ekstra opgaver så hav nogle ekstra i baghånden.

Arbejdsbeskrivelse:

Råd til eleven:

1. **Forud** for emne – dvs. *inden du møder* til undervisning - Se eventuelle youtube videoer som handler om det emne du skal have næste dag. www.youtube.com/NetworkMadeEasy
 2. Lav projektopgave/fejlfinding på fysisk udstyr (gruppearbejde GNS3/VIRL kan evt. bruges når du ikke er i klassen. Se evt. videoer der kan hjælpe dig med hvordan GNS3/VIRL installeres.
 3. Læs de angivne sider i bogen.
 4. **Tag kapiteltests første gang** (gruppearbejde) undlad at svare på spørgsmål du intet ved om
- Se hvilke emner du svarede forkert på i kapiteltesterne og læs (max 30 min.) Tag kapiteltest igen (hjemmearbejde - kan undlades hvis 100% første gang)
5. Gennemgår de spørgsmål i kapiteltesten der volder problemer. Hvis der er spørgsmål/emner i kapiteltesten der er svære at forstå så kig nærmere på problemstillingen sammen.

Dag	Emne	Øvelse m.m.	Kapiteltests
1	Se videoer (IPv6) på NetworkMadeEasy inden du møder 08.00 – 15.00 Routing Concepts, EIGRP	Lab 1-1, Basic RIPng and Default Gateway Configuration Lab 2-1, EIGRP Load Balancing Lab 2-2, EIGRP Stub Routing	1,2

Dag	Emne	Øvelse m.m.	Kapiteltests
		Lab 2-3, EIGRP for IPv6 Lab 2-4, Named EIGRP Configuration	
2	Se videoer (OSPF) under 'Routing (advanced)' på NetworkMadeEasy inden du møder Opsamling ch 1, 2 test 8.00 – 15.00 OSPF Opsamling ch. 1, 2 test	Lab 3-1, OSPF Virtual Links Lab 3-2, Multi-Area OSPFv2 and OSPFv3 with Stub Area Lab 3-3, OSPFv3 Address Families	3
3	9.45 – 15.00 Redistribution (6L) Opsamling ch 3 test	Lab 4-1, Redistribution Between EIGRP and OSPF Lab 4-2, Controlling Routing Updates Lab 4-3, Redistribution Between EIGRP for IPv6 and OSPFv3	4
4	8.00 – 12:30 Path Control + NAT 14.00 Opsamling ch 4 test	Lab 5-1, Configure and Verify Path Control Using PBR Lab 5-2, Configure IP SLA Tracking and Path Control	5
	kapiteltestene åbnes hele weekenden		
5	8.00 – 13.15 Enterprise Internet Connection Opsamling ch 5 test	Lab 6-1, Configure NAT Services	6
6	Se videoer (BGP) under 'Routing (advanced)' på NetworkMadeEasy inden du møder 8.00 – 15.00 BGP Opsamling ch 6 test	Lab 7-1, Configuring BGP with Default Routing Lab 7-2, Using the AS_PATH Attribute Lab 7-3, Configuring IBGP and EBGP Sessions, Local Preference, and MED Lab 7-4, IBGP, Next Hop and Synchronization Lab 7-5, Configuring MP-BGP	7
7	8.00 – 15.00 Security Opsamling ch 7 test	Lab 8-1, Secure the Management Plane Lab 8-2, Routing Protocol Authentication	8
8	09.45 – 15.00 Opsamling ch 8 test	Challenge lab	9,10
9	08.00 – 12.30	Challenge lab	
	kapiteltestene åbnes hele weekenden		
10	8.00 – 13.15 Challenge Lab / Praktisk prøve		
11	8.00 – 15.00 Challenge Lab / Praktisk prøve		
12	8.00 – 15.00 Oprydning, Feedback og karaktergivning		

Netværksteknologi II

v. 1,0

Nummer: 16872-3
Varighed: 12½ dage.

Målpinde:

1. Eleven kan redegøre for grundlæggende switching terminologi og begreber.
2. Eleven kan overskue et større campus netværks design, inklusive topologi, switchenes roller, og forskellene mellem lag 2 and multilayer switche.
3. Eleven kan anvende VLANs, VTP, Trunking, og port-channeling.
4. Eleven kan anvende Spanning Tree Protokollen.
5. Eleven kan implementere InterVLAN routing vha. Switch Virtuel Interface (SVI) og routed ports.
6. Eleven kan implementere DHCP i et multilayer netværk.
7. Eleven kan redegøre for og implementer first-hop redundancy protocols i et multilayer switched netværk.
8. Eleven kan redegøre for de forskellige former for network management og switch sikkerhed vha. AAA, NTP, 802.1x, og SNMP.
9. Eleven kan anvende avancerede funktioner til at forbedre campus netværk modstandsdygtighed og tilgængelighed.
10. Eleven kan redegøre for etablering af fysisk switch redundans f.eks. med Stackwise, VSS, eller redundante routermoduler.
11. Eleven kan implementere avancerede sikkerhedsfunktioner.

Vejledning:

Niveau og indhold svarer til CCNP R&S Implementing Cisco IP Switched Networks v7.1 eller nyere.

Med udgangspunkt i Netacad's CCNP-SWITCH v7.1 kursus:

Teoribog: Implementing Cisco IP Switched Networks (SWITCH) Foundation Learning Guide, CCNP SWITCH 300-115.

Planen for gennemlæsning af bogen se dagsplanen.

Det kan forventes at du hver dag har hjemmearbejde der skal løses.

Læseskema Netværksteknologi II (CCNP-SWITCH)

De angivne øvelser er minimum – Men mangler de ekstra opgaver så hav nogle ekstra i baghånden.

Arbejdsbeskrivelse:

Råd til eleven:

1. **Forud** for emne – dvs. *inden du møder* til undervisning - Se eventuelle youtube videoer som handler om det emne du skal have næste dag. www.youtube.com/NetworkMadeEasy
2. Lav projektopgave/fejlfinding på fysisk udstyr (gruppearbejde GNS3/VIRL kan evt. bruges når du ikke er i klassen. Se evt. videoer der kan hjælpe dig med hvordan GNS3/VIRL installeres.
3. Læs de angivne sider i bogen.

4. **Tag kapiteltests første gang** (gruppearbejde) undlad at svare på spørgsmål du intet ved om

Se hvilke emner du svarede forkert på i kapiteltesterne og læs (max 30 min.) Tag kapiteltest igen (hjemmearbejde - kan undlades hvis 100% første gang)

5. Gennemgår de spørgsmål i kapiteltesten der volder problemer. Hvis der er spørgsmål/emner i kapiteltesten der er svære at forstå så kig nærmere på problemstillingen sammen.

Dag	Emne	Øvelse m.m.	Kapiteltests
1	11.45 – 15.00 Fundamentals Review Network Design	Chapter 1 Lab - Preparing the switch	1,2
2	8.00 – 15.00 Campus Architecture	Lab 3-1 Static VLANs, Trunking and VTP	3

Dag	Emne	Øvelse m.m.	Kapiteltests
	11.00 Opsamling ch. 1, 2 test	Lab 3-2 Etherchannel	
3	8.00 – 13.15 Spanning Tree Opsamling ch. 3 test	Lab 4-1 Implement Spanning Tree Protocols Lab 4-2 Multiple Spanning Tree 4	4
Weekend	kapiteltestene åbnes hele weekenden		
4	8.00 – 13:15 Inter VLAN Routing 14.00 Opsamling ch 4 test	Lab 5-1 Inter-VLAN Routing Lab 5-2 DHCP	5
5	8.00 – 15.00 First Hop Redundancy 10.00 Opsamling ch. 5 test	Lab 6-1 First Hop Redundancy Protocols (HSRP, VRRP) Lab 6-2 Hot Standby Router Protocol for IPv6 Lab 6-3 Gateway Load Balancing Protocol (GLBP)	6
6	8.00 – 15.00 Network Management 10.00 Opsamling ch. 6 test	Lab 7-1 Synchronizing Campus Network Devices using NTP Lab 7-2 Configure Campus Network Devices to support SNMPv3	7
7	09.45 – 15.00 Switching Features 10.00 Opsamling ch 7 test	Lab 8-1 IP Service Level Agreements and Remote SPAN	8
8	8.00 – 13.15 High Availability Campus Network Security 10.00 Opsamling ch 8 test	Lab 10-1 Securing Layer 2 Switches Lab 10-2 Securing VLANs	9,10
Weekend	kapiteltestene åbnes hele weekenden		
9	8.00 – 15.00 Case Study		
10	8.00 – 15.00 Case Study		
11	8.00 – 15.00 Praktisk prøve		
12	9.45 – 15:00 Praktisk prøve		
13	8.00 – 13:15 Oprydning, Feedback og karaktergivning		

Netværksteknologi III

v. 1,0

Nummer: 16873-3
Varighed: 12½ dage.

Målpinde:

1. Eleven kan anvende anerkendte og gennemprøvede fejlfindingsprincipper samt almindelige fremgangsmåder.
2. Eleven kan anvende struktureret fejlfinding og redegøre for dens delprocesser.
3. Eleven kan anvende fejlfinding i dagligdags vedligeholdelsesarbejde.
4. Eleven kan indsamle information om lag 2 og 3 med relevante kommandoer.
5. Eleven kan redegøre for specialiserede værktøjer til at opsamle trafik, samle information (f.eks. SNMP og NetFlow) og modtage eventnotifikationer (EEM).
6. Eleven kan udarbejde dokumentation af netværket med henblik på fejlfinding og uddybende forståelse.

Vejledning:

Niveau og indhold svarer til CCNP R&S Troubleshooting and Maintaining Cisco IP Networks v7.1 eller nyere.

Niveau og indhold svarer til CCNP R&S Troubleshooting and Maintaining Cisco IP Networks v7 eller nyere.

Med udgangspunkt i Netacads CCNP-TSHOOT v7.1 (eller nyere) kursus:

Teoribog: Troubleshooting and Maintaining Cisco IP Networks (TSHOOT) – Foundation Learning Guide, CCNP TSHOOT 300-135.

Planen for gennemlæsning af bogen se dagsplanen. Det kan forventes at eleven hver dag har hjemmeopgaver der skal løses.

Arbejdsbeskrivelse:

Råd til eleven:

1. Lav projektopgave/fejlfinding på fysisk udstyr (gruppearbejde GNS3/VIRL kan evt. bruges når du ikke er i klassen. Se evt. videoer der kan hjælpe dig med hvordan GNS3/VIRL installeres.
2. Læs de angivne sider i bogen.
3. **Tag quiz første gang** (gruppearbejde) undlad at svare på spørgsmål du intet ved om *Optional: hvis du har tid - Se hvilke emner du svarede forkert på i quizerne og læs (max 30 min.) Tag quiz igen (hjemmearbejde - kan undlades hvis du fik 100% første gang).*

Dag	Dagsplan	Quiz
1	08.00 - 13.59 Projekt opgave 14.00 - 15.54 læs kap. 1 og 2 i bogen - fortsætte på projekt opgaven når eleven er færdig	
2	08.00 - 09.00 Tag quiz (gruppearbejde) 09.00 - 12.00 Projekt opgave 12.00 - 13.00 Opsamling på quiz 1, 2 13.00 - 14.00 Projekt opgave 14.00 - 15.54 læs kap. 3 i bogen - fortsætte på projekt opgaven når eleven er færdig	1,2
3	08.00 - 08.30 Tag quiz (gruppearbejde) 08.30 - 12.00 Projekt opgave 12.00 - 13.00 Opsamling på quiz 3 13.00 - 14.00 Projekt opgave 14.00 - 15.54 læs kap. 4 og 5 i bogen - fortsætte på projekt opgaven når eleven er færdig	3
4	08.00 - 09.00 Tag quiz (gruppearbejde) 09.00 - 12.00 Projekt opgave 12.00 - 13.00 Opsamling på quiz 4, 5 13.00 - 14.00 Projekt opgave 14.00 - 15.54 læs kap. 6 i bogen - fortsætte på projekt opgaven når eleven er færdig	4,5
5	08.00 - 08.30 Tag quiz (gruppearbejde) 08.30 - 12.00 Projekt opgave 12.00 - 13.00 Opsamling på quiz 6 13.00 - 14.00 Projekt opgave 14.00 - 15.54 læs kap. 7 i bogen - fortsætte på projekt opgaven når eleven er færdig	6
Weekend	Quizzerne åbnes hele weekenden.	
6	08.00 - 08.30 Tag quiz (gruppearbejde) 08.30 - 12.00 Projekt opgave 12.00 - 13.00 Opsamling på quiz 7 13.00 - 14.00 Projekt opgave 14.00 - 15.54 læs kap. 8 i bogen - fortsætte på projekt opgaven når eleven er færdig	
7	08.00 - 14.00 Oplæg til fejlfindingsprocedure med efterfølgende fejlfinding på projekt opgaven 14.00 - 15.54 læs kap. 8 i bogen - fortsætte på projekt opgaven når eleven er færdig	7
8	08.00 - 08.30 Tag quiz (gruppearbejde) 08.30 - 12.00 Fejlfinding på projekt opgaven 12.00 - 13.00 Opsamling på quiz 8 13.00 - 14.00 Fejlfinding på projekt opgaven 14.00 - 15.54 læs kap. 9 i bogen - fortsætte på projekt opgaven når eleven er færdig	8
9	08.00 - 14.00 Fejlfinding på projekt opgaven 14.00 - 15.54 læs kap. 9 i bogen - fortsætte på projekt opgaven når eleven er færdig	
10	08.00 - 08.30 Tag quiz (gruppearbejde) 08.30 - 12.00 Fejlfinding på projekt opgaven 12.00 - 13.00 Opsamling på quiz 9 13.00 - 14.00 Fejlfinding på projekt opgaven 14.00 - 15.54 læs kap. 10 i bogen - fortsætte på projekt opgaven når eleven er færdig	9
Weekend	Quizzerne åbnes hele weekenden.	
11	08.00 - 08.30 Tag quiz (gruppearbejde) 08.30 - 12.00 Fejlfinding på projekt opgaven 12.00 - 13.00 Opsamling på quiz 9 og 10 13.00 - 15.54 Fejlfinding på projekt opgaven	10
12	10.00 - 12.00 Teoretisk prøve 12.00 - 15.00 Oprydning, Feedback og karaktergivning.	

Backupteknologi

v. 1,0

Nummer: 16874-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan i samarbejde med andre installere, planlægge og konfigurere en automatisk backupløsning i et klient/server miljø og kan herigennem demonstrere viden, færdigheder og kompetencer på et rutineret niveau.
2. Eleven kan genindlæse backup fra backupmedier og udføre systemgenoprettelse.
3. Eleven kan anvende strukturerede metoder til fejlsøgning- og fejludbedring af backup.
4. Eleven kan redegøre for og deltage i opsætningen af storageløsninger til backup som f.eks. DAS, NAS, SAN og Cloud.
5. Eleven kan redegøre for anvendelsen af ISCSI og Fiberchannel.
6. Eleven kan redegøre for storagevirtualization.
7. Eleven kan redegøre for principperne bag begrebet TCO (Total Cost of Ownership).
8. Eleven kan redegøre for backupstrategier i et server/klient miljø som indeholder overvejelser omkring backuphyppighed, mediekapacitet, medieperformance, portability og availability.
9. Eleven kender strategier for håndtering af storage management.
10. Eleven kan vurdere fordele og ulemper ved en cloudbaseret backup.

Vejledning:

Eleverne skal i løbet af kurset installere, konfigurere og administrere en komplet Enterprise backupløsning baseret på f.eks. (DPM - Data Protection Manager, Veeam eller lignende), som den ville se ud i et virkeligt datacenter.

Der anbefales at give eleverne mulighed for at bruge et SAN eller NAS som understøtter ISCSI i deres setup.

Med TCO (Total Cost of Ownership) menes metoder for minimering af anskaffelses-, opgraderings- og vedligeholdelsesomkostninger.

Cloudteknologi

v. 1,0

Nummer: 16875-2
Varighed: 5 dage.**Målpinde:**

1. Eleven kan redegøre for sikkerhed i cloudteknologier, herunder datasikkerhed, opetid mv.
2. Eleven kan redegøre for cloudbegreberne: Public cloud, Private cloud, Hybrid cloud, herunder fordele og ulemper ved de forskellige typer i forhold til hinanden og i forhold til et on-site system.
3. Eleven kan redegøre for begrebet IAAS (Infrastructure As A Service).
4. Eleven kan redegøre for begrebet SAAS (Software As A Service).
5. Eleven kan redegøre for begrebet PAAS (Platform As A Service).
6. Eleven kan søge information om love og forordninger i forhold til anvendelse af Cloudteknologier.

Vejledning:

Det er væsentligt at eleverne bliver klar over de forskellige begreber i Cloud (PAAS, SAAS, IAAS) og Public, Private, Hybrid Cloud, samt kan udpege kendetegn, fordele og ulemper ved de forskellige typer.

Der skal arbejdes med praktiske eksempler på Cloud, herunder IAAS, PAAS, SAAS i et virkelighedsnært miljø. Man kan lade eleverne opbygges deres egen Cloud og integrere den i forskellige sammenhænge med de øvrige fag. Der kan f.eks. bygges Cloud ud fra Windows og Linux servere. Man kan også lade HyperV, VMWare eller lignende, danne basis for Cloud.

Som inspiration kan at man evt. benytte Amazon Cloud services, Microsoft Azure, Google Cloud services, O365, mv. de etablerede udbydere har flere forskellige pris modeller, nogle tilbyder gratis trials i en periode.

Andre Cloud services kan være webmail, web-drev, online scanning for virus mv. video/film tjenester osv.

Det er vigtigt at eleverne bliver klar over de udfordringer der er med sikkerhed, backup mv. og ansvarsfordelingen ved brug af Cloud.

Omkostnings niveau og retabilitet er også noget af det som eleverne bør lære om, det anbefales at de laver sammenligninger på latency, delay, services, priser, SLA mv. for at kortlægge hvad en Cloud service reelt koster i forhold til on-premise løsninger og herigennem bliver i stand til at lave en cost-benefit analyse.

Det er tanken at man ser på EU's dataforordning i forhold til anvendelse/placering af data. Man kan med fordel kombinere faget med SQL, Linux og Windows server fagene og benytte sig af CLOUD servere i fagene, gerne i kombination med de traditionelle løsninger. Det vil være ønskeligt at eleverne prøver at tænke Cloud ind i nye måder at løse problemer på. Eleverne bør også prøve at tænke Cloud på en anden og ny måde, "hvad kan man forestille sig" og "hvad kunne vi tænke os af Cloud services – hvad ville være lækkert at kunne" og på den måde få startet en innovativ proces, hvor de selv stiller krav og finder på nye løsninger og anvendelser.

Links til inspiration:

Amazon Cloud

<https://aws.amazon.com/education/awseducate/>

O365

<https://support.office.com/en-us/article/Get-started-with-Office-365-Education-AB02ABE5-A1EE-458C-B749-5B44416CCF14>

Azure

<https://www.microsoft.com/en-us/education/products/azure/default.aspx>

Google Cloud

https://edu.google.com/higher-ed-solutions/google-cloud-platform/?modal_active=none

Deployment Service

v. 1,0

Nummer: 16876-2
Varighed: 5 dage.

Målpinde:

1. Eleven kan redegøre for fordele og ulemper forbundet med forskellige Deployment metoder.
2. Eleven kan installere og konfigurere Deployment Service til brug ved udrulning af software og operativsystemer over netværk.
3. Eleven kan installere og tilrette et OS til brug som basis for et Deployment image.
4. Eleven kan producere pakker og images til udrulning af software og operativsystemer over netværk.
5. Eleven kan tilføje reference images og nødvendige device drivere til et Deployment Share via Deployment Workbench.
6. Eleven kan oprette og tilrette Task Sequences i forbindelse med et givent Deployment scenarie.
7. Eleven kan administrere forskellige roller ud fra individuelle behov, som tredje parts software, hardware, specifikke device drivers osv.
8. Eleven kan administrere og tilrette software på liveinstallationer via Group Policies.

Vejledning:

Faget svarer til 6292 dog med mindre ændringer i målpinde 1 og 2.

Kravspecifikation:

- Man kan med fordel bruge en Hypervisor Type 2, f.eks. VMWare Workstation eller Virtual Box, til at gennemføre undervisningen.
- Windows Server 2016 standard med følgende roller:
- Active Directory
- DNS
- DHCP
- WDS (evt. ekstra disk D, hvorpå WDS installeres)

(Opret evt. en dedikeret AD server med rollerne DNS og DHCP og en Membersserver til WDS)

- Windows 10
- ADKsetup (bla. Windows System Image Manager)
- Evt. MicrosoftDeploymentToolkit2013_x64 (MDT)

AD 1) Eleven kan selvstændigt opbygge et lille klient/servermiljø, hvor Deployment teknologien bruges på forskellige OS med tilrettede Images.

AD 2) Der er flere deployments teknologier på markedet, som f.eks. Windows Deployment Services (WDS). Eleven kan redegøre for fordele og ulemper forbundet med forskellige Deployments metoder.

AD 3) Installer og konfigurer WDS på server. Sørg for at PXEClient er tilføjet i DHCP servicen.

AD 4 & 5) Eleven skal kunne uploade en standard boot og install Image (VIM) til WDS server. Eleven skal også kunne oprette et "capture" boot Image, hvor OS tilpasses på klienten i audit mode, afsluttes med en sysprep og uploades til en WDS server.

Eleven kan hente og bruge driver pakker, hotfixes, service packs, security updates mm. til et givent Image - via et sharepoint på en server.

AD 6, 7 & 8) Eleven kan ved hjælp af "Windows System Image Manager" (Windows Kits), som er en del af ADKsetup, generere en eller flere Task Sequences i form af "unattended" fil(er) med tilretninger af Image og tilføjelse af drivere og programmer, som efterfølgende tilknyttes et givent Image på WDS serveren.

Eksempel på en unattended fil: <https://www.youtube.com/watch?v=VcCHTVp0FAA>

I stedet for at bruge "Windows System Image Manager" kan "DeploymentWorkbench", som er en del af MicrosoftDeploymentToolkit2013_x64, bruges til at importere et OS Images, hvor drivere og programmer tilføjes og Task Sequences oprettes, gældende for et

givent Deployment scenarie. Sluttelig genereres en Boot.WIM, som bla. kan afvikles fra en WDS server og med installation fra et deployment share..
AD 9) Via GPO kan programmer installeres automatisk, dog kun med MSI og ZAP filer. Der findes en del MSI Creator Tools, som med mere eller mindre succes, kan omdanne f.eks. en EXE install fil til en MSI fil.

Inspirationsvideoer:

<https://www.youtube.com/watch?v=SVdZufpnO68>

<https://www.youtube.com/watch?v=lwHyeqVOP2Y>

IT Service Management II

v. 1,0

Nummer: 16877-3

Varighed: 5 dage.

Målpinde:

1. Eleven kan identificere og udvælge sager, der skyldes ukendte årsager, til videre undersøgelse og diagnose
2. Eleven kan undersøge, diagnosticere og enten omgå eller løse komplekse fejl baseret på problemløsningsmetoderne indlært under IT Service Management I faget
3. Eleven kan uddrage og formidle essensen af et problemløsningsforløb
4. Eleven kan udarbejde, strukturere, evaluere og vedligeholde relevant og brugbar viden i form af løsningsbeskrivelser og procedurer
5. Eleven kan ved it-ændringer vurdere og beskrive risici i form af trusler, sårbarheder og konsekvenser for såvel funktionalitet som tilgængelighed, kapacitet, performance, sikkerhed og beredskab
6. Eleven kan med udgangspunkt i risikovurderingen udvælge og gennemføre passende sikringsforanstaltninger for it-ændringer, herunder godkendelser, designprincipper, tests, dokumentation og kommunikation samt planer og procedurer for idriftsættelse, verifikation og fall-back
7. Eleven kan redegøre for best practice for change management, test management, deployment management samt release management og anvende denne praksis på konkrete it-ændringer
8. Eleven kan skelne mellem klassiske leverancemodeller (vandfald) og agile leverancemodeller (continual delivery) og redegøre for, hvordan risiko, kvalitet, ressourcer og tid styres i de to former for leverancemodeller

Vejledning:

Faget omfatter primært aktiviteter, der løses i 'back-office', dvs. i 2nd level, 3rd level etc.

Udvikling

v. 1,0

Nummer: 16878-3

Varighed: 5 dage.

Målpinde:

1. Eleven kan redegøre for teknologiske udviklingstrends inden for it-området, og kan anvende denne viden i egen udviklingsproces.
2. Eleven kan redegøre for værktøjer og metoder, der typisk anvendes i forbindelse med udvikling og opbygning af it-systemer.
3. Eleven kan planlægge og tilrettelægge udviklingen og opbygningen af et it-system.
4. Eleven kan gennemføre udvikling og opbygning af it-systemer, og kan herunder vurdere slutresultatets kvalitet i forhold til gældende krav, standarder og normer.
5. Eleven kan anvende strukturerede teknikker i forbindelse med udvikling og opbygning af it-systemer.
6. Eleven kan anvende it-systemer til planlægning, gennemførelse og dokumentation af udviklings- og opbygningsopgaver, samt til systematisk kvalitetskontrol.
7. Eleven kan anvende innovative løsninger i forbindelse med produktudvikling.

Vejledning:

Denne uge udgør sammen med faget afsluttende datateknikere (infrastruktur/programmering) de 5 uger, der skal bruges til en svendeprøve for en datateknikerelev.

Bedømmelsen på dette fag skal være på baggrund af elevernes procesrapport, produktrapport og den fysiske opstilling.

Systemudvikling og projektstyring

v. 1,0

Nummer: 16879-3
Varighed: 10 dage.

Målpinde:

1. Eleven kan på baggrund af egne overvejelser arbejde med de forskellige faser i projektstyring fra etableringen af et projekt, den overordnede planlægning af projektet, opstilling af hoved- og delmål, tidsestimering og deraf udledte ressourceforbrug, samt styring af projekter med tilhørende afrapportering.
2. Eleven kan selvstændigt og ud fra egne beregninger opbygge og idriftsætte et avanceret it-system.
3. Eleven kan tage initiativ til at udarbejde nødvendig produkt- og præsentationsdokumentation i forhold til præsentation af systemet.
4. Eleven kan selvstændigt forberede og i plenum gennemføre en præsentation af systemet.
5. Eleven kan forklare evt. opståede problemer i forhold til opgaveløsning og argumentere for valgte løsninger, også i ikke-rutine situationer.
6. Eleven kan reflektere og vurdere systemudviklingsmetoder og anvende en situationsbestemt metode.

Vejledning:

Eleven skal have erfaring med at planlægge og gennemføre projektforløb på baggrund af en given caseopgave. Eleverne kan anvende relevante dele af Prince 2 eller lignende til projektstyring. Der bør undervises i systemudviklingsmetoder som Waterfall og agile metoder og deres respektive fordele og ulemper. Eleverne skal præsenteres for indhold og opbygning af henholdsvis proces- og produkt rapport og forstå forskellene så grundlaget er på plads inden svendepreven. Rapportskrivning og udarbejdelse af dokumentation bør indgå som en del af undervisningen. Forløbet skal afsluttes med en mini-svendepreve hvor eleven laver en præsentation, holder et oplæg og får feedback på de afleverede rapporter. Forløbet kan med fordel gennemføres i grupper så eleverne kan sparre med hinanden.

Faget skal placeres på sidste hovedforløb inden svendepreven og er tænkt som forberedelse til datateknikerens svendepreve.