

Projektin hallinta ja johtaminen

Hilkka Heikkilä
Jukka-Pekka Santanen

Jyväskylän Yliopisto
Informaatioteknologian tiedekunta

Hilkka Heikkilä
Syksy 2004



Luennon sisältö

13.30 - 15.00 (Hilkka)

- Odotukset ja toiveet sovellusprojekteilta
- Projekti vs. prosessi
- Prosessimallit
- Projektin hallinta
 - Suunnittelu
 - Seuranta
 - Päättäminen

15.00 - 15.15

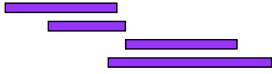
- Kahvitauko

15.15 - n. 16.45 (Jukka-Pekka)

- Projektin johtaminen

16.45 - n. 17.30 (Hilkka)

- Projektityöskentely käytännössä

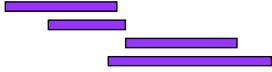


Odotukset

Toiveet

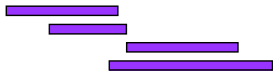
Pelot

Jännityksen aiheet



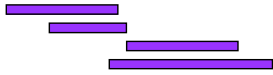
*Tee hyvää eli paha,
niin eressäs sen löydät.*

- sananlasku Nurmijärveltä



Ohjelmistoprojekti

- Projekti = prosessin instanssi
- Projekti on määritelty ja ajallisesti sidottu tehtävä, jolla on tietty päämäärä. Lisäksi
 - projektilla on merkitys
 - elinkaari
 - riippuvuuksia
 - projekti on yksilöllinen
 - rajallinen määrä resursseja



Ohjelmistoprosessi

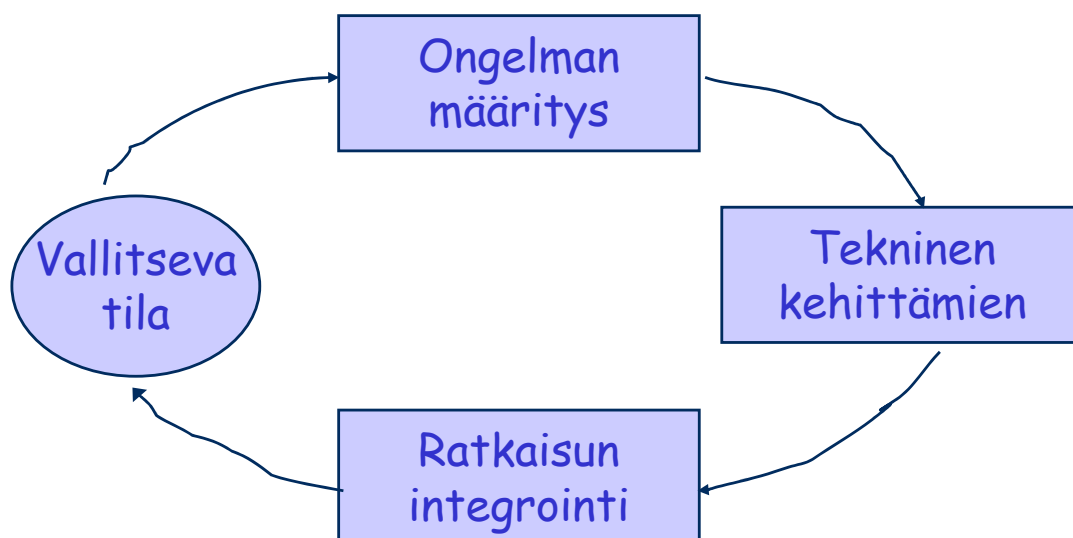
- Prosessi määrittelee yleisesti
 - Kuka tekee
 - Mitä tekee
 - Koska tekee
 - Kuinka saavuttaa tavoitteet
- Ohjelmistotuotantoprosessi
 - Ohjelmiston rakentamisen vaiheet ja niiden tulokset
 - Ohjelmiston elinkaaren määrittely
 - Yleisrakenne sille miten ohjelmisto tuotetaan

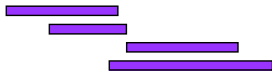
Miksi ohjelmistoprojekteja?

- Ohjelmistotuotannossa on aloitettu puhumaan projekteista ja prosesseista samoihin aikoihin kuin ohjelmistotuotannosta ja sen kriisistä 1968
- Kriisi: Ongelmia ohjelmistojen tuotannossa
 - aikataulut eivät pitäneet
 - kustannukset nousivat
 - laajojen ohjelmistojen tuottaminen vaikeaa
 - ohjelmistojen laatu oli huono
 - mallia insinööritieteistä
 - projektimuotoinen ohjelmistokehitys
- Mutta ihan samanlainen projektinhallinta ei sovellu ohjelmistokehitykseen kuin muihin insinööritieteisiin
 - ohjelmistojen abstraktista luonne
 - ei ole valmiita komponentteja
 - kahta samanlaista projektia ei ole

hallittavuutta
seurattavuutta
läpinäkyvyyttä

Prosessi ongelmaratkaisuna





Ohjelmistoprosessimalli

- Sisältää ohjelmistotuotantoprosessin keskeiset osa-alueet
- Kuvaa osaprosessien suoritusjärjestyksen
- Ei konkreettisia yksityiskohtaisia toimintasääntöjä vaan pelkkä yleisrakenne
- Malli tuottaa dokumentaation
- Malli on yleinen -> sovellettavissa kaikkialle, mutta mukautettava organisaation tarpeisiin



Prosessimallin vaihejako

- Määrittelee prosessin tärkeimmät osavaiheet
- Määrittelee osavaiheiden suoritusjärjestyksen
- Määrittelee osavaiheiden keskinäiset vaikutussuhteet
- Määrittelee osavaiheiden keskinäisiin rajapintoihin liittyvät toimenpiteet kuten dokumentointi, tarkastukset yms.
- Jokaisen vaiheen alku ja loppu on selvästi havaittavissa
- Prosessiin liittyy kaikille osavaiheille yhteisiä tukitoimenpiteitä

Tukitoiminnot / sateenvarjotehtävät

- dokumentointi (lähinnä asiakasdokumentointi)
- laadunvarmistus
- tuotteen hallinta
- projektinhallinta
- formaalit tekniset tarkastukset
- konfiguraation hallinta- kokoonpanon hallinta
- uudelleenkäytön hallinta
- mittaaminen
- riskien hallinta

Ohjelmistotuotannon osatehtävät

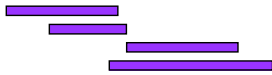
Määrittely

Suunnittelu

Ohjelmointi

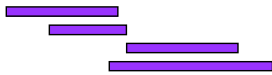
Testaus

Käyttö ja Ylläpito



Prosessimallityypit 1/2

- Rakentava
 - peräkkäisiä työvaiheita, jokaisen jälkeen syntyy jokin vaihetuote
 - määrittely-suunnittelu- toteutus
 - seuraava vaihetta ei voi aloittaa ennen edellisen päättymistä
- Kasvattava (inkrementaalinen)
 - tehdään ensin toimiva ydinjärjestelmä, jota kasvatetaan useassa iteraatiossa kunnes saavutetaan riittävä järjestelmä



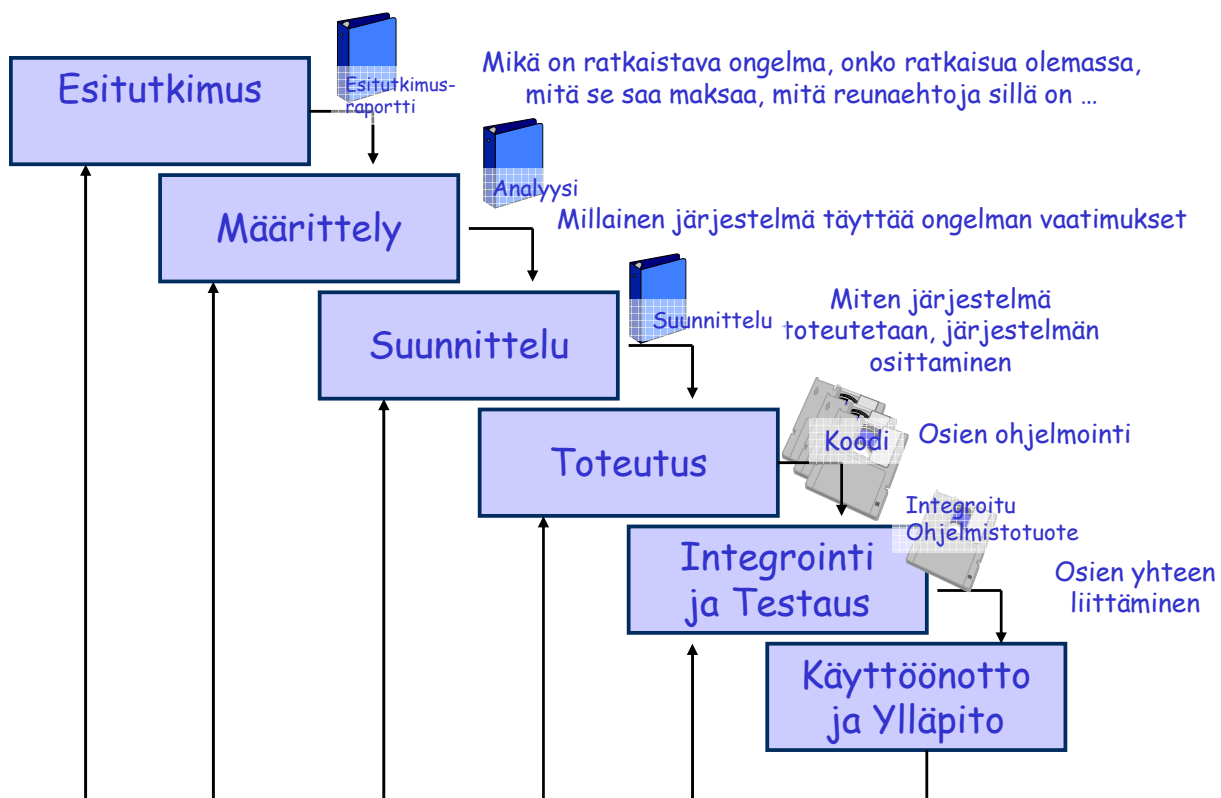
Prosessimallityypit 2/2

- Kehittävä (iteratiivinen)
 - tehdään ensin koko järjestelmä vajavaisin toiminnoin
 - järjestelmää kehitetään useissa iteraatioissa paremmaksi
- Kokoava
 - järjestelmä kootaan uudelleenkäytettävistä osista
 - tarvitaan myös komponenttiprosessi, uudelleenkäyttöprosessi

Prosessimallit 1/2

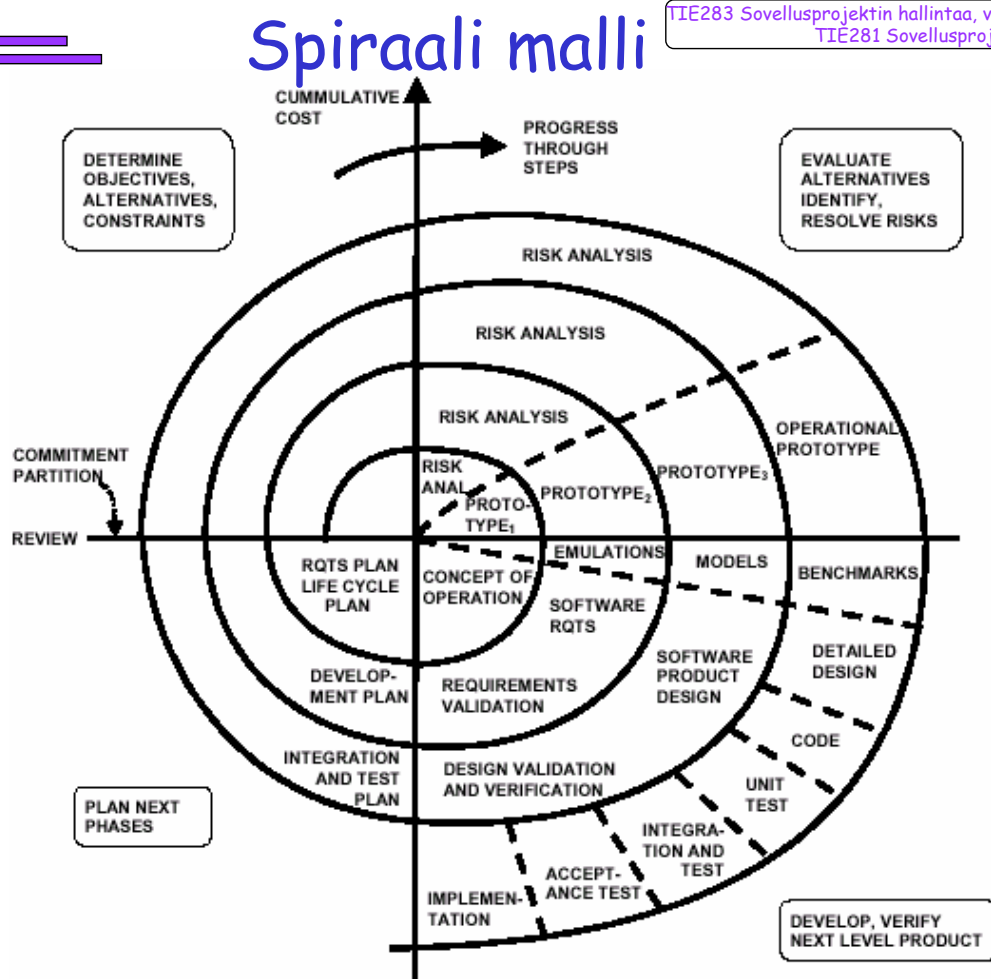
- Koodaa ja korjaa (Code and fix)
- Vesiputousmalli (Waterfall model)
 - Vesiputousmalli
 - Vesiputousmalli, jossa on osaprojekteja
 - Vesiputousmalli, jossa vaiheet osittain menevät päällekkäin
 - Vesiputousmalli, jossa riskiä on vähennetty

Vesiputousmalli



Prosessimallit 2/2

- Inkrementaalinen malli (Incremental models)
 - Spiraalimalli
 - Evolutiivinen protoilu
 - Vaiheittainen toimitus, inkrementaalinen toteutus
 - Aikataulu suunnittelu (Design-to-schedule)
 - Evolutiivinen toimitus (Evolutionary delivery)
 - Käyttötapa ohjautuva inkrementaalinen kehitys



Analysis

Architectural
Design

Vaiheittainen toteutus eli inkrementaalinen toteutus

Stage 1: Detailed
design, coding,
testing & delivery

SW Product
1.0

Stage 2:
Detailed design,
coding, testing &
delivery

SW Product
1.1

Stage 3: Detailed
design, coding,
testing & delivery

SW Product
1.2

Stage 4: Detailed
design, coding,
testing & delivery

SW Product
1.3

Analysis

Architectural
Design

Aikataulusuunnittelu

High priority

Stage 1: Detailed
design, coding,
testing & delivery

SW Product
1.0

High priority

Stage 2:
Detailed design,
coding, testing &
delivery

SW Product
1.1

Medium priority

Stage 3: Detailed
design, coding,
testing & delivery

SW Product
1.2
the shipped release

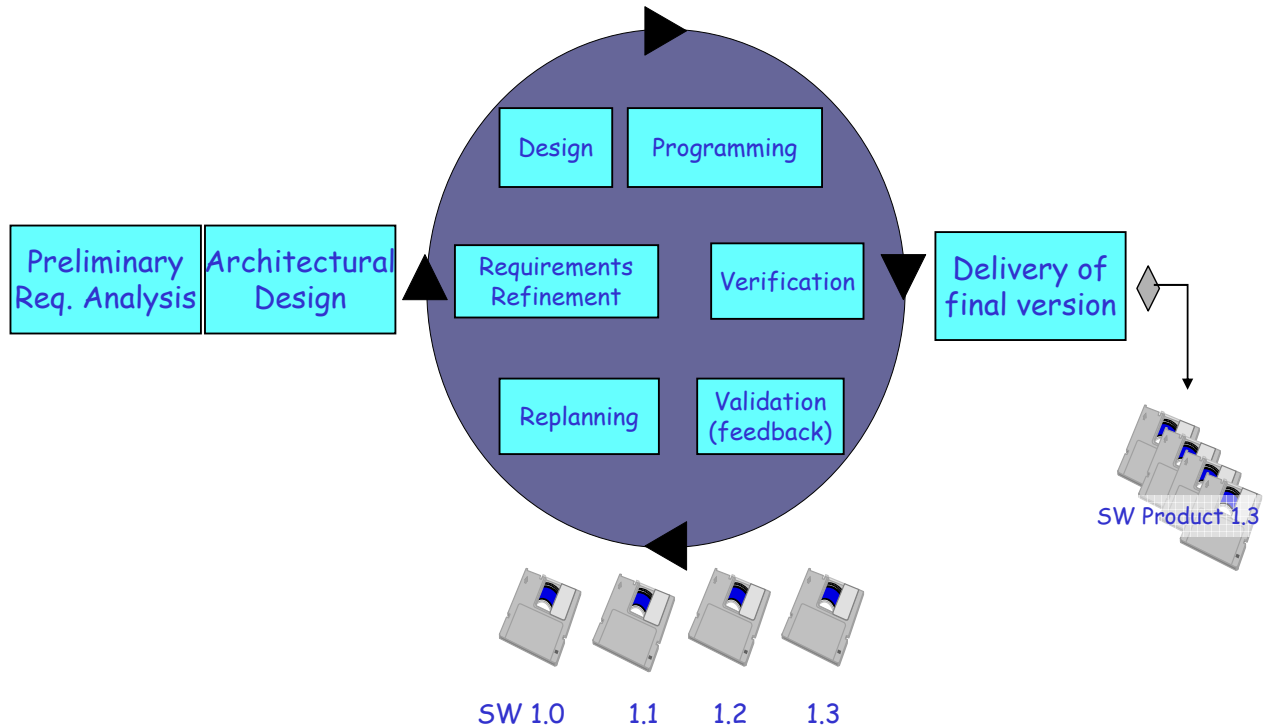
Run out of time here

Low priority

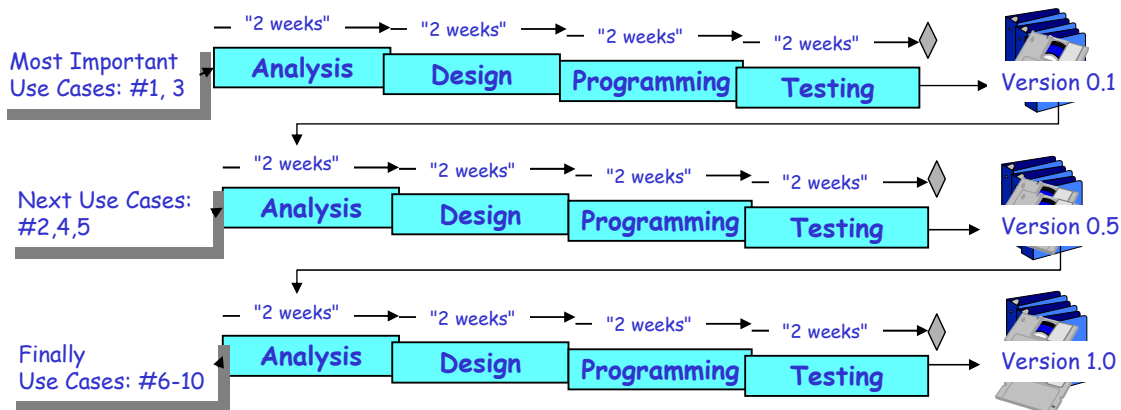
Stage 4: Detailed
design, coding,
testing & delivery

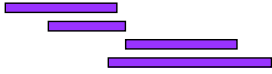
SW Product
1.3

Evolutiivinen toimitus



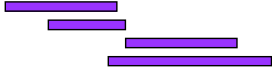
Käyttötapaohjattu inkrementaalinen tuotekehitys





Muita prosessimalleja

- Komponenttiprosessimalli
- Rinnakkaisen kehitystyön malli
- Formaalit menetelmien malli
- Ketterät/ kevyet mallit
- "Cleanroom" menetelmä malli



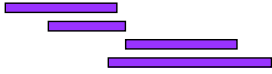
eXtreme Programming

Extreme Programming (XP)

- Neljä arvoa
 - Kommunikointi
 - kommunikoinnin puute usein syynä ongelmiin
 - käytännöt ohjaavat suunnittelijat, asiakkaat ja päälliköt kommunikoimaan
 - Yksinkertaisuus
 - Yksinkertaisin tapa toimia
 - Palaute
 - konkreettista palautetta järjestelmästä
 - päivittäin: yksikkötestaus, tehtävien edistyminen
 - kuukausittain: hyväksyntätestit, aikataulu, tuotettava järjestelmä
 - Rohkeus
 - muuttaa systeemiä
 - heittää koodia pois
 - pariohjelmointi

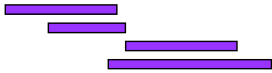
Extreme Programming (XP)

- Perusperiaatteet
 - Nopea palaute (henkilökohtainen, testaus)
 - kriittistä oppimisen kannalta
 - Oleta yksinkertaiset ratkaisut
 - Inkrementaalinen muutos
 - pienten muutosten sarja
 - Hyväksyä muutos
 - Laadukas työ
- Toissijaisia periaatteita
 - Pienet alkuperäiset sijoitukset
 - Konkreettiset kokeet
 - Ota vastuuta
 - Opettaa oppimaan
 - Avoin, rehellinen kommunikointi
 - Paikalliset sovellukset
 - Rehellinen mittaaminen



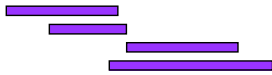
Oletukset

- Ohjelmistot suunnitellaan pienissä (4-12 henkeä) tiimeissä samassa tilassa
- Ohjelmistoja ylläpidetään ja laajennetaan määräämättömän ajan
- Uusia, ennustamattomia vaatimuksia tulee kun ohjelmistoja käytetään
- Sama tiimi kehittää ohjelmistoa, ei siirretä



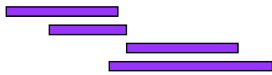
XP käytännöt

- Metafora
 - vertauskuvallinen kuvaus järjestelmästä
 - korvaa tavanomaisen arkkitehtuurin
- Suunnittelupeli
 - jokaisen iteraatiokierroksen (1-3 viikkoa) aloitus
 - valitaan iteraatiossa toteutettavat käyttäjätarinat ja arvioidaan toteuttamisen aikataulu
- Paikalla oleva asiakas
 - asiakas koko ajan mukava
 - asiakas suunnittelee ja suorittaa hyväksymistestit



XP käytännöt

- Yksinkertainen suunnittelu
 - mahdollisimman yksinkertaiset tekniset ratkaisut
 - toteutetaan vain suunnittelupelissä valittuja käyttäjätarinoita, ei mitään ylimääräistä
- Koodin yhteisomistus
 - kuka tahansa voi ja pitää muuttaa mitä tahansa osaa koodista
 - muuttajan vastuulla uusien testien laatiminen
- Pariohjelmointi
 - kasi kehittäjä tuottaa koodia saman tietokoneen ääressä
 - XP:n kiistanalaisin käytäntö



XP käytännöt

- Koodin parantelu - refaktorointi
 - rakenteellisesti huonolaatuisen koodin korjaaminen selkeämmäksi ja hyvien tapojen mukaiseksi
 - koodin toiminnallisuus ei saa muuttua
- Ohjelmointi standardi
 - yhteisten ohjelmoinnin tyylisääntöjen noudattaminen
 - mahdollistaa yhteisomistuksen ja kommunikoinnin
- Testaus
 - yksikkö- ja hyväksymistestit, suunnitellaan ennen koodausta
 - automatisoidut testit -> iterointi, regressiotestaus

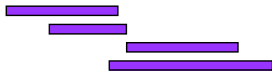
XP käytännöt

- Pienet julkaisut
 - kasvattava ohjelmankehitys pienin inkrementein
- Jatkuva integrointi
 - uusi, yksikkötestit läpäissyt koodi liitetään heti ohjelmistoon
 - aina käytettävissä viimeisin versio, mutta jatkuvan regressiotestauksen kustannuksella
- 40 tunnin työviikko
 - ylitöitä vain poikkeustilanteessa
 - tavoitteena yhteisen työtehon pitäminen korkeana

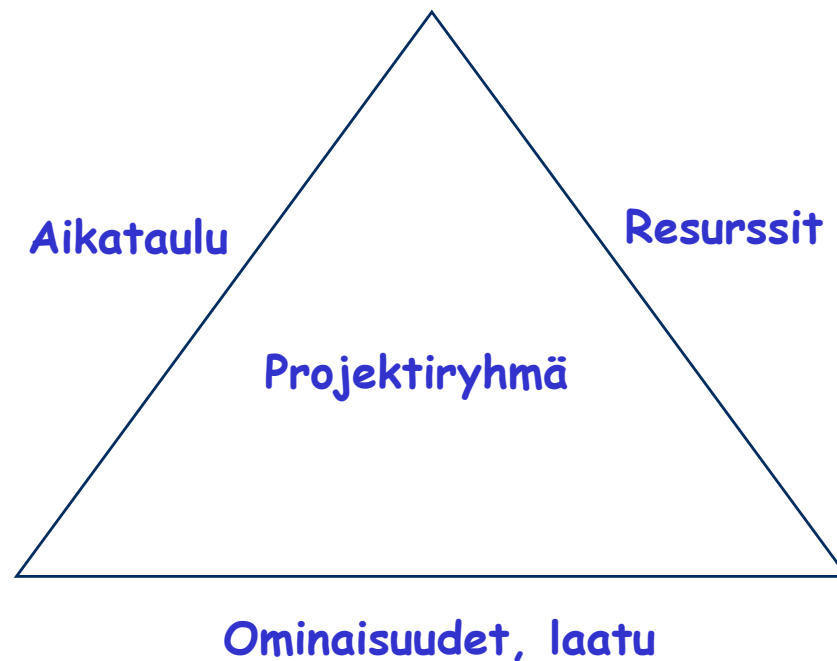
Prosessimallin valinta

Vaikuttavia tekijöitä

- organisaation kypsyys
- teknologiaympäristö
- tuotteen tyyppi
- asiakkaan kanssa tehtävän sopimuksen tyyli
- projektiryhmän taidot
- tuotteen arvioitu elinaika
- aikataulu
- laatuvaatimukset
- projektin luonne
- sovelluksen luonne
- tarvittava kontrolli



Projektin dimensiot



Erilaisia projekteja

- Esitutkimusprojekti
 - vaihtoehtojen kartoittaminen
 - alustava määrittely ja tarveanalyysi
 - kannattavuuden ja riskien arviointi
- Määrittelyprojekti
 - lopputuloksen määrittely
 - alustava tekninen suunnittelu
 - riskien arviointi
- Toteutusprojekti
- Käyttöönottoprojekti
- Ylläpitoprojekti
- Koulutusprojekti
- Kehitysprojekti
- Tuotekehitysprojekti
- ...

Ohjelmistoprojektin hallinta

Asiakkaan huomioiminen

Tarvittava laitteisto

Laadun hallinta

Riskien hallinta

Mittaaminen

Kustannusten
arviointi

Tekniset suuntaviivat



Projektin aikataulu

Tuotteen hallinta

Projektin resursointi

Projektin seuranta

Ihmisten johtaminen

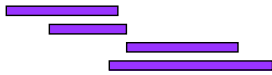
Kommunikointi

Jyväskylän Yliopisto
Informaatioteknologian tiedekunta

Hiikka Heikkilä
Syksy 2004

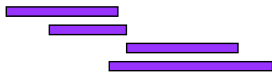
Ohjelmistoprojektin neljä keskeistä käsitettä (4 P's)

- Ihmiset (People)
 - tärkein osa-alue onnistuneessa projektissa
- Tuote (Product)
 - rakennettava ohjelmisto
- Prosessi (Process)
 - raamit toiminnalle ja työlle
- Projekti (Project)
 - kaikki työ, joka vaaditaan että saadaan tuote tehtyä



Ihmiset

- TÄRKEIN
- Ohjelmistoprojektin osapuolet
 - Ylempi johto
 - Projektipäällikkö
 - Ohjelmistokehittäjät - projektiryhmä
 - ohjelmoijat, suunnittelija, testaajat, tuotteenhallinta, myynti, dokumentointi jne.
 - Asiakkaat
 - Loppukäyttäjät



Ohjelmistotiimi

- Ohjelmistotiimiä rakentaessa on huomioitava
 - ongelman vaikeus
 - ohjelmiston koko
 - tiimin elinkaari
 - ongelman modularisointi
 - ohjelman laatu- ja luotettavuusvaatimukset
 - toimituspäivän ehdottomuus
 - projektin kommunikoinnin tarve (asiakas - tiimi)
 - jäsenten kompetenssien vastaavuus tarpeeseen

Prosessi

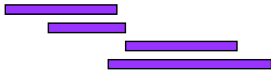
- Valitaan projektille sopiva prosessimalli
- Yksityiskohtaiset tehtävät määritellään projektin alkuvaiheessa valitun projektimallin mukaisesti
- Projektipäällikön perusteltava valintansa ja mietittävä soveltuvuutta
 - asiakkaan
 - tuotteen
 - projektiympäristön kannalta

Projektinhallinnan pääperiaatteet

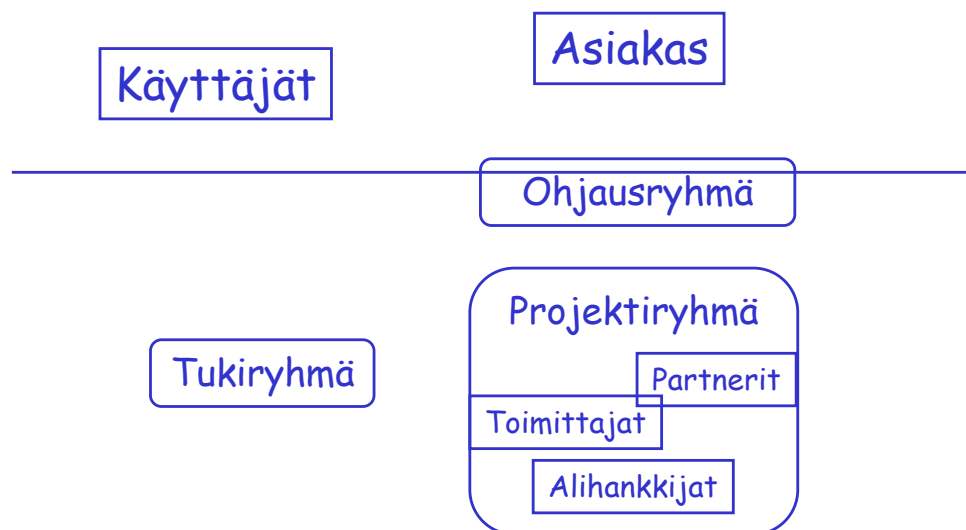
- Miksi järjestelmä kehitetään?
- Mitä tehdään?
- Kuka tekee?
- Milloin tekee?
- Kuka on vastuussa toiminnallisuudesta?
- Mihin organisaatiossa vastuuhenkilöt ovat sijoittuneet?
- Miten tehtävät hoidetaan teknisessä ja hallinnollisessa mielessä?
- Miten kauan kutakin resurssia tarvitaan (ihmiset, ohjelmistot, työkalut, tietokanta jne.)?
- Miten riskeihin valmistaudutaan?



Ohjelmistoprojektin suunnittelu



Projektiorganisaatio



Ohjausryhmä

- Asettaa projektin ajalliset, tekniset ja kustannukselliset tavoitteet
- Nimeää projektipäällikön
- Hyväksyminen projektisuunnitelman
- Antaa projektille sen tarvitsemat resurssit
- Tekee projektin kannalta keskeiset päätökset
- Projektin seuranta
- Muutosten hyväksyminen
- Hyväksyy projektin tulokset
- Päättää projektin lopettamisesta

Projektipäällikön vastuualueet 1/2

Laatii projektisuunnitelman

- projektin tavoitteet ja yleissuunnitelman tavoitteiden saavuttamiseksi
- projektin kustannusarvio
- projektin työvaiheet
- tarkistuspisteet
- toimitettavat tuotokset
- aikatauluarviot
- työntekijöin valinta ja arviointi - jokainen työntekijä omia erikoistaitojaan vastaaviin tehtäviin

Projektipäällikön vastuualueet 2/2

- Käynnistää projektiryhmän työskentelyn ja ohjaa ryhmää
- Johtaa projektin toimeenpanoa, tehtävien antoa ja valvoo työn edistymistä
 - Päivittää projektisuunnitelman ja aikataulua
 - Projektin seuranta ja tarkastukset (aikataulun, tuotteen laadun)
 - Raportointi
- Varustaa projektiryhmän tarvittavilla tiedoilla ja koulutuksella
- Laatii projektiloppuraportin ja suorittaa projektin päättämisen

Projektiryhmäläisen tehtävät

- Minkä tyyppisiin projekteihin perustyyppit sopivat?
- tehtävät
 - laadukas työ kaikissa vaiheissa
 - osallistuminen projektisuunnitelman laadintaan
 - raportointi
 - työn edistyminen
 - riskit
 - ongelmat
 - poikkeamat
 - dokumentoida työnsä tulokset
 - noudattaa päätettyjä teknisiä standardeja
 - kehittää omaa ammattitaitoaan ja työmenetelmiä

Ohjelmistoprojektin suunnittelu

- Jotta projektia voi ohjata, tarvitaan kehys, jonka mukaan projekti etenee. Tehdään ennen varsinaisia projektin työvaiheita
- Tarkoituksena on tehdä käytännöllinen suunnitelma, jotta voidaan hallita, seurata ja valvoa monimutkaista teknistä projektia
-> Lopputuote tulee tehdyksi laadukkaasti ajallaan.

Suunnittelun vaiheet

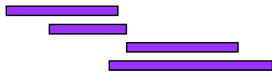
- Rajausta: ymmärtää ongelma ja mitä pitää tehdä, riippuvuudet
- Arviointi: paljonko työtä, paljonko aikaa?
- Riskit: Mitä voi mennä vikaan, kuinka se voidaan välttää, mitä asialle voi tehdä
- Aikataulu: miten resurssit varataan ajansuhteen, mitkä ovat etapit
- Hallintasuunnitelma: kuinka hallitsemme laatua? kuinka hallitsemme muutosta

Toiminnallinen jakaminen

- Tarkennetaan kunnes saavutetaan mielekäs tehtävä- tai työkokonaisuus, jonka voi
 - suunnitella
 - budjetoida
 - aikatauluttaa
 - seurata
 - hallita
- Tunnistetaan tietotarpeet ja nimetään vastuuhenkilöt, joka katselmoi tuotokset
- Yhdistetään työ- / tehtäväkokonaisuuksia
- Tehdään projektisuunnitelma ja budjetti
- Aikataulutetaan tehtävät
- Tunnistetaan ongelmakohdat ja tarkennetaan suunnitelmaa

Kustannusten arviointi

- projektin soveltamisala täytyy olla tarkasti määritelty
- työtehtäviin tai/ja toiminnallisuuksiin pilkkominen jos tarpeellista
- historialliset mittarit auttavat, muista luotettavuus
- ainakin kahta erilaista tekniikkaa tulisi käyttää
- välttä yli-optimistisuutta, oleta vaikeuksia
- mukauta arvioita ihmisten mukaan - heillä on suuri vaikutus
- muista että epävarmuus on luontaista



Kustannusten arviointi

- arviointitekniikat
 - asiantuntija-arviot
 - menneet (samantapaiset) projektikokemukset
 - perinteiset arviointitekniikat
 - tehtäväjako ja työmäärän arviointi
 - kokoarviointi (LOC/FP)
 - työkalut
 - Erilaiset lähestymistavat
 - Top - Down
 - Bottom - Up
 - Iteratiivinen (neuvotteleva)



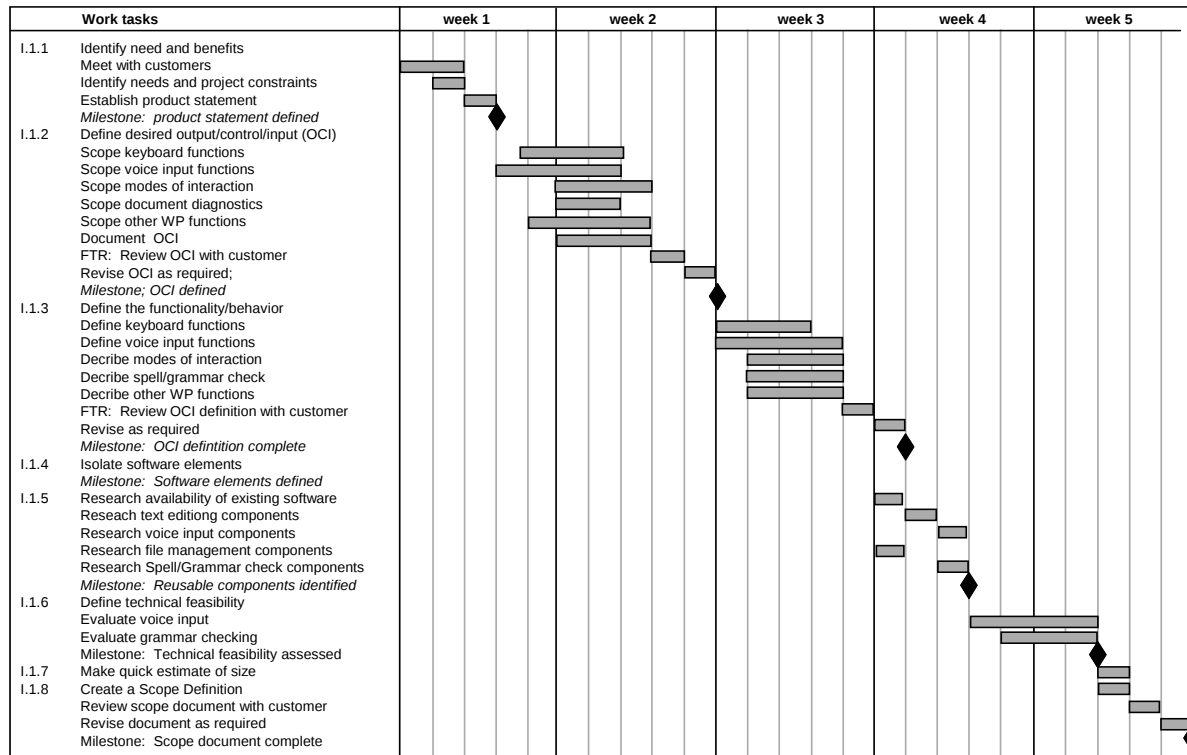
Projektin aikataulutus

- Aikataulutus
 - tehtäväverkko
 - Gantt - kaavio
- Otettava huomioon resurssit
 - montako tehtävää voidaan tehdä rinnakkain?
 - henkilökohtaiset kalenterit
 - jaetut resurssit

Aikataulutuksen periaatteet

- Jakaminen erillisiin tehtäviin - valitaan sopivat ohjelmistokehityksen tehtävät
- Vuorovaikutussuhteet - varmista että resurssit ovat käytössä, mitä työvaiheita pitää olla tehtynä ennen muita
- Ajanvaraus: aloitus- ja lopetuspäivät, työmäärä
- Tarkistus: aikataulut ja henkilöstön varaukset
- Määrittele vastuut
- Määrittele tuotokset
- Määrittele etapit - välietapitkin eli yhden tehtäväryhmän työvaiheet on saatu kunnialla päätökseen ja tulokset täyttävät annetut laatuvaatimukset

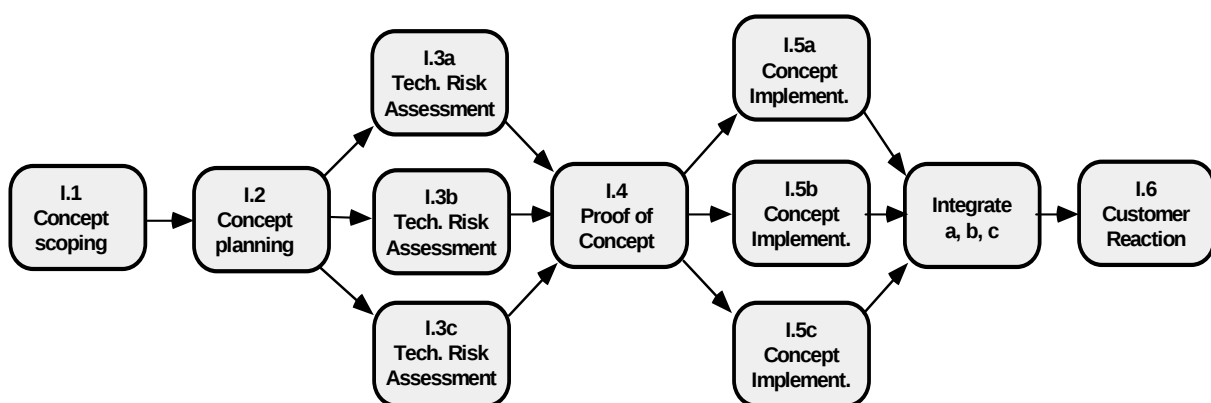
Gantt kaavio



Jyväskylän Yliopisto
Informaatioteknologian tiedekunta

Hiikka Heikkilä
Syksy 2004

Tehtäväverkko



Three I.3 tasks are
applied in parallel to
3 different concept
functions

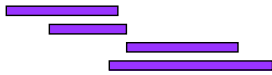
Three I.3 tasks are
applied in parallel to
3 different concept
functions

Kriittinen polku

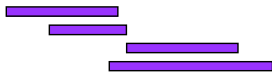
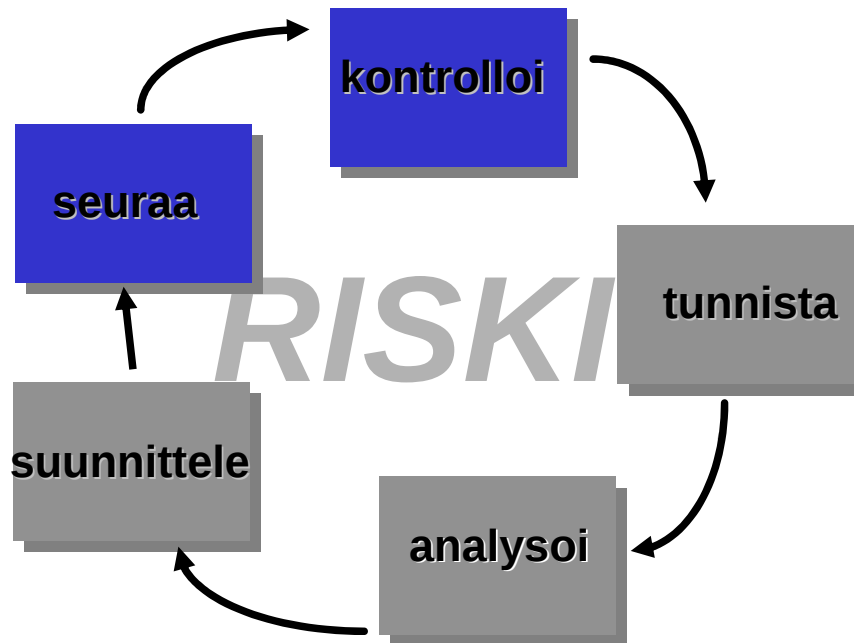
- tehtäväketju, jonka jonkin tehtävän myöhästyminen samalla myöhästyttää koko projektia
 - voi olla useita
 - määrättävissä tehtäväverkon perusteella
 - tehtäväverkon aikataulutus = tehtävien sitominen kalenteriin
 - jätettävä pelivaraa yllättäviä myöhästymisiä varten
 - kriittisen polun tehtävät tarkkaan seurantaan
 - muualla kuin kriittisellä polulla on pelivaraa, mutta jonkun tehtävän myöhästyminen voi tehdä siitä kriittisen polun

Riskien hallinta

Mitä voi mennä vikaan?
Millä todennäköisyydellä menee vikaan?
Mitä haittaa se aiheuttaisi?
Mitä voimme asialle tehdä?

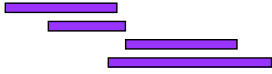


Riskien hallinta



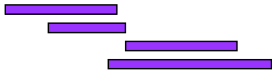
Riskien seurantataulukko

Riski	Todennäköisyys	Vaikutus	RMMM
			Riskien Lieventäminen Seuranta ja Hallinta



Riskeihin vaikuttaa

- Tuotteen koko
- Liike-elämän vaikutus
- Asiakas
- Prosessin kypsyys
- Teknologia
- Ihmiset



Projektin seuranta

Missä mennään?

Mitä on tehty?

Mitä ei ole tehty?

Miten pitkällä aloitetut tehtävät?

Mitkä tehtävät aloittamatta?

Mitä ongelmia on nähtävissä?

Projektin seuranta ja ohjaus

Seuranta

- Virallinen raportointi
 - Vaihetuotteiden hyväksynät, katselmukset, testitapausten läpimenot
- Viikkopalaverit /kuukausipalaverit
 - Projektin tilanne tehtävätasolla
- Management by walking
 - Henkilökohtaiset tilanteet
 - Yleinen tunnelma

Ohjaus

- Ehkäistään poikkeamia
- Korjataan poikkeamia
- Muutetaan suunnitelmia
- Ohjataan projektia päämäärään

Projektin päättäminen

- Loppukokous
 - analysointi porukalla
 - juhliminen, palkitseminen
- Loppuraportti
 - viimeinen tilanneraportti: toteutuma vs. suunnitelma, kesken olevat työt, vikalista
 - kertomus projektista: mitä tehtiin? mitä ei tehty?
 - projektin arviointi: missä onnistuttiin? missä ei? mitä kannatta tehdä samalla lailla / eri lailla seuraavalla kerralla? parannusehdotukset?
- Tuotteen luovutus
- Jatkotoimet
- Jälkiseuranta: asiakaspalaute

Projektin arviointi

- Projektin sisällön onnistuminen
 - vastaako projektin tulos asiakkaan tarpeita?
 - saadaanko tulokset hyödynnettyä?
 - ovatko johto ja projektiryhmä tyytyväisiä?
- Projektin aikataulullinen onnistuminen
 - toteutuiko projektin tavoitteena ollut kalenteriin sidottu aikataulu?
 - pysyttiinkö arvioidussa työmäärissä?
- Projektin taloudellinen onnistuminen
 - toteutuiko projekti budjetin mukaisesti?
 - saavutettiinkö taloudelliset tavoitteet?
 - onko projektin tulosten hyödyntäminen kannattavaa?

Miksi ohjelmistoprojektit epäonnistuvat?

Epärealistiset aikataulut suunnittelussa

Tekniset vaikeudet

Riskien hallinta puutteellista

Määrittely puutteellinen

Muutosvastarinta

Muuttuvat asiakasvaatimukset

Työmäärät arvioitu alakanttiin

Muut projektit

Projektin hallinnan epäonnistuminen

Ennustetut tai ennustamattomat riskit

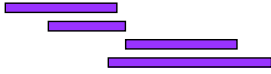
Henkilöstövaihtuvuus

Huono kommunikointi projektissa

Optimistinen asenne

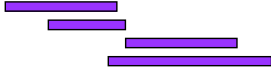
Omat kyvyt yliarvioitu

Organisaatiomuutokset



Miten menestyt projekteissa

- Opi edellisistä projekteista
- Panosta suunnitteluun
- Kysele työmääräarvioiden perusteita
- Riskien analysointi
- Mittareihin perustuva seuranta, vikojen seuranta
- Keskustele ja kysele edistymisestä
- Myönteinen ilmapiiri -> myöhästymisistä uskalletaan kertoa ajoissa
- Virheistä oppiminen
- Suunnitellut työkalu- ja/tai menetelmämuutokset
- Jaksa loppuun asti!
- Projektin loppuarviointi



Projektin aluksi

- Projektisuunnitelma
- Vaatimusmäärittely