Tietotekniikan Sovellusprojektien ohje

Jukka-Pekka Santanen

30.1.2017

Tiivistelmä

Sovellusprojektissa 3–5 tietotekniikan opiskelijasta koostuva ryhmä määrittelee, suunnittelee, toteuttaa, testaa ja dokumentoi tilaajalle ohjelmiston ohjaajien valvonnassa. Ohjeessa annetaan neuvoja Jyväskylän yliopiston tietotekniikan syventäviin opintoihin sisältyvien Sovellusprojektien menestykselliseen suorittamiseen. Ohjeen avulla jäsenet pystyvät hallitsemaan projektin eri vaiheita ja tehtäviä sekä laatimaan projektin tuloksia ja läpivientiä kuvaavia dokumentteja. Ohje on saatavissa osoitteista `http://www.mit.jyu.fi/palvelut/sovellusprojektit/projohje.pdf` ja `http://www.mit.jyu.fi/palvelut/sovellusprojektit/projohje.ps`.

Avainsanat: Arvostelu, dokumentointi, esittelytilaisuus, määrittely, ohjelmointi, ohjelmistokehitysprojekti, opiskelijaprojekti, projektin hallinta, projektin läpivienti, projektin päättäminen, projektipalaverit, projektityö, raportointi, ryhmätyö, suunnittelu, testaus.

Sovellusprojektien opiskelijoiden suositellaan käyvän ohjeesta läpi kyseistä asiaa käsittelevät luvut projektin alussa ja lopussa, ennen projektien palavereja ja esittelyjä sekä erityisesti projektien dokumentteja laatiessaan.

Sisältö

1 Johdanto

Sovellusprojekti antaa tietotekniikan opiskelijoille tietotaitoja ja kokemusta työelämän ohjelmistokehitysprojektien läpiviennistä, toimintatavoista ja ryhmätyöstä. Projektissa toteutetaan kurssien harjoitustöitä huomattavasti laajempi ohjelmointisovellus yritykselle, valtion tai kunnan organisaatiolle tai Jyväskylän yliopiston jollekin laitokselle. Projektiorganisaatio koostuu opiskelijoiden muodostamasta projektiryhmästä, tilaajan eli toimeksiantajan edustajista sekä asettajan eli informaatioteknologian tiedekunnan edustajina toimivista vastaavasta ohjaajasta ja teknisestä ohjaajasta.

Projektiohje on suunnattu Jyväskylän yliopiston tietotekniikan syventäviin opintoihin sisältyvien Sovellusprojektien jäsenille. Ohje on muotoutunut ohjelmistokehityksen yleisten toimintatapojen sekä aiempien opiskelijaprojektien yhteydessä havaittujen hyvien ja huonojen kokemusten ja ratkaisujen pohjalta. ”Kantapään kautta oppiminen” on yleisesti hyväksytty tapa, mutta projektiryhmän ei silti kannattane käydä läpi kaikkia aiempien projektiryhmien kokemia karikkoja.

Ohje neuvoo projektin eri vaiheissa laadittavien dokumenttien sisällössä. Ohjeessa kuvataan myös ohjelmistokehityksen tehtäväkokonaisuuksia, projektin kokoontumisia ja arvostelukäytänteitä sekä projektin aloittamisessa ja päättämisessä huomioitavia asioita. Projektissa kehitettävän ohjelmiston ja muiden sovittujen tulosten laadun ja toteutusratkaisujen lisäksi arvostelussa kiinnitetään huomiota projektin läpivientiin ja ryhmän toimintaan.

Sovellusprojektin jäsenten oletetaan aiempien opintojen perusteella hallitsevan ohjelmiston määrittelyssä, suunnittelussa, toteutuksessa ja testauksessa tarvittavia tietotaitoja. Ohjelmistoprojekteja, prosesseja, prosessimalleja sekä projektin suunnittelua ja hallintaa kuvataan luentomateriaalissa [Santanen, 2016], joten niitä ei juurikaan käsitellä ohjeessa.

Luvut on koottu Sovellusprojektin vaiheisiin, kokoontumisiin tai dokumentteihin liittyviksi asiakokonaisuuksiksi. Lukuja ei ole kuitenkaan kirjoitettu täysin erillisiksi itsenäisiksi osioiksi. Jos siis kaipaat neuvoja tiettyyn projektin vaiheeseen tai dokumenttiin, kannattaa kyseisen alaluvun lisäksi tutustua myös muihin asiaa käsitteleviin alalukuihin.

Luvussa 2 annetaan yleisiä ohjeita projektin läpiviemiseen ja sovelluksen kehittämiseen. Luvussa 3 käsitellään projektin tuloksia ja niiden sisällöllisiä vaatimuksia. Kyseisessä luvussa kuvataan projektin ja sovelluksen suunnitteludokumenttien ja raporttien sisältöä, käyttöohjetta ja ohjelmalistauksia sekä ajankäytön raportointia.

Luvussa 4 käsitellään projektiryhmien kokoontumisia ja niistä laadittavia pöytäkirjoja. Projektien arvosteluperusteita ja projektien päättämiseen liittyviä asioita esitellään luvussa 5.

2 Projektin läpivienti ja sovelluksen toteutus

Sovellusprojektit poikkeavat huomattavasti toisistaan mm. kohdealueen, aiheen, tavoitteiden, jäsenten, noudatettavan prosessin ja käytänteiden tai työkalujen osalta. Eroistaan huolimatta projektien tehtäväkokonaisuudet sekä sovellusten määrittelyssä, suunnittelussa, ohjelmoinnissa ja testauksessa huomioitavat asiat ovat pääosin yhteneviä.

2.1 Työmäärä, ajankäyttö ja työnjako

Sovellusprojektin laajuus on 10–15 opintopistettä vastaten 267–400 työtuntia kutakin ryhmän jäsentä kohti. Lisäksi projektiin liittyvät luennot ja perehdytykset on koottu oheiskursseiksi *Projektiviestintä IT-alalla* (3 op) sekä *Sovellusprojektin hallintaa, viestintää ja työkaluja* (1 op). Kunkin projektin jäsenen tulee kirjata ajankäyttöraportissaan oheiskurssien työtunteihin niihin kuuluvat luennot, ryhmätyöt, väliesittelyt ja niiden valmistelun sekä dokumenttien rakenteen ja kirjoitusasun muokkauksen.

Projektin **toteutusaikaa on noin 13–16 viikkoa**, joten jokaisen projektin jäsenen tulee varata **viikottain vähintään 20–30 tuntia** projektityöhön. Projektin jäsenten tulee parin ensimmäisen viikon kuluessa keskenään sopia, kuinka suureen työmäärään he voivat sitoutua Sovellusprojektissa. Aiempien projektien yhteydessä on havaittu projektin vastaavan työmäärältään vähintään kahta laajaa luentokurssia. **Projektin vaatima työmäärä on ehdottomasti otettava huomioon kyseisen lukukauden opintosuunnitelmaa laadittaessa ja muita sitoumuksia sovittaessa.**

Sovellusprojektin työmäärää ei pystytä kunnolla arvioimaan projektin alussa, joten **työhön tulee varata riittävästi aikaa heti alusta lähtien.** Tällä tavoin pystytään vähentämään projektin lopun kiireitä sekä välttämään projektin viivästyminen. Työmäärän arvioinnin hankaluus johtuu pääosin eroista aiheissa ja kehitysvälineissä sekä jäsenten, ohjaajien ja tilaajan edustajien tiedoissa ja taidoissa. Lisäksi tilaajan ja käyttäjien toiveiden ja tarpeiden jäsentymättömyys saattaa vaatia arvioitua enemmän panostusta vaatimusten määrittelyyn.

Sovellusprojektien loppuesittely järjestetään syyslukukausina joulukuun puolessa välissä ja kevätlukukausina toukokuun puolessa välissä. Kehitettävä sovellus voi viivästyä ennakoimattomien ongelmien vuoksi. Tästä johtuen sovellus tulee suunnitella valmistuvan vähintään kaksi viikkoa ennen loppuesittelyä. Välttämättä projektien dokumentointi ei ole kokonaan valmis ja hyväksytty loppuesittelyyn mennessä, joten **projektille tulee varata ”pelivaraa” vähintään pari viikkoa loppuesittelyn jälkeenkin**. **Projektin mahdollinen viivästyminen tulee ehdottomasti ottaa huomioon mm. työtehtävistä sopiessaan ja lomamatkoja varatessaan.**

Kaikkien projektiryhmän jäsenten tulee osallistua ohjelmistoprojektin eri tehtäväkokonaisuuksiin, joita ovat projektin suunnittelun, hallinnan ja raportoinnin ohella sovelluksen määrittely, suunnittelu, toteutus ja testaus sekä niihin sisältyvä dokumentointi. Tällä tavoin ryhmän kaikki jäsenet saavat riittävän kuvan projektin läpiviennistä ja ohjelmistokehityksestä kokonaisuutena. Projektin jäsenten tehtävien tai vastuualueiden liian korostunut jako saattaa laskea arvosanaa. Toisaalta kaikkia tehtäviä ei tule suorittaa yhdessä, vaan **tehtävien jako on olennainen osa projektia**.

Projektiryhmän tulee valita keskuudestaan projektipäällikkö. Tehtävää voidaan tarvittaessa kierrättää projektin jäsenten kesken, mutta tällöin tehtävän opiskeluun ja siirtämiseen tulee varata riittävästi aikaa. **Projektipäällikön olennaisimmat tehtävät** ovat projektin läpiviennin suunnittelu, hallinta ja raportointi, projektin tehtävien jakaminen ja hallinta, projektin ja sovelluksen tilan säännöllinen määrittely sekä tiedottaminen projektiryhmälle ja -organisaatiolle. **Projektipäällikön tehtävä tuo vastuun ohessa valtaa**, joka tulee näkyä päällikön ja ryhmän jäsenten toimintatavoissa.

2.2 Määrittelyn ja suunnittelun päämäärät ja edut

Projektityö aloitetaan tutustumalla aiheeseen ja sen taustoihin, tilaajan kokonaistavoitteisiin ja käyttäjien tarpeisiin, sovelluksella tuettavaan prosessiin sekä ohjelmointityökaluihin. **Projektiryhmä laatii ensimmäisen 3–6 viikon aikana vaatimusmäärittelyn sekä projekti- ja sovellussuunnitelman** aiheesta saadun materiaalin sekä projektipalavereissa suoritetun tavoitteiden ja vaatimusten tarkennuksen ja rajauksen pohjalta.

Riittävän huolellisesta määrittelystä ja suunnittelusta sekä ohjelmien asteittaisesta tarkentamisesta on havaittu olevan runsaasti hyötyä aiemmissa projekteissa. Niiden jäsenet ovat projektinsa jälkeen painottaneet määrittelyn ja suunnittelun tärkeyttä erityisesti virheiden ja ongelmien välttämisessä. Suunnitteluvaiheessa väärällä tavalla "säästetty" työmäärä kuluu moninkertaisesti projektin lopussa heikkojen toteutusratkaisujen ja virheiden jäljittämiseen ja korjaamiseen.

Toisaalta monesti tilaaja pystyy antamaan palautetta vasta ohjelmisto(demo)n perusteella, joten liian yksityiskohtainen suunnittelu puutteellisin tiedoin ei yleensä ole järkevää. Sovelluksen toteuttaminen kannattaa aloittaa mahdollisimman nopeasti, jotta tilaajan edustajat pystyvät antamaan palautetta sovelluksesta ja mahdollisiin muutoksiin pystytään reagoimaan aikaisin. **Suunnitelmat yleensä tarkentuvat ja muokkautuvat tilaajan tavoitteiden tarkentuessa sekä jäsenten tietojen ja taitojen karttuessa.**

Suunnittelu helpottaa ja nopeuttaa huomattavasti Sovellusprojektien kaltaisten laajojen ja monimutkaisten sovellusten kehittämistä sekä projektin läpiviennin hallitsemista. Suunnitteludokumenttien avulla projektin jäsenet, ohjaajat ja tilaajan edustajat pystyvät seuraamaan työnsä edistymistä ja pitämään mielessä projektin olennaisimmat tavoitteet. Suunnitelmien laatimisen kautta projektin jäsenet joutuvat tutustumaan aihealueeseen, käyttäjien tarpeisiin, projektin tavoitteisiin ja käytettäviin työkaluihin.

Suunnitteludokumentit edesauttavat projektiryhmän sekä ohjaajien ja tilaajan edustajien välille muodostuvan **yhteisen kielen kehittymistä**. Niiden avulla voidaan havaita projektin jäsenten, ohjaajien ja tilaajan edustajien erilaisia ja virheellisiä käsityksiä jo projektin alkuvaiheissa, jolloin niihin voidaan puuttua riittävän ajoissa. Riittävän määrittelyn ja suunnittelun avulla voidaan säästää runsaasti aikaa ja resursseja projektin myöhemmissä vaiheissa.

Vaatimusmäärittelyn ja sovellussuunnitelman voidaan katsoa olevan projektiryhmän, ohjaajien ja tilaajan välinen sopimus toteutettavasta sovelluksesta. Suunnitteluun tukeutuen saadaan tavoitteita vastaava ja laadukas sovellus. Lisäksi ajatuksella laadittujen ja huolellä ylläpidettyjen suunnitteludokumenttien pohjalta projektin ja sovelluksen loppuraportointi on suoraviivaista, sillä monesti se voidaan toteuttaa pienin muokkauksin ja suunnitteludokumentteja täydentämällä.

On kuitenkin huomattava, että **toteutettu sovellus ja läpiviety projekti eivät koskaan täysin vastaa suunniteltua**. Ensimmäisten (kuten myöhempienkin uutta tekniikkaa soveltavien) ohjelmistoprojektien yhteydessä väistämättä projektin jäsenet eivät pysty huomioimaan kaikkia oleellisia asioita. Projektin jäsenten tulee siis asennoitua mahdollisiin muutoksiin tietojen ja taitojen karttuessa projektin edetessä. Projekti- ja sovellussuunnitelman laatiminen kannattaa toteuttaa rohkeasti, sillä ne joka tapauksessa antavat tukevan perustan projektin läpiviennille ja sovelluksen toteuttamiselle. Muutoksista suunnitteludokumentteihin voidaan tarvittaessa sopia projektipalavereissa, kunhan sovitut muutokset tulevat dokumentoitua ko. dokumentteihin ja palaveripöytäkirjoihin.

Määrittelyn ja suunnittelun vastuualueita ja tehtäviä kannattaa jakaa ryhmän jäsenten kesken. **Kaikkien projektiorganisaatioon kuuluvien tulee olla tietoisia kunkin ryhmän jäsenen suunnitelmista, ratkaisuista ja tuloksista**, joten niistä tulee tiedottaa oma-aloitteisesti. **Suunnittelun aikana yhden ryhmän jäsenistä tulisi tutustua ohjelmointityökaluihin** etenkin, jos ne ovat jäsenille uusia. Tämä nopeuttaa ohjelmoinnin aloittamista, ja kehitysvälineisiin tutustunut pystyy tarvittaessa perehdyttämään muita jäseniä. Pelkän kokeilun sijaan kannattaa ohjelmointikielellä ja/tai sovelluskehitystyökalulla toteuttaa sovelluksesta demo, prototyyppi tai jonkin suunnitellun osion ensimmäinen versio.

2.3 Ohjelmointi, kommentointi ja varmuuskopiointi

Puutteelliset tiedot ja taidot sekä käytännön ongelmat ilmenevät yleensä vasta sovellusta **ohjelmoitaessa**. Siksi ohjelmointia sisältäviin projektin vaiheisiin tulee **varata vähintään yhdeksän viikkoa**. **Tärkeää on koko projektin ajan pitää mielessä tilaajan kokonaistavoitteet sekä ohjelman käyttäjilleen tarjoamat tiedot ja toiminnot.** Säännöllinen vaatimusten ja suunnitelmien läpikäynti sekä tarvittaessa täydentäminen ja muokkaaminen auttavat käyttäjien tarpeiden sekä tilaajan tavoitteiden mielessäpitämistä ja tarkentamista.

Erityisesti tulee muistaa **dokumentoida ja kommentoida ohjelmaosiot** jo niitä ohjelmoitaessa. Tällöin suunniteltu ohjelmaosion toiminta muistuu myöhemmin nopeasti mieleen, sillä yleensä ohjelmoija unohtaa vaativan toteutusratkaisun muutamassa päivässä. **Ohjelmakoodit ja -kommentit tulee kirjoittaa englanniksi.**

Monimutkainen tai ”jippomainen” ohjelmakoodi ei tarkoita sitä, että sovelluksen toteutusratkaisut ovat laadukkaita, ylläpidettäviä ja jatkokehitettäviä. Yleensä tällöin on kyse täysin päinvastaisesta. **Lähdekoodi tulee laatia siten, että se on** kääntäjäohjelman ohella **kenen tahansa ohjelmoijan helposti tulkittavissa**. Lähdekoodin modulaarisuus, yksinkertaisuus ja yhdenmukaisuus sekä selkeä rakenne, nimeäminen ja kommentointi ovat siten oleellinen osa ohjelmointia.

Lähdekooditiedostot ja muut tulokset tulee riittävän usein sijoittaa versiohallintaan tai varmuuskopioitavalle verkkolevylle. Tällöin voidaan tarvittaessa palata aiempaan versioon, jos tehdyt muutokset osoittautuvat huonoiksi. Lisäksi voidaan välttää päivien tai jopa viikkojen työn menetys virheen tai kovalevyn rikkoutumisen vuoksi, josta on kokemusta parista aiemmasta projektista. Erityisesti yrityksille toteutettavien projektien yhteydessä on huomattava, että **salassapitosopimus edellyttää lähdekoodi- ja dokumenttitiedostojen huolellista käsittelyä**.

2.4 Sovelluksen testaus

Kukin **ohjelmaosio on pyrittävä testaamaan erillisenä** ns. yksikkö- eli moduulitestauksella välittömästi valmistuttuaan, mikäli se vain on mahdollista. Kokemus on osoittanut, että tämä säästää runsaasti projektin jäsenten kokonaistyöaikaa sekä vähentää virheenjäljittämistä ohjelmaosioita yhdistettäessä ja sovellusta kokonaisuutena testattaessa.

Pitää kuitenkin muistaa, että yhdistettäessä ohjelmaosioita kokonaisuudeksi, ne vaikuttavat toistensa toimintaan. Yhdistettyjen ohjelmaosioiden yhteistoiminta tulee varmistaa integraatiotestauksella. Ohjelmistojenkin osalta siis pätee se, että kokonaisuus on enemmän kuin osiensa summa.

Ohjelmisto kokonaisuutena tulee saattaa järjestelmätestausvaiheeseen hyvissä ajoin (vähintään kolme viikkoa) ennen loppuesittelyä. Projektiryhmän kannattaa suorittaa järjestelmätestauskerta aina julkistaessaan ohjelmistosta uuden version. Tällä vähennetään virhekorjauksiin tarvittavia työtunteja sovellusta viimeisteltäessä. Tilaajan edustajille ja ohjaajille tulee antaa riittävästi aikaa sovelluksen koekäyttämiseen.

Sovellusta tulee mahdollisuuksien mukaan **testata eri käyttöjärjestelmissä, mikro- ja ohjelmistokokoonpanoissa sekä erityisesti käyttöympäristössä, johon ohjelmistokehitystyökalua ei ole asennettu**. Testauksessa tulee myös huomioida mm. käyttöjärjestelmään sekä hyödynnettäviin tietojärjestelmiin ja komponentteihin liittyvien asetusten vaikutus ohjelman toimintaan ja käyttöliittymään. Asiakaspalvelinjärjestelmän tapauksessa testausympäristön tulee olla eri kuin kehitysympäristön. Projektiryhmän on mahdollista saada testauskäyttöön erillinen teholtaan heikompi mikro.

Useissa läpiviedyissä Sovellusprojekteissa tilaajan koekäytössä ollut sitä ennen virheettömästi toiminut mutta **liian vähän testattu ohjelmisto ei ole toiminut**. Hallittu testaus ja sen dokumentointi ovatkin siten oleellinen osa sovelluksen kehittämistä.

2.5 Tekijänoikeudet ja muut aineettomat oikeudet

Lähdekoodeissa ja raporteissa tulee selkeästi erotella projektiryhmän (ja haluttaessa yksittäisen projektin jäsenen) toteuttamat osat muualta hankituista ja muokatuista osista. **Kunkin lähdekooditiedoston** tai jopa yksittäisen aliohjelman **alussa tulee mainita tekijöiden nimet kera vuosilukujen** ja mahdollisten lisenssien, jotta tekijänoikeudet ja ohjelman hyödyntämisoikeudet tulevat mahdollisimman selkeästi esille.

Jos projektiryhmä hyödyntää työssään olemassaolevaa lähdekoodia (kuten avoimen lähdekoodin komponentteja, kurssien luentomateriaalia tai harjoitusten esimerkkikoodeja), sen tekijä(t) on aina mainittava kyseisen lähdekooditiedoston alussa. Kyseiseltä tekijältä tulee pyytää kirjallinen lupa koodin käyttöön, ettei riko hänen tekijänoikeuksiaan. Monesti tekijä on joko lähdekoodissa tai erillisessä tiedostossa maininnut koodin hyödyntämisen ehdoista.

Ulkoisten lähdekoodien ja komponenttien käytölle tulee aina saada tilaajan ja ohjaajien hyväksyntä. Epäselvissä tapauksissa kannattaa kysyä neuvoa vastaavalta ohjaajalta. Tekijänoikeuksista ja muista aineettomista oikeuksista löytyy lisätietoa URL-osoitteen `http://www.mit.jyu.fi/santanen/ohjeita.html#oikeudet` kautta.

Sovellusprojekteissa sovitaan oikeuksien siirrosta joko projektin alussa laadittavalla projektisopimuksella (malli [Santanen, 2008]) tai jäsenet sitoutuvat sijoittamaan tuloksensa sovitun vapaan lähdekoodin lisenssin alaisuuteen. Tilaajan vaatimuksesta voidaan joutua laatimaan salassapitoa varten vaitiolosopimus (malli [Santanen, 2006]).

2.6 Projektin läpiviennin ja sovelluksen kehittämisen ohjaus

Projektiryhmän ei kannata taistella vaatimuksiin, toteutusratkaisuihin, ohjelmointiin tai johonkin muuhun liittyvän ongelman kanssa päiväkausia. Tietoa ongelmien ratkaisemiseen kannattaa etsiä verkosta löytyvistä dokumenteista ja projekteille hankituista kirjoista. Teknisen ohjaajan, vastaavan ohjaajan ja tilaajan edustajien puoleen kannattaa rohkeasti kääntyä. Tarvittaessa ohjaajat etsivät neuvojiksi projektiorganisaation ulkopuolisia asiantuntijoita.

Tekninen ohjaaja neuvoo ohjelmointikieleen, ohjelmistokehitystyökaluun, tekniikoihin ja toisinaan myös aihealueeseen liittyvissä ratkaisuissa ja ongelmatilanteissa. Hän valvoo pääasiallisesti lähdekoodin laatua ja kommentoinnin tasoa, joten lähdekoodista tulee pyytää palautetta tekniseltä ohjaajalta erityisesti sovelluksen toteutuksen alku- ja loppuvaiheissa. Lähdekoodit tulee lisäksi katselmoida vähintään kahdesti projektin kuluessa.

Vastaava ohjaaja valvoo projektin läpivientiä määritellyillä resursseilla ja aikataulussa sekä tulosten valmistumista ja laatua kokonaisuutena. Hän siis ohjaa projektin hallittua ja tietoista läpivientiä kohti laadukkaita ja vaatimuksia vastaavia tuloksia. Hän puuttuu myös sovelluksen käytettävyyteen koekäyttämällä sen käyttöliittymää. Hän antaa lisäksi palautetta dokumenttien sisällöstä, rakenteesta ja kieliasusta. Tällä pyritään ohjaamaan projektin jäseniä sujuvaan tekniseen dokumentointiin sekä tukemaan myöhempien teknisten dokumenttien ja tutkielmien laatimista.

Vastaavalta ohjaajalta voi pyytää projektin käyttöön projektikansion ja sen välilehdet, CD-taskut kansioon, CD-aihioita sekä muita projektin tarvitsemia toimistotarvikkeita. Myös projekteissa tarvittavien kirjojen, ohjelmien ja laitteiden hankkimista sekä tarvittavia perehdytyksiä kannattaa ehdottaa vastaavalle ohjaajalle.

Tilaajan edustajat tuntevat yleensä aihealueen parhaiten projektiorganisaatioon kuuluvista. Tästä johtuen projektin sisällöllinen ohjaus tapahtuu suurelta osin tilaajan toimesta. Tilaajan edustajat pystyvät yleensä parhaiten mm. osoittamaan projektissa hyödynnettävät käyttäjät, priorisoimaan sovellukseen toteutettavat ominaisuudet sekä määrittämään käyttöliittymään sisällytettävät tiedot ja toiminnot. Joissakin projekteissa tilaajan edustajat osallistuvat myös ohjelmointikielen ja -työkalun käytön ohjaamiseen.

ATK-tuki vastaa pääosin projektin käyttöön asennettujen mikrojen käyttöjärjestelmien ja ohjelmien asennuksesta ja ylläpidosta sekä verkon ja laitteiden toiminnasta. Pyynnöt ohjelmien asennuksen osalta ja virheilmoitukset sekä niistä saadut vastaukset tulee suunnata ATK-tuen ohella projektiorganisaation jäsenille, jotka ovat täten tietoisia asennuksista ja työtä hidastavista ongelmista.

3 Projektin läpiviennin ja sovelluksen dokumentointi

Olennainen osa ohjelmistoprojektia on projektin sekä siinä toteutettavan sovelluksen ja tulosten riittävä dokumentointi. **Dokumentointia** tulee tehdä koko ajan projektin edetessä, **eikä sitä saa lykätä viimeiselle viikolle**. Lisäksi samaa asiaa ei pääsääntöisesti kannata kuvata useammassa dokumentissa ristiriitaisuuksien välttämiseksi.

3.1 Projektikansio

Sovellusprojektissa julkisiksi sovitut lähdekoodit, dokumentit ja materiaalit kootaan projektikansioon, jonka projektin loputtua vastaava ohjaaja sijoittaa julkisesti nähtäville tietotekniikan laitoksen yleisiin tiloihin. Luottamuksellisia tuloksia sisältävässä projektissa osa tuloksista jää pelkästään tilaajan haltuun.

Tilaajan ja ohjaajien hyväksyttyä projektin yksittäiset tulokset projektiryhmältä kuluu vähintään viikko projektikansion kokoamiseen ja sille hyväksynnän saamiseen tilaajan edustajilta ja ohjaajilta. Viive johtuu osittain tarkastamisen ja hyväksynnän vaatimasta ajasta sekä tilaajan edustajien ja ohjaajien muista sitoumuksista, joten jäsenten tulee huomioida kyseinen ajantarve.

Dokumentointiin kuuluvat sekä CD:lle tallennettuina että paperille tulostettuina **kaikki projektiryhmän tulokset**, joita ovat seuraavat:

- toteutetun sovelluksen, projektiorganisaation ja ohjelmointityökalujen WWW-esittely (sivun `http://www.mit.jyu.fi/palvelut/sovellusprojektit/toteutetut` kautta löytyvät aiempien Sovellusprojektien tiivistetyt esittelyt),
- itsearvioinnit niiltä ryhmän jäseniltä, jotka sallivat sen sijoittamisen kansioon,
- projekti- ja sovellusraportit,
- vaatimusmäärittely,
- käyttö-, asennus- ja käännösohjeet,
- projekti- ja sovellussuunnitelmat sekä muut suunnitteludokumentit,
- projekti- ja vaitiolosopimukset,
- ajankäyttöraportit,
- projektipalaverien esityslistat, pöytäkirjat ja tilakatsaukset,
- väliesittelyjen ja loppuesittelyn esitysmateriaalit, muistiot ja arviointilomakkeet,
- olennaisimmat ryhmätöissä ja perehdytyksissä laadituista materiaaleista,
- lista lähdekooditiedostoista, ohje niiden kääntämiselle, käännösten yhteydessä havaitut virheilmoitukset ja varoitukset sekä tuloksena saatavat ajokelpoiset tiedostot,
- ohjelmalistaukset (sivujen yläreunassa tiedoston nimi ja sivunumero),
- oleellisimmat projektiin liittyvät sähköpostit ja muu kirjeenvaihto, kuten lähdekoodeista ja toteutetun sovelluksen eri versioista annetut palautteet, sekä
- liitteet, joita ovat mm. mahdolliset tilaajan edustajien tai ohjaajien toimittamat dokumentit, kuten projektin aiheen kuvaus.

Em. dokumenteista ja lähdekoodeista kootaan projektin lopullinen tulospaketti eli **projektikansio**. Dokumentit kannattaa sijoittaa kansioon em. järjestyksessä, sillä muut dokumentit ovat lähinnä liitteitä projekti- ja sovellusraportille sekä vaatimusmäärittelylle. **Materiaali kannattaa pääosin tulostaa kaksipuoleisena**, elleivät ohjaajat ja tilaajan edustajat vaadi toisin.

Projektikansioon sijoitetun CD:n ohella laitokselle tulee toimittaa CD:stä arkistokappale. Sisältö tulee kuvata CD:llä esimerkiksi `readme`- tai `lueminut`-tiedostossa. Tiedostot tulee sijoittaa yhteenkuuluviin kokonaisuuksiin hakemistoihin jaettuina. Tietotekniikan laitokselle toteutettavassa projektissa laaditut lähdekoodit ja dokumentit kannattaa tallentaa lisäksi projektin WWW-hakemistoon, YouSourceen ja/tai Sovellusprojektien Trac-wikiin.

On kuitenkin huomattava **salassapitosopimusten sitovuus** erityisesti maksullisten Sovellusprojektien yhteydessä. Tällöin tilaajan vaatimuksia on kuultava ja sovittava laitokselle jäävän kansion sisällöstä sekä dokumenttien että ohjelmakoodin ja ajokelpoisen ohjelman osalta. Osassa Sovellusprojekteista **projektiryhmä laatii ensimmäisen kuukauden aikana projekti- ja vaitiolosopimuksen**, joiden mallit löytyvät WWW-dokumenteista [Santanen, 2008] ja [Santanen, 2006]. Ilmaisissa projekteissa oikeuksiensiirtosopimuksen sijaan jäsenet monesti laativat ja allekirjoittavat sitoumuksen, jolla he lupaavat sijoittaa tulokset sovitun vapaan lähdekoodin lisenssin alle.

Edellisten projektien kansioita on nähtävillä projektisolun kokoushuoneen kirjahyllyssä. Kansioihin tutustuminen on erittäin suositeltavaa etenkin projektin alussa ja kunkin dokumentin kirjoittamisen yhteydessä. Kansioiden kanteen on merkitty kyseisen projektin arvosana. Lisäksi kansiot sisältävät toimeksiantajan edustajan ja vastaavan ohjaajan lausunnot projekteista sekä mahdolliset projektin jäsenten itsearvioinnit.

Projektikansioita ei toivota lainattavan, jotta ne olisivat koko ajan kaikkien kuluvan lukukauden Sovellusprojektien käytössä. Lyhytaikaisen lainauksen yhteydessä tulee lainaajan ja kansion nimet sekä lainapäivämäärä kirjoittaa kansiohyllyn päädyssä olevaan listaan. Kansiota palautettaessa tulee kyseiseen listaan kirjata palautuspäivämäärä.

3.2 Erilliset dokumentit eri käyttötarkoituksiin

Dokumentointi tukee projektin kuluessa sen etenemisen ja tavoitteiden asettamista ja seurantaa. Dokumenttien avulla projektiorganisaatioon kuuluvat ”näkevät metsän puilta”, sillä yksityiskohtien parissa kokonaisuus monesti hämärtyy. Projektin lopussa viimeisteltävien raporttien kautta sovelluksen käyttö ja toteutusratkaisut sekä projektin kokemukset siirretään tulevien kehittäjien tietoon. Lisäksi dokumentointi toimii projektin edetessä vaatimusten ja toteutusratkaisujen kartoituksen tukivälineenä sekä alustavana tehtävälistana. **Ryhmän jäsenten tulee siis säännöllisesti tutustua ja tarvittaessa päivittää laatimiaan dokumentteja, jotta projektin tavoitteet ja tila ovat tiedossa ja ajantasaisina.**

Sovellusprojekti suoritetaan määritellyn aikataulun mukaan. Ennen ohjelmoinnin aloittamista tulee suorittaa vähintään alustava vaatimusmäärittely ja sovelluksen kokonaisrakenteen suunnittelu. **Vaatimusmäärittelyn sekä projekti- ja sovellussuunnitelmien laatimiseen kannattaa varata aikaa 3–6 viikkoa** aiheesta ja valitusta prosessista riippuen. Muitakin kuin ketteriä prosessimalleja noudatettaessa tulee vaatimusmäärittelyä ja suunnitelmia tarkentaa ja päivittää poikkeamia kohdattaessa ja muutoksia tehtäessä.

Projektisuunnitelma keskittyy projektin tavoitteiden, organisaation, ympäristön ja resurssien kuvaamiseen sekä käytänteiden ja läpiviennin suunnitteluun. Joidenkin projektien yhteydessä toteutetaan sovellussuunnitelman sijaan tilaajan mallien mukaiset sovelluksen määrittely- ja suunnitteludokumentit. Joka tapauksessa **projektin ja sovelluksen toteutuksen suunnitelmat on laadittava erillisinä dokumenteina, jotta niiden toteutumista voidaan myös seurata ja hallita toisistaan erillään.**

Projektin lopussa tulee raportoida sekä projektin että sovelluksen toteutuma. Nämä osiot on järkevä jakaa vähintään kahdeksi dokumentiksi. Jotkut tilaajat vaativat oman dokumenttipohjansa käyttöä sovelluksen ominaisuuksien, toiminnan ja rakenteen raportointiin. Kyseisissä raporteissa tulee esittää selkeästi toteuma ja tehdyt muutokset suunnitteludokumentteihin verrattuna. Jos suunnitteludokumentit on järkevästi laadittu, niitä vastaavat raportit voidaan laatia suoraan suunnitelmien pohjalta.

Projektiorganisaatiolle julkistettuihin dokumentteihin, sovellukseen ja lähdekoodeihin tulee liittää **versionumerointi** ja/tai päiväys. Ensimmäinen tilaajan edustajien ja ohjaajien hyväksymä versio on numeroltaan `1.0.0`, joten sitä aiempien versioiden numerointi on muotoa `0.i.j`. Jotta tiedostot olisivat tunnistettavissa projektin ulkopuolellakin, ne suositellaan nimettävän tyyliin `ProjektiDokumenttiVersio.Tarkenne`. Dokumentteihin kannattaa sisällyttää myös muutoshistoria sisältäen muokkaajan nimen lisäksi lyhyet kuvaukset olennaisimmista tehdyistä lisäyksistä ja muutoksista versioiden välillä.

3.3 Projektisuunnitelma

Projektisuunnitelmassa määritellään mm. seuraavat asiat:

- Mistä projektissa on kyse, ja mikä on dokumentin tarkoitus (tiivistelmä- ja johdantoluvut)?
- Aiheeseen ja tavoitteisiin sekä ohjelmointityökaluihin ja -tekniikoihin tutustumisen yhteydessä esiintulleet vieraat ja uudet termit tulee kuvata.
- Projektin taustojen avulla kuvataan projektin sijoittuminen kokonaisuuteen. Tilaajan kokonaistavoitteista ja käyttäjien tarpeista johdetaan projektin tavoitteet.
- Projektin tavoitteet sovelluksen, muiden tulosten ja oppimisen (katso luku 5.1) osalta tulee kirjata. Lisäksi kunkin dokumentin tarkoitus ja sisältö tulee kuvata.
- Projektin resurssit sisältävät organisaation (ryhmän jäsenet, ohjaajat, tilaajan edustajat, ATK-tuki ja muut sidosryhmät), projektin käyttöön annetut tilat, laitteet ja ohjelmat (kera maininnan antajasta) sekä tarvittavat perehdytykset pitäjineen.
- Projektin käytänteet tiedotuksen, dokumentoinnin ja kokouksien osalta tulee kuvata. Erityisesti tulee suunnitella päätöksenteko ja tulosten hyväksyminen sekä tehtävien ja aikataulun tilasta ja muutoksista tiedottaminen. Lisäksi tiedostojen ja hakemistojen nimeämiskäytänteet ja hakemistorakenne tulee suunnitella.
- Sovelluksen testausperiaatteiden tulee sisältää vähintään testattavien kohteiden esittelyn sekä maininnan testauksen suorittajista.
- Projektin tehtävien jakautumista jäsenten kesken tulee vähintään hahmotella vastuualueittain. Näihin sisältyvät projektipäällikön tehtävät sekä projektin kustakin olennaisimmasta tuloksesta vastaava jäsen.
- Projektin tehtäväkokonaisuudet ja tehtävät kera arvioiden niihin kuluvista työtunneista tulee kirjata.
- Projektissa noudatettavaa prosessimallia tai sen läpivientiin räätälöityä prosessia tulee kuvata oleellisilta osin.
- Projektin aikataulu tulee määrittää vaiheitten ja niihin liittyvien tehtävien viikoittaisen aikataulun sekä sovittujen tavoitepistepäivämäärien ja/tai katselmointiajankohtien osalta.
- Riskien kartoituksen, arvioinnin ja hallinnan avulla määritellään riskien toteutumisen haitat, todennäköisyydet ja vakavuudet sekä keinot niiden ehkäisyyn, ennakointiin, seurantaan ja niistä toipumiseen.

Projektisuunnitelma kuvaa projektin ennakoidun läpiviennin, joten se tarjoaa jäsenille työkaluja projektin tilan määrittämiseen, työnohjaukseen, kohdattujen poikkeamien tiedostamiseen ja muutosten hallintaan. Tietotason karttuessa ja tavoitteiden muuttuessa monesti joudutaan muuttamaan projektin läpivientiä suunnitelmassa kuvatusta.

3.4 Vaatimusmäärittely

Vaatimusmäärittely kuvaa, mitä tietoja ja toimintoja sovelluksen tulee tarjota käyttäjille ja muille tietojärjestelmille. Se sisältää yleensä seuraavat sovellukseen liittyvät asiat:

- Projektin tavoitteiden määrittämiseksi ja rajaamiseksi tulee kuvata projektin taustoja ja kokonaisuutta, tilaajan kohdealueeseen liittyviä kokonaistavoitteita, käyttäjien tarpeita ja ongelmia sekä käyttäjien ja järjestelmien prosesseja, joita tukemaan sovellusta ollaan kehittämässä.
- Toiminnalliset vaatimukset tai käyttäjätarinat kuvaavat ne tiedot ja toiminnot, jotka sovelluksen tulee tarjota eri käyttäjäryhmille sekä muille ohjelmistoille ja laitteille.
- Tekniset vaatimukset kuvaavat mm. tekniikoihin, käyttöjärjestelmiin, muistiin, nopeuteen, laitteisiin ja sovelluksessa käytettäviin ohjelmiin sekä niiden versioihin liittyviä vaatimuksia.
- Rajoitteet rajaavat yleensä toiminnallisia ja teknisiä vaatimuksia jollain lailla. Niissä monesti kuvataan reunaehtoja tai miltä osin vaatimusten ei tarvitse toteutua.
- Käyttötapaukset kuvaavat yleensä loppukäyttäjän tavat käyttää sovellusta.
- Vaatimukset tulee priorisoida luokittelemalla ne muutamaan tärkeystasoon. Lisäksi kullekin vaatimukselle tulee kirjata sen toteutumista kuvaava tila.

Sovelluksen taustalla olevien prosessien kartoittamisen kautta yleensä löytyvät olennaisimmat vaatimukset ja rajoitteet. Taustojen ja kokonaisuuden kuvaaminen auttaa ymmärtämään asiayhteyksiä, joihin vaatimukset liittyvät. Vaatimusten priorisoinnin kautta projektissa kehitettävä sovelluksen versio tulee sisältämään tilaajan ja loppukäyttäjien kannalta vähintään kaikkein olennaisimmat tiedot ja toiminnot.

3.5 Sovellussuunnitelmat

Sovellussuunnitelmat kuvaavat, **miten sovellukseen toteutetaan** määritellyt vaatimukset. Ne käsittelevät ainakin osaa seuraavista asioista:

- Sovellukselle asetetut tavoitteet, toimintojen ja ominaisuuksien priorisointi sekä mahdolliset projektin ulkopuolelle rajattavat ominaisuudet tulee kirjata, ellei niitä ole mainittu vaatimusmäärittelyssä.
- Sovelluksen kokonaisrakenteen hahmottelu sisältää mm. olennaisimmat moduulit sekä yhteydet muihin sovelluksiin, tiedostoihin ja tietokantoihin (kuvien käyttö suositeltavaa).
- Rajapintojen kuvaus sisältää mm. ohjelmaosioiden lähtö- ja tulostiedot.
- Sovelluksen tietorakenteita ja toimintaa voidaan kuvata erilaisilla kaavioilla, kuten ER-tietokantakaavioilla sekä luokka-, sekvenssi- ja tilakaavioilla.
- Käyttöliittymän kuvaus sisältää mm. näkymäkartat näkymistä ja niiden keskinäisistä suhteista sekä näkymien rakenteet ja sisällöt (kuvien käyttö suositeltavaa).
- Käyttöliittymän ikkunat sekä niiden sisältämät tiedot ja toiminnot tulee suunnitella käytettäviksi sekä standardisoida yhtenäisiksi.
- Ohjelmakoodin kommentointitavat ja muuttujien nimeämiskäytännöt kuvaamalla lähdekoodista tulee yhtenäisempi eri tekijöiden kesken.

Tilaajan vaatimuksesta sovellukselle toteutetaan joissakin projekteissa muitakin määrittely- ja suunnitteludokumentteja, joiden sisältö on vähintään yksityiskohdiltaan tilaajakohtainen. **Arkkitehtuurisuunnitelma**ssa kuvataan ohjelmiston moduulit sekä niiden väliset vuorovaikutukset ja rajapinnat. **Luokkasuunnitelma**ssa esitellään sovelluksen sisältämien moduulien luokkarakenne luokkakaavioiden avulla. **Tietokantasuunni-**

telmassa kuvataan tietokannan taulut sekä niiden kentät ja suhteet. **Käyttöliittymäsuunnitelma**ssa kuvataan käyttöliittymän yleiset toiminnot, näyttöjen sisällöt kuvina sekä näyttöjen suhteet näyttökartan avulla.

3.6 Testaussuunnitelmat

Testaussuunnitelmien avulla saadaan testauksesta hallittu, järjestelmällinen ja toistettava. Projektissa kannattaa laatia järjestelmätestaukselle testaussuunnitelma kuvaten vähintään testausympäristöt ja suoritettavat testitapaukset. Testaussuunnitelmassa tulee mainita, mistä tarvittava testidata saadaan tai miten se luodaan. Erityisesti tulee huomioida, että ohjelmaosiot tulee testata ensin erillään, jonka jälkeen ne testataan yhdistettynä osaksi isompaa kokonaisuutta. Testaus- tai projektisuunnitelmassa tulee lisäksi mainita, ketkä minkäkin testaustehtävän suorittavat.

Testitapauksissa kuvataan suoritettavan yksittäisen "testin" askeleet kera tarvittavien tietojen ja toimintojen. Testitapauksia laadittaessa voi hyödyntää mm. vaatimuksia, käyttäjien kartoitettuja työprosesseja ja/tai aiempien ohjelmien syötteitä ja vasteita. Testitapauksiin saattaa olla järkevä sisällyttää myös mm. sovelluksen tukemat eri syöttö- ja tulostusformaatit, käyttöliittymissä vältettävien virheiden ennakointia sekä erilaisten erikoistapausten käsittelyä. Etenkin yksikkötestaukseen voi olla järkevää ohjelmoida testitapauksia, jolloin ne voidaan suorittaa säännöllisesti sovelluksen uusille versioille.

Kustakin testaussuunnitelman mukaisesti suoritetusta testauskerrasta tulee laatia erillinen **testausraportti**. Suoritetun testauskerran osalta tulisi ensin kuvata testattavan järjestelmän kokoonpano (laitteet, ohjelmistot ja niiden versiot), jotta testausympäristö voidaan pystyttää uudelleen samassa kokoonpanossa. Kunkin testitapauksen tulos sekä havaitut virheet ja tehdyt huomiot tulee kirjata testausraporttiin.

3.7 Projektiraportti

Projektiraportin tulee sisältää ainakin seuraavat tiedot:

- Tiivistelmä sisältää 2–5 lauseen mittaisen kuvauksen projektista, toteutetusta sovelluksesta ja raportin tarkoituksesta.
- Sisällysluettelo sisältää kaikki luvut ja liitteet.
- Johdanto johdattaa lukijan dokumentissa käsiteltävään asiaan kuvaten projektin ja dokumentin tarkoitusta.
- Käytetyt termit tulee selostaa.
- Tehtävän taustaa sekä projektia ja toteutettua sovellusta tulee esitellä lyhyesti.
- Kuinka alkuperäiset tavoitteet toteutuivat (toteutuma verrattuna suunnitelmaan)?
- Projektin resurssit sisältävät organisaation (ryhmän jäsenet, ohjaajat, tilaajan edustajat ja sidosryhmät) sekä projektin käyttöön annetut tilat, laitteet ja ohjelmat.
- Perehdytykset ja koulutukset tulee esitellä sekä arvioida niiden hyödyllisyyttä projektin tavoitteille ja jäsenille.
- Projektin käytänteitä ja hallintatapoja sekä toteutumista ja hyötyä tulee kuvata.
- Projektin tehtävät, niiden jakautuminen jäsenten kesken (työnjako tai vähintään vastuualueet) ja työtunnit tulee kuvata kera muutosten suunnitelmaan verrattuna.
- Projektin toteutunut prosessi ja aikataulu tulee esitellä ja verrata sitä suunniteltuun. Mistä muutokset aikataulussa johtuivat, ja kuinka suuria ne olivat ajallisesti?
- Työtuntien määrien kehittymistä viikoittain ja jakautumista tehtävittäin tulee kuvata graafeina. Niissä havaittavia muutoksia ja eroja tulee analysoida sanallisesti.
- Riskien toteutumista, ennakointia ja hallintaa tulee kuvata.

- Miten tehtävät selkiytyivät työstettäviksi?
- Mikä oli vaikeaa ja mitä opitte? Mitä tekisitte toisin, jos aloittaisitte alusta?
- Kunkin ryhmän jäsenen oppimaa ja kokemuksia projektista tulee kuvata.
- Yhteenveto sisältää mm. pohdintaa projektin ja sovelluksen onnistumisesta.
- Lähdeluettelo sisältää kaikki projektityössä käytetyt lähteet, kuten kirjat, artikkelit, WWW-sivut ja käyttöohjeet sekä projektin toteuttamat muut dokumentit.

Projektiraportissa tulee lisäksi analysoida projektin ja sen jäsenten toimintatapoja, ongelmia, vaikeuksia ja tehtyjä ratkaisuja sekä sitä, miten ne vaikuttivat projektin läpivientiin. Kyseisten asioiden pohtiminen on olennaista kunkin projektin jäsenen oppimisen kannalta.

3.8 Sovellusraportti

Sovellusraportissa (tai luokkadokumenteissa) tulee käsitellä ainakin seuraavia asioita:

- Toteutettua sovellusta tulee lyhyesti tarkastella **taustojen** kautta sekä sen **suhdetta suurempaan kokonaisuuteen**.
- **Käyttöliittymä** tulee esitellä ainakin yleisellä tasolla käyttöliittymäkuvien ja näyttökartan avulla. Monesti yksittäisten ikkunoiden tai sivujen tarkempi esittely kannattaa sijoittaa käyttöohjeeseen.
- **Sovelluksen kokonaisrakenne**tta tulee kuvata tekstin lisäksi myös kaavioina. Kuvauksen tulee kattaa ainakin ohjelmaosioiden keskinäiset suhteet sekä ohjelman yhteydet muihin sovelluksiin, tiedostoihin, tietokantoihin ja laitteisiin.
- **Ohjelmaosioiden toiminnan** osalta tulee yleisellä tasolla kuvata, mitä kukin ohjelmaosio tekee. Mitä algoritmeja ja ideoita niissä hyödynnetään?
- **Luokkiin** (tai aliohjelmiin) **jaon** osalta tulee esitellä, mitä eri luokkia kukin ohjelmaosio sisältää sekä mitä metodit (tai aliohjelmat) tekevät ja mitä attribuutteja (tai muuttujia) niissä käsitellään? Luokkiin välitettävät parametrit sekä lyhyet kuvaukset metodien toiminnasta tulee myös liittää.
- **Rajapintojen** osalta tulee kuvata, mitä syöttötietoja kukin ohjelmaosio tarvitsee ja millä formaatilla ne välitetään? Millaiset tiedot on kustakin osiosta saatavissa ulos, ja mitkä ovat kyseiset metodikutsut?
- Sovelluksessa ja luokissa käytettyjen oleellisimpien **tietorakenteiden** esittely (kuvien kera) tulee sisältää ainakin tärkeimpien tietokantataulujen, taulukoiden, tiedostojen ja tietueiden sekä globaalien muuttujien (vältä käyttöä) kuvaukset.
- Sovellukselle **asetettuja tavoitteita ja vaatimusten toteutumista** tulee analysoida. Toteutumattomien vaatimusten lisäksi tulee selkeästi kuvata myös toteutuneet vaatimukset sekä niiden muutokset ja syyt.
- Toteutusratkaisujen kehittymistä, tehtyjä muutoksia ja niiden syitä sekä valittujen ratkaisujen onnistumista tulee kuvata.
- **Toteutetut testaukset** ja niiden tulokset tulee esitellä ainakin tiivistetysti. Kustakin testauskerrasta kannattaa laatia erillinen testausraportti.
- Toteutetun sovelluksen sisältämät **virheet ja puutteelliset toteutusratkaisut** tulee kuvata. Miten ja miksi ohjelman saa kaatumaan?
- **Jatkokehitykseen ideat** ja ohjeet jatkokehittäjälle tulee kirjata.
- **Ylläpitäjän tiedot ja ohjeet** tulee liittää raporttiin tai erilliseksi ohjeeksi.

Sovelluksen rakenteen ja toiminnan kuvaamisen yhteydessä suositellaan kuvien käyttöä, jolloin sovellus on kokonaisuutena paremmin lukijan hahmotettavissa. **Ohjelman yksityiskohtien kuvaaminen** on yleensä järkevää sijoittaa ohjelmakoodiin kommentteina. Ohjelmointikielille löytyy lähdekoodin ja sen sisältämien kommenttien perusteella **luokkadokumentit** muodostavia ohjelmia (kuten Doxygen, Javadoc ja PyDoc). Suosituksena on kuvata sovelluksen kokonaisrakennetta ja sen perusteita sovellusraportissa sekä yksityiskohtia luokkadokumenteissa.

Yrityksissä sovellussuunnitelma ja -raportti ovat yleensä saman dokumentin eri versioita. Sovellusprojekteissa ne on eriytetty, jotta suunnitelman ja toteutuman vertailu olisi helpompaa.

3.9 Käyttö-, asennus- ja käännösohjeet

Selkeä käyttöohje ottaa huomioon eri käyttäjäryhmät, joille ohjelma on suunnattu. Käyttöohjeen käyttöliittymän kuvissa tulee olla järkevää dataa liittyen samaan kohteeseen ja käyttötilanteeseen, jolloin ohjetta läpikäyvän on helpompi ymmärtää näkymien yhteys. Älä unohda esimerkkejä ja varoituksia tilanteista, joissa ohjelma saattaa toimia epäloogisesti tai kaatua. Lisäksi tulee muistaa mainita, mihin sovelluksen versioon ohje liittyy.

Projektikansioon tulee sisällyttää **ohjeet myös sovelluksen ylläpitäjiä ja jatkokehittäjiä varten**. Dokumentteihin tulee sisällyttää **asennusohje tai asennusohjelma sekä käännösohje** (esim. makefile). Ohje kannattaa laatia ruokaohjeen tyyliseksi sisältäen sekä tarvittavat ohjelmistot ja laitteet että asennuksessa suoritettavat toimenpiteet. Ohjelmat tulee kääntää maksimivaroituksilla ja kaikki jäljelle jäävät varoitukset on kuvattava asennusohjeessa. Tällöin ohjelmaa myöhemmin kääntävä henkilö on tietoinen odotettavista varoituksista ainakin mainitussa sovelluskehitysympäristön versiossa.

3.10 Ohjelmalistaus

Ohjelmalistaukset kommentteineen liitetään projektikansioon CD:n ohella kaksipuolisesti paperille tulostettuina. **Varmistakaa viimeistään ennen listausten tulostamista, etteivät koodirivinne pituudet ylitä paperinleveyttä.**

Ohjelmalistauksessa sivut tulee olla numeroitu ja tiedoston nimi kunkin sivun yläreunassa. **Kunkin lähdekooditiedoston alkuun tulee sijoittaa sisällysluettelo (aliohjelmista), tekijöiden nimet ja päiväys sekä mahdollisesti myös versionumero ja lisenssin nimi.** Noudattamalla valitun luokkadokumenttigeneraattorin kommentointikäytänteitä voidaan lähdekoodista muodostaa luokkadokumentit säännöllisesti.

Lähdekooditiedoston alussa on järkevää listata oleellisimmat tehdyt lisäykset ja muutokset, ellei lähdekoodien hallintaan käytetä versiohallintaohjelmistoa. Tällöin virheiden jäljittäminen toimivaan ohjelmaan tehtyjen muutosten ja lisäysten jälkeen on helpompaa, koska virheet todennäköisesti löytyvät viimeksimuokatuista kohdista. Lisäksi versiohallintaan tai kunkin ohjelmatiedoston alkuun tulee tällä tavoin tallennettua ohjelmaosion kehityshistoriaa.

3.11 Ajankäyttöraportit

Kukin projektin jäsen pitää kirjaa projektiin käyttämistään työtunneista ja niiden sisällöstä. Luentoihin, väliesittelyihin sekä dokumenttien rakenteen ja kirjoitusasun muokkaukseen käytetyt työtunnit tulee kirjata oheiskurssille. Muut työtunnit kirjataan projektille.

Tehtyjä töitä toivotaan kuvattavan useammalla sanalla siten, että työkohteet ja kohdatut ongelmat tulevat selkeästi esiin. Tällä tavoin projektin eteneminen ja mahdolliset ongelmat tulee kirjattua ns. organisaatiomuistin kertymistä varten. **Työt tulee luokitella**, jolloin projektin lopussa on helpompi havaita kokonaistyömäärän jakautuminen mm. määrittelyyn, suunnitteluun, ohjelmointiin, dokumentointiin, testaukseen ja projektihallintaan sekä tutustumiseen, kokouksiin, esityksiin, luentoihin ja perehdytyksiin.

Ajankäyttöraportin yhteenveto on esitettävä ohjaajille ja tilaajan edustajille kunkin projektipalaverin tilakatsauksessa. Ajantasainen yhteenlaskettu tuntimäärä tulee olla nähtävissä jäsenittäin sekä mielellään tehtäväkokonaisuuksittain ja viikoittain graafeina. Palaverien tilaraporttia voi käyttää muistilistana sisältäen yhteenvedon suoritetuista ja suorittamattomista tehtävistä. Yksityiskohtaisemman tehtäväkohtaisen tilatiedon hallintaan voi käyttää jotain tehtävien hallinnan ohjelmistoa.

3.12 Kirjoittamisesta

Sovellusprojektien vastaavat ohjaajat kiinnittävät sovelluksen ja dokumenttien sisällön lisäksi huomioita myös niiden rakenteeseen, esitystapaan ja kieliasuun. **Selkeä ja sujuva esitystapa auttaa ja nopeuttaa dokumenttien tietosisällön tulkitsemista.** Samalla projektin jäsenet saavat kokemusta teknisten dokumenttien ja tieteellisten tutkielmien laatimisesta ja sisällöstä.

Sovellusprojektien yhteydessä kirjoitettavat dokumentit tulisi pääasiallisesti suunnata henkilöille, joiden tietämys aiheesta, sovelluksen toteutuksesta ja työkaluista on hieman vähäisempää kuin projektin jäsenten, ohjaajien ja tilaajan edustajien. Dokumenttia ei siis tulisi kirjoittaa tavalla, joka olettaa lukijan jo aiemmin tutustuneen kyseiseen aihealueeseen, sovellukseen tai projektiin.

Alaluvussa annetaan muutamia olennaisia neuvoja Sovellusprojekteihin liittyvien dokumenttien kirjoittamiseen. Kattavampi ohje opinnäytetöiden kirjoittamiseen [Santanen, 2000] on saatavilla tietotekniikan laitoksen WWW-sivuilta HTML-, pdf- ja ps-muodossa.

Word-tekstinkäsittelyohjelman käyttöön liittyviä vinkkejä löytyy mm. Petri Heinosen laatimista ohjeista [Heinonen, 2001] ja [Heinonen, 2002]. Lisäksi tietotekniikan laitoksen pro gradu -tutkielmien ohjeiden WWW-sivun kautta on saatavilla tutkielmien Word- ja LaTeX-pohjat, joita voi halutessaan hyödyntää myös Sovellusprojektien dokumenttien yhteydessä. **Dokumentteja laadittaessa kannattaa hyödyntää aiempien sovellusprojektien vastaavia dokumentteja.**

Sovellusprojektin dokumentit tulee kirjoittaa tilaajan kanssa sovittavalla tekstinkäsittelyohjelmalla. Kirjoittamisessa tulee pyrkiä mahdollisimman sujuvaan tyyliin ottamalla huomioon mm. tässä alaluvussa ja kirjoitusohjeessa [Santanen, 2000] mainitut neuvot sekä käsiteltävät aiempien projektien dokumentoinnissa havaitut kirjoitus- ja muotoiluvirheet.

Projektidokumenttien **kansisivulla** tulee mainita

- Sovellusprojektin nimi,
- tekijöiden nimet,
- kyseisen dokumentin nimi,
- päivämäärä ja versio sekä
- maininta Jyväskylän yliopistosta ja informaatioteknologian tiedekunnasta.

Useita dokumentteja läpikäyvä arvostaa lyhyttä **tiivistelmää** tekstin alussa, jolloin hän saa nopeasti käsityksen sen sisällöstä. Näin hänen kiinnostuksensa dokumenttiin saattaa herätä, ja hän perehtyy siihen tarkemmin. Tiivistelmän jälkeen kokenut lukija saattaa vielä käydä läpi **johdannon** dokumentin alusta ja **yhteenvedon** sen lopusta, joten niidenkin kirjoittamiseen tulee nähdä vaivaa. Dokumentin tulee sisältää **sisällysluettelo**.

Jokaisen luvun alussa olevan "**johdantokappaleen**" avulla lukija johdatetaan luvussa käsiteltäviin asioihin. Erityisesti luvun ja ensimmäisen alaluvun otsikoiden välissä tulee olla vähintään parin virkkeen johdantokappale.

Teksti ei saa koostua ainoastaan esiintulleita asioita sisältävistä listoista, joten ranskalaisten viivojen liiallista käyttöä tulee välttää. Monesti listojen käyttö kuitenkin selkeyttää ja tukee tekstiä sekä nopeuttaa tiedon hakemista ja kokonaisuuksien hahmottamista. Listojen, kuvien ja taulukoiden yhteydessä tulee käyttää "sidostekstiä", jolla lukija johdatetaan niissä käsiteltävään asiaan.

Sivut ja luvut tulee numeroida. Myös sisällysluettelon tulee sisältää luku- ja sivunumeroinnin. **Kuvat ja taulukot** tulee myös numeroida, jolloin niihin viittaaminen on helpompaa. Lukuihin, kuviin ja taulukoihin tulee viitata käsittelykohdassa niiden numeroilla. Kuva- ja taulukkotekstit tulee keskittää sivusuunnassa, ja ne tulee kirjoittaa normaalia tekstiä kapeammiksi.

Liian pitkiä lauserakenteita, tekstikappaleita, alalukuja ja lukuja tulee välttää, sillä ne hankaloittavat asian ymmärtämistä. Englanninkielisten termien käyttöä tulee välttää, jos vastaava suomenkielinen termi löytyy. Monesti kannattaa englanninkielinen vastine laittaa kursivoituna sulkuihin suomenkielisen termin (engl. *term*) perään.

Tärkeiden sanojen ja lauseiden esiintuomisessa kannattaa käyttää **normaalitekstistä eroavaa kirjasinta**, kuten **lihavointia**. *Kursivointia* käytetään pääasiassa suorissa lainauksissa, termien englannin- tai suomenkielisten vastineitten osoittamiseen sekä käyttöliittymien tekstien yhteydessä. Tekstin seassa olevat ohjelmakoodit, käskyt, parametrit, syötteet ja vasteet tulee kirjoittaa tasavälisellä mahdollisimman paljon `ohjelmakoodia muistuttavalla kirjasimella`. Mahdolliset matemaattiset kaavat kannattaa myös kirjoittaa erottuvalla *matemaattisella* kirjasintyylillä.

4 Projektiryhmien kokoontumiset

Sovellusprojektit kokoontuvat viikottaisten projektiorganisaation palaverien lisäksi oheiskurssien luennoille, kolmeen esittelytilaisuuteen sekä ryhmäkohtaisiin perehdytyksiin ja katselmointeihin.

4.1 Projektipalaverit

Projektin jäsenet tapaavat ohjaajat ja tilaajan edustajat säännöllisesti yleensä **kerran viikossa järjestettävässä projektipalaverissa.** Ryhmästä yhden opiskelijan tulee toimia **puheenjohtajana** vastaten esityslistan mukaisesta ja sujuvasta palaverin läpiviennistä. Toisen jäsenen tulee toimia **sihteerinä** kirjaten käsitellyt asiat ja tehdyt päätökset.

Muistiinpanojensa pohjalta sihteerin tulee laatia palaverista **pöytäkirja**, josta jäsenet, ohjaajat ja tilaajan edustajat saavat nopeasti käsityksen edellisessä palaverissa käsitellyistä ja sovituista asioista. Pöytäkirjat tulee laatia riittävän huolellisesti ja kattavasti, sillä mahdollisissa ristiriitatilanteissa sovitut asiat tulee pystyä tarkistamaan pöytäkirjasta. Toisaalta pöytäkirjaan kannattaa kirjata ainoastaan olennaisimmat huomiot, päätökset ja toimenpiteet karsien tarpeettomia yksityiskohtia.

Palavereissa yleisesti käsitellyt asiat ovat seuraavat:

- Esityslistan hyväksymistä ennen valitaan puheenjohtaja ja sihteeri.
- Palaverin laillisuus ja päätösvaltaisuus todetaan yleensä esityslistan toimitusajankohdan ja osapuolten läsnäolon perusteella.
- Edellisen palaverin pöytäkirja käydään läpi tiivistetysti sekä se hyväksytään sellaisenaan tai korjauksin. Puheenjohtajan riittää tuoda esille kustakin pöytäkirjan kohdasta ainoastaan olennaisimmat käsitellyt asiat ja päätökset.
- Viime palaverissa sovittujen toimenpiteiden toteutuminen tarkastetaan.
- Projektipäällikkö kuvaa projektin ja sovelluksen tilat.
- **Palaverin koollekutsumisen asiakokonaisuudet käydään läpi.**
- Vaihtoehdot, ratkaisut ja ehdotetut päätökset projektin ja sovelluksen osalta tulee tuoda esille ja saada niille hyväksyntä.
- Läsnäolijoille sovitut suoritettavat toimenpiteet varmistetaan.
- Sovitaan seuraavan palaverin ajankohdasta ja paikasta.

Projektin tilakatsauksessa kannattaa hyödyntää ajankäyttöraportista saatavia taulukoita ja graafeja. Siinä otetaan kantaa ainakin seuraaviin asioihin:

- Mitä kukin ryhmän jäsen on tehnyt edellisen palaverin jälkeen?
- Mitkä ohjelmaosista, dokumenteista ja projektin muista tuloksista ovat valmiina, kesken tai aloittamatta?
- Mitä seuraavaan palaveriin mennessä projektiryhmä aikoo tehdä?
- Miten kunkin jäsenten työtunnit poikkeavat suunnitellusta?
- Miten tulosten ja tehtävien tilanne sekä aikataulu poikkeavat suunnitellusta?
- Mitkä seikat estävät tai hidastavat projektin läpivientiä tai tulosten kehittämistä?
- Mitä päätöksiä ja toimenpiteitä tarvitaan projektin etenemiseen?

Omaa työtään, kohdattuja haasteita ja ratkaisuja sekä vaihtoehtoja ja valintoja tulee rohkeasti tuoda esille. Ratkaisujen, virheiden ja ongelmien ”pimittämisestä” voi olla seurauksena työmäärän kasvu johtuen vääristä valinnoista tai ratkaisuista.

Asioiden hahmotteleminen paperille tai valkotaululle voi olla hyvä tapa saada aivot askaroimaan tehtävän tai pulman parissa. Etenkin siitä on hyötyä työn etenemistä mahdollisesti haittaavien ongelmien ja oleellisten kysymysten keksimisessä. Mieleen nousevat **ns. tyhmät kysymykset kannattaa aina esittää**, sillä epäselviksi jääneet asiat saattavat johtaa projektia väärään suuntaan ja viivästyttää sitä aivan turhaan. Kysymyksiin saa yleensä välittömästi vastauksen tilaajan edustajilta tai ohjaajilta.

4.2 Projektipalaverien esityslistat ja pöytäkirjat

Jotta palaveri sujuisi hallitusti ja nopeasti sekä tärkeimmät käsiteltävät asiat olisivat jo palaverin alussa kaikkien osallistujien tiedossa, projektiryhmän tulee laatia **palaverille esityslista**. Siinä tulisi mielellään olla osoitteet tai linkit palaverissa käsiteltäviin dokumentteihin, ohjelmaosioihin ja materiaaleihin. Esityslista tulee lähettää sähköpostitse kaikille projektiorganisaatioon kuuluville ja muillekin kyseiseen palaveriin osallistuville vähintään vuorokautta ennen palaveria.

Käsiteltävät asiat kannattaa esityslistassa järjestää siten, että ensin käsitellään projektiin liittyvät asiat, sovelluksen vaatimukset ja rajaukset sekä viimeiseksi sovelluksen toteutustekniset yksityiskohdat. Jos palaverissa ei tulla käsittelemään tilaajaa koskevia asioita, asiasta tulee mainita esityslistassa, jolloin he voivat halutessaan jäädä pois palaverista. Kaikki tilaajien edustajat eivät nimittäin halua, ehdi tai pysty puuttumaan sovelluksen toteutusratkaisuihin, jolloin he ovat kiitollisia mahdollisuudesta käyttää kyseinen aika muihin tehtäviinsä.

Käsitellyt asiat ja tehdyt päätökset kirjataan pöytäkirjoihin. Pöytäkirja tulee jakaa useammaksi sopivan lyhyeksi kohdaksi siten, että kohtien otsikoiden perusteella saa nopeasti tiivistetyn käsityksen palaverissa käsitellyistä asioista. Pöytäkirjojen kohdat kannattaa jakaa melko lyhyiksi tekstikappaleiksi, jolloin niissä käsitelty asia on helpommin hahmotettavissa. Jos jokin kohta tuntuu liian pitkältä, sen voi tarvittaessa jakaa loogisiin alakohtiin.

Tehdyt päätökset tulee selkeästi korostaa käsitellyistä asioista. Kaikkiin kohtiin ei välttämättä päätöstä liity, sillä monesta asiasta palavereissa vain keskustellaan tai tiedotetaan ilman selkeää päätöstä. Päätöksiä kannattaa korostaa esimerkiksi kirjaamalla ne kunkin kohdan loppuun kera *Päätökset*-sanan.

Kullekin palaveriin osallistuneelle sovitut toimenpiteet tulee kirjata listana pöytäkirjan loppuun omaksi numeroimattomaksi kohdaksi. Myös tilaajan edustajien sekä vastaavan ja teknisten ohjaajien toimenpiteet tulee kirjata esille. Em. sijoituksella ne tulevat kaikille palaveriin osallistuneille selkeämmin esille, ja niiden suoritus voidaan nopeasti tarkistaa kunkin palaverin aluksi. Toimenpiteitä on nimittäin huomattavasti hankalampi löytää yksittäisistä kohdista. Toimenpiteitä ei tule ”niputtaa”, eikä niitä kannata kirjata toistamiseen kohtien päätöksiksi.

Pöytäkirjan lopussa kannattaa listata palaverissa käsitellyt ja jaetut materiaalit sekä mahdolliset liitteet. Niihin tulee viitata käsitellyssä pöytäkirjan kohdassa.

Pöytäkirjaan kirjattaviin asioihin kuuluvat aina läsnäolijat, puheenjohtaja, sihteeri (pöytäkirjan laatija), paikka ja aika sekä seuraavan palaverin paikka ja aika (katso luvun 4.1 lista). Pöytäkirjoja kirjoitettaessa mieleentulleet kysymykset kannattaa kirjata erilliseen dokumenttiin, jolloin ne ovat valmiina seuraavaa palaveria varten.

Pöytäkirjan tulee olla riittävän tarkka kuitenkin siten, että käsitelty asia kuvataan yksityiskohtaisesti ainoastaan olennaisimpien asioitten osalta. Käsiteltyjen asioiden ja päätösten tiivistäminen sekä olennaisten asioiden esilletuonti on hyödyllinen opittava taito. Ensimmäisten ja viimeisten palaverien pöytäkirjat ovat yleensä pidempiä ja yksityiskohtaisempia johtuen käsiteltyjen asioitten määrästä ja useista tehdyistä päätöksistä.

Pöytäkirja on helpointa kirjoittaa heti palaverin jälkeen, jolloin käsitellyt asiat ovat hyvin vielä mielessä. Puheenjohtajan (tai projektipäällikön) tarkastamisen jälkeen sihteeri lähettää pöytäkirjan projektin sähköpostilistan kautta kaikille projektiorganisaation jäsenille ja muille palaveriin osallistuneille välittömästi niiden valmistuttua. Tällöin tilaajan edustajat ja ohjaajat voivat tarkastaa kirjatut asiat ja päätökset tuoreeltaan. Lisäksi he pääsevät ennakolta valmistelemaan vastauksia mahdollisiin pöytäkirjassa esitettyihin kysymyksiin.

Pöytäkirjaa ei mielellään tulisi toimittaa liitetiedostona, vaan ASCII-muodossa osana sähköpostia tai ilmoittamalla sen URL-osoite. Kaikilla ei välttämättä ole tarvittavaa tiedostomuotoa tukevaa ohjelmaa, tai tiedoston avaaminen vaatii ylimääräisiä toimenpiteitä. Lisäksi WWW-sivuille pöytäkirjasta on tarvittaessa päivitettävissä muokattu versio.

4.3 Ensimmäiset projektipalaverit

Kokoukset ja palaverit lienevät suurelle osalle projektin jäsenistä uusi kokemus ja lisäksi niissä tulee yleensä runsaasti uutta asiaa. Lienee siis syytä mainita muutama huomioitava asia.

Projektin alussa tulee sille keksiä **akronyymi** eli lyhyt ja ytimekäs nimi. Akronyymin mukainen sähköpostilista luodaan projektiryhmän, ohjaajien ja tilaajan edustajien väliseen tiedotukseen. **Projektin tiedotusvastuu siirtyy ensimmäisen projektipalaverin yhteydessä opiskelijaryhmälle.**

Ryhmän jäsenten kannattaa laatia ensimmäiset palaveripöytäkirjat yhdessä, sillä kaikille projektin jäsenille on todennäköisesti pienen tietotulvan alta jäänyt mieleen eri asioita. Kaikkien projektipalaveriin osallistuvien nimet, työnimikkeet ja yhteystiedot tulee kirjata pöytäkirjaan. Jos jonkun henkilön nimi ei tule esille tai unohtuu, sitä tulee kysyä.

Ensimmäisessä palaverissa tilaajat kuvaavat aiheen lisäksi yleensä **projektin aihealuetta ja taustoja, tilaajan kokonaistavoitteita sekä käyttäjien tarpeita ja ongelmia sekä niihin liittyviä prosesseja**. Kyseisiä asioita vaaditaan esiteltävän myös projektisuunnitelmassa ja vaatimusmäärittelyssä. Jos taustat on jo kirjattu pöytäkirjoihin, ne on helppo siirtää osaksi muita dokumentteja.

Muita huomioitavia asioita ensimmäisissä palavereissa (ja siten myös pöytäkirjoissa) ovat tilaajien, ohjaajien ja ryhmän jäsenten käyttämät **"uudet" käsitteet**. Epäselviksi jääneiden termien merkitys tulee selvittää viimeistään seuraavassa palaverissa. Ne tulee kirjata sekä pöytäkirjaan että projektisuunnitelmaan.

Ensimmäisissä projektipalavereissä **sovellukselle asetettavien vaatimusten** lisäksi monesti kuvataan muitakin mahdollisia sovellukseen toivottavia ominaisuuksia. Tällöin tilaajan edustajilta, mutta toisinaan myös ohjaajilta ja ryhmän jäseniltä, voi sadella ideoita runsaasti, joten niistä joudutaan **rajaamaan projektiin sisällytettävät asiat**. Projektin tavoitteet tuleekin (ainakin alustavasti) selkeästi sopia kunkin palaverin lopussa sekä **kirjata tarkasti pöytäkirjaan sovitut mukaanliitettävät ja ulkopuolelle rajattavat sovelluksen ominaisuudet ja projektin tehtävät**.

Ensimmäisessä projektipalaverissa käsitellään yleensä ainakin seuraavat asiat:

1. kokouksen avaus (ml. puheenjohtajan ja sihteerin valinta),
2. laillisuus ja päätösvaltaisuus,
3. esityslistan hyväksyminen,
4. osallistujien esittäytyminen,
5. tilaajan organisaation ja toiminnan esittely,
6. aihe, taustoja ja kokonaistavoitteita,
7. käyttäjien tarpeita ja projektin tavoitteita,
8. mahdollisia toteutusratkaisuja,
9. aikataulu ja prosessimalli,
10. sopimukset, lisenssit ja projektimaksu,

11. projektissa noudatettavat käytänteet,
12. projektin nimi ja tilat sekä jäsenten yhteystiedot,
13. projektipäällikön ja varapäällikön valinta,
14. muut esille tulevat asiat,
15. läsnäolijoille sovitut toimenpiteet,
16. seuraavan palaverin aika ja paikka sekä
17. kokouksen päättäminen.

Ensimmäisen palaverin esityslistan voi siten aika pitkälti laatia yo. listan perusteella.

Käytänteet ovat yhteisesti noudatettavia projektiryhmältä vaadittuja tai ryhmän itselleen määrittämiä toimintatapoja. Olennaisimmat sovittavat käytänteet liittyvät seuraaviin asioihin:

- ohjelmointikieli, ohjelmointityökalut, palvelinohjelmistot ja valmiskomponentit,
- laadittavat dokumentit, dokumentoinnin kieli ja työkalut, luokkadokumentaatio,
- tulosten toimittaminen projektiorganisaatiolle (sovitaan mm. WWW- tai Trac-wikisivuston perustamisesta ja versiohallintaohjelmiston hyödyntämisestä),
- tulosten hyväksyntä ja katselmointi (erillinen katselmointitilaisuus kannattaa järjestää vähintään lähdekoodeille ainakin kahteen kertaan),
- tiedotuksen kohteet, tavat ja säännöllisyys,
- lähdekoodin nimeämis- ja kommentointikäytänteet,
- hakemistorakenne ja tiedostojen nimeäminen,
- versiohallinta ja -numerointi sekä
- projektin muut läpiviennin käytänteet (kuten tilakatsaukset viikoittain).

4.4 Viimeiset projektipalaverit

Viimeisissä projektipalavereissa käsitellään projektin päättämiseen, tulosten hyväksymiseen ja toimittamiseen sekä sovelluksen mahdolliseen asentamiseen, ylläpitoon ja jatkokehittämiseen liittyviä asioita. **Vähintään kuukautta ennen projektin oletettua päättymistä projektiryhmän tulee alkaa esittelemään valmistuneiden, keskenolevien ja aloittamattomien tehtävien tilaa sekä niihin tarvittavien työtuntien arvioita.** Tällöin projektin tilaajan edustajat ja ohjaajat pystyvät tarvittaessa rajaamaan vaatimuksia toteutettavan sovelluksen, dokumenttien ja muiden tulosten osalta.

Oleellisia viimeisissä palavereissa käsiteltäviä asioita ovat mm. seuraavat:

- Mitä sovelluksen ominaisuuksia sovitaan jatkokehitykseen?
- Sovelluksen ja dokumentoinnin valmistumisesta ja hyväksymisestä sekä niiden aikataulusta sovitaan.
- Lähdekoodin tarkastuksesta sovitaan teknisen ohjaajan ja mahdollisesti myös tilaajan kanssa.
- Mitä vielä tarvitaan projektin hyväksymiseen?
- Koodien, ajokelpoisten ohjelmien ja dokumenttien salattavista osista sovitaan.
- Salaisten projektien tapauksessa sovitaan mahdollisesta verkkolevyn, WWW-sivuston ja sähköpostiarkiston tyhjentämisestä. Ryhmän verkkolevylle tallentamat materiaalit säilyvät yleensä noin puoli vuotta, sekä muut tallennuspaikat "ikuisesti".
- Toimitetaanko projektin tulokset (kuten ohjelmat, dokumentit ja muu materiaali) tilaajalle CD:n ohella projektikansiona?
- Tuleeko sovelluksen ja dokumenttien vanhempia versioita säilyttää ja liittää myös projektidokumentaatioon?
- Projektilausuntojen, -todistusten ja -laskun toimittamisaikataulusta sovitaan.

- Laitteet, ohjelmistot, kirjat ja muu materiaali palautetaan hankkijalleen tai asiasta sovitaan.
- Projektin tuloksista tiedotetaan tarvittaessa sovitusti sidosryhmille.

Projektin päättämiseen liittyviä asioita on käsitelty myös luvussa 5.4.

4.5 Esittelytilaisuudet

Projektiryhmät kokoontuvat loppuesittelyn lisäksi kahteen yhteiseen väliesittelyyn. Esittelytilaisuuksissa kullakin ryhmällä on korkeintaan 20 minuuttia aikaa kuvata aihettaan ja sovellustaan. Esittelymateriaali tulee etukäteen hyväksyttää tilaajan edustajilla erityisesti salaisten projektien tapauksessa.

Väliesittelyissä ohjaajat, puheviestinnän opettaja ja projektiryhmien jäsenet arvioivat muiden ryhmien esitystä, aiheita, sovellusten käyttöliittymiä sekä jonkin verran myös ohjelmien toteutusta. **Esittelyistä tulee laatia muistiot** lähinnä oman ryhmän esityksen sisällön, sujumisen ja kysymysten sekä saadun suullisen ja kirjallisen palautteen pohjalta.

Ensimmäisessä väliesittelyssä esitellään laadittuja suunnitelmia ja arvioidaan niiden toteuttamismahdollisuuksia. Ensimmäisessä esittelyssä pyritään keskustelun kautta saamaan uusia ideoita ryhmille. Ryhmälle saattaa palautteiden pohjalta syntyä ideoita mm. sovelluksen toteutusratkaisuista tai tavoista kehittää omaa työtään.

Toinen väliesittely on pari viikkoa ennen loppuesittelyä, jolloin sovelluksiin voidaan vielä tehdä pieniä korjauksia ja muutoksia. Toisessa esittelyssä muiden projektien jäsenten on melko helppo kommentoida ainakin sovellusten käyttöliittymiä.

Projektiryhmän jäsenet perustelevat ja kuvaavat työtään esittelyissä, joten he joutuvat tarkastelemaan toimintaansa muilta näkökannoilta. Perustelut ja kuvaukset vaikuttavat myös projektin dokumentointiin, sillä esittelyissä ryhmä joutuu muokkaamaan sanottavansa riittävän yksinkertaiseen ja ymmärrettävään muotoon.

Esiintymisen, keskustelun ja perustelemisen voidaan katsoa olevan taitoja, joita yliopisto-opiskelun tulee kannustaa ja parantaa. Molempien väliesittelyjen kautta esiintyminen yleisölle tulee ryhmän jäsenille tutummaksi, jolloin loppuesittely ja muutkin myöhemmät esitelmät sujunevat rutiinilla luontevammin. Esittelyissä huomioitavia asioita on käsitelty enemmän luvussa 4.6.

4.6 Projektin loppuesittely

Sovellusprojektin loppuesittely on tärkeä osa projektin onnistumista. Se on projektin mainostilaisuus, jossa yleisönä on muiden projektiryhmien, tilaajien edustajien ja ohjaajien lisäksi mm. laitoksen henkilökuntaa ja opiskelijoita.

Loppuesittelyyn kuuluvat

- **etukäteen suunniteltu ja jäsennelty** esitys,
- projektin taustojen ja tavoitteiden kuvaus,
- sovelluksen kokonaisrakenteen, toteutusratkaisujen ja työkalujen kuvaus,
- sovelluksen toiminnan esittely käytännössä, sekä
- projektin läpiviennin toteutuman ja kokemusten tiivis kuvaus.

Kullekin projektille varataan **korkeintaan 20 minuuttia** aikaa projektinsa esittelyyn, jottei yleisö pitkästyisi usean projektin esittelyssä. Kyseiset 20 minuuttia kuluvat yllättävän nopeasti, joten **esitys tulee suunnitella ja harjoitella etukäteen**. Älkää ylittäkö teille varattua aikaa, sillä tuolloin muiden ryhmien esittelyille jää vastaavasti vähemmän aikaa. Kunkin ryhmän esittelyn jälkeen yleisöllä on kymmenisen minuuttia aikaa esittää kysymyksiä ja kommentteja, sekä tilaajan edustajat ja ohjaajat arvioivat kehitettyä sovellusta ja projektin läpivientiä.

Loppuesittelyssä tulee kiinnittää huomiota ainakin seuraaviin seikkoihin:

- Ryhmän jäsenten tulee osallistua tasapuolisesti esittelyyn jakamalla se sopiviin kokonaisuuksiin keskenään. Tällöin kuulijat saavat käsityksen yhtenäisestä ryhmästä, jonka kaikki jäsenet ovat osallistuneet tasapuolisesti työhön.
- Englanninkielisiä ja yleisölle outoja aihealueen ja tietotekniikan termejä tulee välttää tai niiden merkitys tulee selkeästi selittää.
- Ryhmän jäsenten tulee esitellä itsensä, tilaajan edustajat ja ohjaajat.
- Ilman projektin taustojen lyhyttä esittelyä kuulijat eivät saa aihealueesta ja siten tilaajan kokonaistavoitteista ja käyttäjien tarpeista riittävää kokonaiskäsitystä.
- Ohjelmointiympäristö työkaluineen tulee esitellä.
- Sovelluksen rakenne ja toteutus tulee kuvata olennaisimmilta osin. Liian pikkutarkka kuvaus aiheuttaa pitkästyneisyyttä yleisön joukossa. Jos joku kuulijoista on kiinnostunut yksityiskohdista, hänellä on mahdollisuus esittää asiaa koskeva kysymys tai ottaa ryhmään yhteyttä esityksen jälkeen.
- Sovelluksen toimintaa tulee esitellä ”juonellisesti” (käytöllä tähdätään johonkin tavoitteeseen) ainakin oleellisimpien tietojen ja toimintojen osalta. **Esittelyn tulee edetä rauhallisesti** selittäen, mitä kussakin vaiheessa tehdään. Normaalilla sovelluksen käyttönopeudella valkokankaalla näkyy vain sovelluksen ikkunoiden ja valikoiden vilkuntaa. Yleisön tulisi siis pysyä esityksessä mukana.
- Projektin läpivientiä sekä jäsenten kokemuksia ja oppimista tulee kuvata lyhyesti.
- Loppuesittelyn jälkeen esitettäviin kysymyksiin kuuluvat perinteisesti seuraavat: mikä oli vaikeaa, mitä opitte ja mitä tekisitte toisin. Jos ette puutu näihin kysymyksiin esittelyssänne, niihin kannattaa ainakin valmistautua pohtimalla niitä etukäteen.
- Esityssalissa olevien mikrofonien käyttö on suositeltavaa etenkin, jos esittelijä ei ole tottunut käyttämään ääntään isossa tilassa. Huonosti kuuluvan loppuesittelyn anti jää auttamattomasti alhaiseksi, joten sillä on alentava vaikutus projektin arvosanaan.

Projektiryhmän tulee varmistaa sovelluksen toimivuus esittelysalin koneessa ennen esittelyn alkua (mielellään jo edellisenä iltana). Toimimattoman ohjelman esitteleminen yleisölle on kokemusten mukaan melko nolo tilanne. Käyttäkää esittelyssä mieluummin vanhempaa toimivaksi todettua versiota kuin viimeisintä testaamatonta versiota. Tällöin on vähemmän ongelmia odotettavissa, vaikka uudessa versiossa olisikin huomattavasti enemmän toimintoja. Viime hetkellä toteutetut näyttävät osiot eivät siis mitenkään korvaa sovelluksen toimimattomuutta.

5 Projektin arvostelu ja päättäminen

Sovellusprojekti voidaan katsoa hyväksytyksi ja päättyneeksi, kun ohjaajat ja tilaajan edustajat ovat hyväksyneet sekä toteutetun sovelluksen että dokumentit. Käytännössä projektin jäsenet voivat katsoa projektin päättyneeksi saadessaan arvosanat.

5.1 Arvosteluperusteet

Vastaavien ohjaajien arvostelussaan huomioimat asiat on esitetty muistilistana osoitteesta `http://www.mit.jyu.fi/palvelut/sovellusprojektit/ohjaajat/projektiarviointi.pdf` löytyvästä pdf-dokumentista. Tiivistäen Sovellusprojektin arvosteluun vaikuttavat

- alkuperäisten tavoitteiden toteutuminen,
- sovelluksen toteutusratkaisut ja laatu,
- projektin suunnittelu, läpivienti, hallinta ja johtaminen,
- poikkeamien ja muutosten hallinnan onnistuminen,
- tiedonkulku ryhmän sisällä sekä ohjaajien ja tilaajan edustajien suuntaan,
- ryhmän toiminta ja yhtenäisyys,
- taidot ongelmatilanteiden ratkaisemisessa ja ristiriitojen käsittelyssä,
- oppimiskyky ja innostuneisuus,
- ryhmän jäsenten aktiivisuus, oma-aloitteisuus ja sitoutuneisuus,
- oman työn sekä mahdollisten ongelmien ja virheiden esilletuonti ja arviointi,
- työnjaon ja tarvittaessa sen uudelleenjaon onnistuminen,
- projektin jäsenten osallistuminen erilaisiin tehtäväkokonaisuuksiin (kuten määrittely, suunnittelu, toteutus, testaus ja dokumentointi),
- dokumenttien sisältö ja kirjoitusasu,
- projektin loppuesittely,
- testattavan version toimittaminen riittävän ajoissa tilaajalle ja ohjaajille sekä
- projektin valmistuminen suunnitellussa aikataulussa.

Ainoana arvosteluperusteena ei siis ole pelkästään sovelluksen toteutus (kuten kurssien harjoitustöissä), **vaan vastaava ohjaaja kiinnittää erityistä huomiota projektin läpivientiin ja valmistumiseen suunnitellussa aikataulussa sekä ryhmän toimintaan.** Vastaava ohjaaja ottaa arvostelussaan huomioon tilaajan edustajien ja teknisen ohjaajan mielipiteet.

5.2 Arvosanojen ja laajuuden määräytyminen

Sovellusprojekteissa käytetään seuraavia **karkeita arvostelutasoja** (asteikko 1–5):

1-taso: Sovellus toimii ja toteuttaa asetetut tavoitteet ”rimaa hipoen”. Dokumentointi ja loppuesittely ovat tyydyttäviä. Projektin läpiviennissä on ollut huomautettavaa. Tiedonkulussa ryhmän sisällä tai ulospäin tilaajalle ja ohjaajille on ollut puutteita. Projektin päättyminen on viivästynyt suunnitellusta.

3-taso: Sovellus toimii hyvin sekä työ on dokumentoitu ja esitelty selkeästi. ”On siis tehty kaikki, mitä on käsketty, muttei sen enempää”. Projektin läpiviennissä ei ole ollut huomautettavaa. Tiedonkulussa ryhmän sisällä sekä ulospäin tilaajalle ja ohjaajille ei ole ollut huomauttamista. Projekti on valmistunut likimain ajallaan.

5-taso: Edellisen kohdan lisäksi mukana on omia innovaatioita, asenne työhön on innostunut tyyliin "vaikeudet on tehty voitettaviksi", ja lopputulos ylittää tavoitteet. Projekti on läpiviety hallitusti, ja ryhmätyö on sujunut hyvin. Tiedonkulku ryhmän sisällä sekä ulospäin tilaajalle ja ohjaajille on toiminut hyvin. Projekti on valmistunut sovitussa aikataulussa.

Projektille annetaan **ryhmäkohtainen arvosana**, joka kuvaa ryhmän toimintaa, projektin läpivientiä ja tuloksia kokonaisuutena. Tämän lisäksi kullekin projektin jäsenelle annetaan **henkilökohtainen arvosana**, joka kuvaa ryhmäkohtaisen arvosanan lisäksi kyseisen projektin jäsenen henkilökohtaista panosta projektiin. Panoksella ei tarkoiteta pelkästään projektin eteen tehtyjä työtunteja, vaan myös kyseisen henkilön aktiivista osallistumista projektiin, tehtävien vaativuutta ja oppimiskykyä projektin aikana. **Opintosuoritusrekisteriin merkitään projektin jäsenen henkilökohtainen arvosana.**

Sovellusprojekteiksi ehdotettujen aiheiden vaativuutta ja työmäärää on etukäteen hankala arvioida johtuen aihealueiden ja työkalujen runsaasta kirjosta sekä projektin jäsenten, ohjaajien ja tilaajan edustajien tietotaitojen eroavaisuuksista. Tästä johtuen Sovellusprojektien laajuuden arvioinnissa on käytössä **10–15 opintopisteen vaihteluväli**. On huomattava, etteivät tehdyt työtunnit suoraan määritä annettavia opintopisteitä, vaan niitä tarkastellaan mm. työn vaativuuteen rinnastettuina. **Projektiryhmän jäsenet saavat saman opintopistemäärän** "laiskimman" jäsenen mukaan.

5.3 Projektilausunnot, todistukset ja itsearviointi

Sovellusprojektin tilaajan edustaja ja vastaava ohjaaja laativat projektista **lausunnot**, jotka liitetään projektikansioon. Lisäksi tilaajan edustaja tai vastaava ohjaaja kirjoittaa pyynnöstä projektin jäsenille **projektitodistukset**.

Kunkin ryhmän jäsenen tulee toimittaa viimeistään lopullisen dokumentoinnin palautuksen yhteydessä vastaavalle ohjaajalle projektistaan ns. **itsearviointi**. Siihen tulee liittää omat arviot arvosanoista (1–5) sekä koko projektiryhmälle että itselleen. Myös parin virkkeen perustelut arvosanoille tulee kirjata. Tätä kautta ryhmän jäsenet joutuvat arvioimaan omaa työtään, kuten yleisesti nykyisin työelämässäkin joutuu tekemään.

Projektin jäsenten toivotaan antavan kyseisessä arvioinnissa arvosanat perusteluineen myös vastaavan ohjaajan, teknisten ohjaajien, tilaajan edustajien ja ATK-tuen toiminnalle. Lisäksi projektien luennoista ja perehdytyksistä toivotaan palautetta viimeistään itsearvioinnissa.

Itsearviointiin tulee kirjata myös tieto siitä, saako itsearvioinnin liittää projektikansioon vai ei. Jos kyseinen maininta uupuu, vastaava ohjaaja tulkitsee sen luvaksi sijoittaa itsearviointi projektikansioon julkisesti nähtäville.

Vastaava ohjaaja avaa projektin jäsenten toimittamat itsearvionnit vasta, kun hän on antanut arvosanat projektille ja sen jäsenille. Kyseisellä kaksinkertaisella arvioinnilla vastaava ohjaaja pystyy arvioimaan arvostelunsa oikeellisuutta ja oikeudenmukaisuutta.

5.4 Projektin päättyessä muistettavaa

Sovellusprojektien päättyessä on yleensä melkoisesti muistettavaa ja lisäksi kiire ansaitulle lomalle tai seuraaviin haasteisiin. Oleellisimmat projektin päättämiseen liittyvät käytännön työt tulevat hoidettua tukeutuen seuraavaan muistilistaan:

- Sovellus, lähdekoodit, dokumentit ja muut projektin tulokset tulee hyväksyttää sekä tilaajan edustajilla että ohjaajilla.
- Tilaajalta tulee selvittää, missä muodossa ja millä tallennusvälineellä projektin tulokset (ajettava ohjelma ja lähdekooditiedostot sekä dokumentit ja muu materiaali) tulee toimittaa.
- Projektin tulokset tulee toimittaa vastaavalle ohjaajalle kahtena CD:nä sekä myös paperille tulostettuina koottuina kansioksi. Toimitus koskee ainoastaan niitä projektiryhmän toteuttamia tuloksia, jotka tilaajan kanssa on sovittu julkisiksi.
- Ryhmän tulee ottaa varmuuskopiot projektimikrojen kovalevyillä olevista tärkeiksi katsomistaan tiedostoista pitäen mielessään salassapidon osalta sovitut asiat.
- Kun projektin käyttöön annetetut mikrot voidaan uudelleenasentaa seuraavalle käyttäjälle, asiasta tulee ilmoittaa projektin ja mikron nimen mainiten vastaavan ohjaajan lisäksi ATK-lähituelle.
- Tilaajan projektin käyttöön antamat laitteet ja materiaali tulee palauttaa tilaajalle tai sopia niiden jäämisestä projektin jäsenten ja laitoksen haltuun.
- Projektikirjat tulee palauttaa vastaavalle ohjaajalle sekä lainatut projektikansiot projektihyllyyn.
- Kunkin ryhmän jäsenen tulee palauttaa itsearviointinsa vastaavalle ohjaajalle.
- Projektikäyttöön saadut avaimet tulee palauttaa vahtimestareille.

Jos projekti viivästyy sovitusta, projektille osoitetun tilan ja laitteiden mahdollisesta käyttöoikeudesta projektin loppuunsaattamiseksi tulee sopia vastaavan ohjaajan kanssa.

5.5 Projektin päätyttyä

Sovellusprojektin päätyttyäkin projektin jäsenille voidaan katsoa jäävän vastuuta toteutetusta sovelluksesta. Mitään automaattista sitoutuneisuutta sovelluksen ylläpitoon tai jatkokehitykseen ei tietenkään vaadita.

Käytännössä Sovellusprojektien yhteydessä ei yleensä ehditä testata toteutettua sovellusta kovinkaan kattavasti. Tällöin **mahdollinen jatkokehitys** voidaan toteuttaa muiden opintojaksojen tai/ja palkallisen työn puitteissa. Projekteilla on siis ollut hyvä (tai huono) tapa jatkua jossain muodossa myös projektin jälkeen, mutta tällöin projektin jäsenten sitoutuneisuudesta jatkokehitykseen sovitaan aina erikseen.

Jotta projektien jäsenet tuntisivat projektin todella olevan mennyttä elämää, projektiryhmien suositellaan järjestävän yhteisen **päättötilaisuuden** projektien päätyttyä. Kyseisessä saunaillassa jäsenet voivat hyvän ruuan ja juoman kera tarkastella taaksejäänyttä aikaa. Informaatioteknologian tiedekunta mielellään maksaa ainakin osan tilaisuuden saunavuokrasta, ravintolakuluista ja hankitusta pikkupurtavasta.

6 Yhteenveto

Ohjeessa annetaan neuvoja Jyväskylän yliopiston tietotekniikan syventäviin opintoihin sisältyvien Sovellusprojektien menestykselliseen läpiviemiseen. Ohjeet kattavat projektin eri vaiheiden ja kokoontumisten kuvauksien lisäksi neuvoja sovelluksen toteuttamiseen sekä erilaisten projektissa laadittavien dokumenttien sisältöön liittyen. Ohjeiden laadinnassa on huomioitu aiemmista projekteista saatuja kokemuksia.

Sovellusprojekteihin liittyvistä asioista voi kysyä lisätietoa projektien vastaavilta ohjaajilta. Kritiikki, kiitokset ja kommentit sekä positiiviset ja negatiiviset kokemukset otetaan mielellään vastaan, jolloin niitä voidaan hyödyntää projektien kehittämisessä.

Sovellusprojektien aihe-ehdotuksia voi esittää vastaaville ohjaajille milloin tahansa. Syyslukukaudeksi ehdotettavien aiheiden tulee kuitenkin olla heidän tiedossaan elokuun loppuun mennessä ja kevätlukukaudeksi ehdotettavien vuoden loppuun mennessä. Projektit alkavat syksyn osalta syyskuun puolessa välissä ja keväällä helmikuun alussa.

Lopuksi haluan kiittää Kaisa Miettistä ja Marko Mäkelää vuoden 1995 projektiohjeesta, josta tämä ohje on vuosien varrella jalostunut. Lisäksi Tapani Tarvainen, Kari Kärkkäinen, Vesa Korhonen ja Ville Isomöttönen ansaitsevat kiitokset erinomaisista ohjetta kehittäneistä palautteista.

Hauskoja ja antoisia hetkiä projektityön parissa!

7 Lisämateriaalia

Haikala Ilkka ja Märijärvi Jukka, "Ohjelmistotuotanto", Suomen ATK-kustannus, 2000.

Heinonen Petri, *Lyhyt kirjoittamisopas teknisestä näkökulmasta*, saatavilla HTML-muodossa <URL: `http://appro.mit.jyu.fi/doc/kirjoitusopas/`>, Jyväskylän yliopisto, Tietotekniikan laitos, 23.2.2001.

Heinonen Petri, *Tekstinkäsittely*, saatavilla HTML-muodossa <URL: `http://appro.mit.jyu.fi/doc/tekstinkasittely/`>, Jyväskylän yliopisto, Tietotekniikan laitos, 5.2.2002.

Karlsson Åke ja Marttala Anders, "Projektikirja - Onnistuneen projektin toteuttaminen", Kauppakaari, 2001.

Korpela Jukka, "Tekijänoikeus: vastauksia usein esitettyihin kysymyksiin", saatavilla <URL: `http://www.cs.tut.fi/~jkorpela/tekoik/tekoik.html`>, Tampereen teknillinen yliopisto, tietotekniikan osasto, 15.4.2013.

Pressman Roger S., "Software Engineering: A Practitioner's Approach", McGraw-Hill, 2001.

Ruuska Kai, "Projekti hallintaan", Suomen ATK-kustannus, Jyväskylä, 1999.

Saarinen Mikael, "Tunne älysi – älyä tuntevasi : opas oman ja työyhteisön tunneälyn kehittämiseen", WSOY, 2001.

Santanen Jukka-Pekka, "Prosessit, projektin hallinta ja ryhmätyö", luentomateriaali, Jyväskylän yliopisto, tietotekniikan laitos, 2016.

Santanen Jukka-Pekka, "Opinnäytteiden kirjoittaminen, lyhyt oppimäärä", saatavilla HTML-muodossa <URL: `http://www.mit.jyu.fi/palvelut/sovellusprojektit/kirjoittamisesta.html`>, Jyväskylän yliopisto, tietotekniikan laitos, 23.8.2000.

Santanen Jukka-Pekka, "Vaitiolosopimusmalli", saatavilla HTML-muodossa <URL: `http://www.mit.jyu.fi/palvelut/sovellusprojektit/vaitiolo.html`>, Jyväskylän yliopisto, tietotekniikan laitos, 3.2.2006.

Santanen Jukka-Pekka, "Projektisopimusmalli", saatavilla HTML-muodossa <URL: `http://www.mit.jyu.fi/palvelut/sovellusprojektit/sopimus.html`>, Jyväskylän yliopisto, tietotekniikan laitos, 27.11.2008.

Sommerville Ian, "Software Engineering", 6th Edition, Addison-Wesley, 2000.

Tekijänoikeuksia ja aineettomia oikeuksia käsittelevää materiaalia löytyy myös URL-osoitteen `http://www.mit.jyu.fi/santanen/ohjeita.html#oikeudet` kautta.