**Kansilehti:**

Esitutkimus

Yleispätevä Lan hallinta järjestelmä

versio 1.0

Kansilehden alareunaan tai seuraavalle sivulle:

|  |  |  |
| --- | --- | --- |
| Oppilaitos | Koulutusohjelma | Opintojakso |
| Tekijä: <vastuuhenkilö> | | Tulostettu: 2.10.2014 |
| Jakelu: <kenelle jaellaan (ja ryhmän jäsenet)> | | |
|  | | |
|  | | |
| Dokumentin tila: Valmis tarkistukseen | | Muokattu: 1.10.2014 |
|  | |  |

**Versiohistoria**

|  |  |  |  |
| --- | --- | --- | --- |
| Versio | Päiväys | Tekijät | Selite (muutokset, korjaukset...) |
| 0.1 | 26.1.2018 | Otto Kujala | tehty liitteitä vaille valmiiksi |
|  |  |  |  |

Sisällysluettelo

[TUOTEIDEA 4](#_Toc504727273)

[PROJEKTIN ORGANISOINTI 4](#_Toc504727274)

[NYKYINEN JÄRJESTELMÄ 4](#_Toc504727275)

[HAVAITUT ONGELMAT JA RISKIT 4](#_Toc504727276)

[TAVOITTEET JA VAATIMUKSET 4](#_Toc504727277)

[RAJAUKSET 4](#_Toc504727278)

[YMPÄRISTÖ JA LIITTYMÄT 4](#_Toc504727279)

[HYÖDYT 4](#_Toc504727280)

[AIKATAULU 5](#_Toc504727281)

[KUSTANNUKSET 5](#_Toc504727282)

[TOTEUTUSVÄLINEET 5](#_Toc504727283)

[PROJEKTIN KANNATTAVUUS 5](#_Toc504727284)

[LISÄTIETOJA 5](#_Toc504727285)

[UML (Unified Modeling Language) 6](#_Toc504727286)

[UML sisältää 9 erityyppistä kaavioesitystä 6](#_Toc504727287)

[Käyttötapauskaaviot 7](#_Toc504727288)

[Esimerkki: Ajoneuvojen vuokrausjärjestelmä 8](#_Toc504727289)

[Esimerkki: Luentosalinvarausjärjestelmän käyttötapaukset (Haikala-Märijärvi) 10](#_Toc504727290)

[Esitutkimusraporttirunko 11](#_Toc504727291)

**TUOTEIDEA**

Ideana on tehdä nettisivut lan tapahtumaan.

Idea tuli mieleen koska kaikki lanit tarvitsevat nettisivut.

**PROJEKTIN ORGANISOINTI**

Projekti ryhmä ja sidosryhmät eivät ole tiedossa.

Asiakas on lanien järjestäjät.

**NYKYINEN JÄRJESTELMÄ**

Nykyistä järjestelmää ei ole.

**HAVAITUT ONGELMAT JA RISKIT**

Tässä vaiheessa esille tulevat riskit kuvataan vähentämis- ja poistomahdollisuuksineen.

Esimerkiksi aikataulu, outo sovellusalue, uudet toteutusvälineet, henkilöstön muut samanaikaiset projektit, alihankkijoiden toimitusaikataulut.

**TAVOITTEET JA VAATIMUKSET**

Tavoitteena on tehdä lan tapahtumaan nettisivu, johon sisältyy ilmoittautumisjärjestelmä, josta voi varata konepaikan.

**RAJAUKSET**

**YMPÄRISTÖ JA LIITTYMÄT**

Vaadittava/tarvittava ympäristö. Laitteisto-, ohjelmisto- ja tietoliikenneliittymät.

**HYÖDYT**

Saavutettavat hyödyt, suhteessa panostukseen. Onko projekti toteutettavissa kohtuullisella panostuksella. Onko projekti ylipäätänsä mahdollista toteuttaa.

Esimerkiksi 10 % kustannussäästö varastossa tai nopeampi tietojen käsittely tai aiemmin toteuttamaton toimintamalli.

**AIKATAULU**

Aikataulusta ei ole tietoa.

Projekti voisi olla valmis kolmen viikon kuluttua.

**KUSTANNUKSET**

**TOTEUTUSVÄLINEET**

Ohjelmointikielenä php, html, MySQL, javascript.

Tietokantana MySQL.

**PROJEKTIN KANNATTAVUUS**

Lukujen 7-10 perusteella voidaan kartoittaa kustannus-hyöty-suhde tai vastaava.

Mikäli sovellusalue on täysin outo, ja mikäli tarvittavat/vaaditut apuvälineet ovat outoja, kannattaa harkita tarkkaan millä ehdoilla projekti käynnistetään. Varsinkin jos henkilöstöresurssejakaan ei ole saatavilla laskettuna projektin kestoaikana.

Ratkaisun projektin aloittamisesta tekee viime kädessä firman johto. Tässä voidaan luetella perusteluita projektin aloittamiselle tai hylkäämiselle.

**LISÄTIETOJA**

Mistä saadaan lisätietoja projektin aiheesta ja sovellusalueesta.

Esimerkiksi viitteet standardeihin, direktiiveihin ja suosituksiin.

Mahdollisesti luettelo hyödyllisiä www-sivuja.

**UML (Unified Modeling Language)**

UML on kuvaustekniikka, johon on yhdistetty 3 tunnetuimman ns. 1. sukupolven oliomenetelmän käyttämät kuvaustekniikat (Boochin oliotekniikka, Runboughin ja kumppaneiden OMT sekä Jakobsonin OOSE). Sen peruskehittäjä on Rational Software.

**UML sisältää 9 erityyppistä kaavioesitystä**

Järjestelmän korkean tason toiminnallisuuden kuvaaminen

1. Käyttötapauskaavio (use case diagram): järjestelmän palvelut

Järjestelmän staattisen ja dynaamisen rakenteen kuvaaminen

1. Luokkakaavio (class diagram): tiedot ja mitä niillä tehdään
2. Oliokaavio (object diagram): tietojen väliset riippuvuudet
3. Sijoittelukaavio (deployment diagram): ohjelmiston sijoittelu laitteistolle
4. Komponenttikaavio (component diagram): ohjelman muodostuminen erillisistä komponenteista

Järjestelmän dynaamista käyttäytymistä kuvaavat kaaviot

1. Sekvenssikaavio
2. Yhteistyökaavio (sequence diagram): miten oliot toimivat yhdessä
3. Tilakaavio (statechart): olioiden käyttäytyminen, elinkaari
4. Aktiviteettikaavio (activity diagram): kontrollin kulku ohjelman suorituksessa

Nämä kaavioesitykset ovat osittain päällekkäisiä ja niissä voi käyttää samoja symboleja. Niitä voidaan soveltaa eri ohjelmistokehityksen vaiheissa. UML:n kaaviot ovat verkkomaisia; ts. niissä on elementtejä (geometrisia kuvioita kuten suorakaide ja ellipsi) ja niiden välisiä suhteita (erilaisia viivoja). Elementeillä on usein oma sisäinen rakenne ja suhteisiinkin voidaan liittää tarkempaa informaatiota. UML ei kuitenkaan määrittele tarkasti symbolien graafista ulkoasua, vaan se on jätetty työkaluvalmistajien (CASE-ohjelmistot) päätettäväksi.

**Käyttötapauskaaviot**

Käyttötapaus on kuvaus järjestelmän ja sen käyttäjien välisestä vuorovaikutuksesta tietyn, käyttäjän kannalta mielekkään tuloksen aikaansaamiseksi. Käyttötapauskaaviota käytetään systeemityön määritysvaiheessa järjestelmän toiminnallisten vaatimusten korkean tason kuvaamiseen; ts. käyttäjävaatimusten kartoittamiseen. Sillä siis kuvataan tapa, jolla tietty käyttäjä tai "käyttäjärooli" (actor) käyttää järjestelmää. Samalla luonnollisella henkilöllä voi olla useita käyttäjärooleja. Käyttötapaus alkaa aina jonkin käyttäjäroolin aloitteesta (heräte) ja päättyy siihen, että käyttäjä on saanut jonkin mielekkään tehtäväkokonaisuuden suoritettua.

Käyttötapauskaavioon voidaan merkitä myös käyttötapauksien välisiä suhteita; esimerkiksi käyttösuhteita (include), joiden avulla voi kuvata usealle käyttötapaukselle yhteisen osan erillisenä käyttötapauksena.

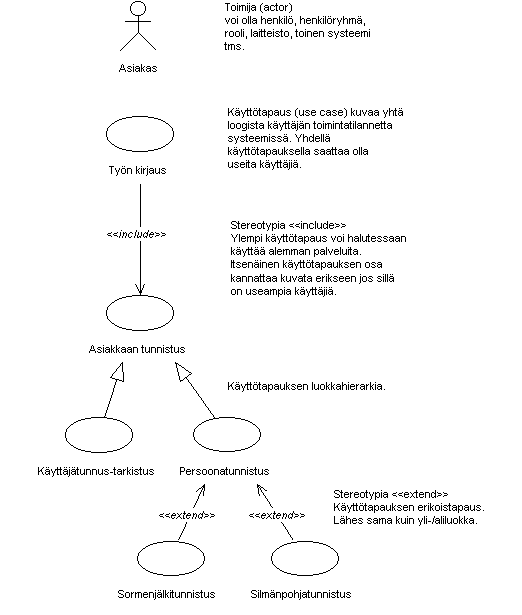
Käyttötapauskaavioiden keskeinen idea on olla riittävän yksinkertaisia ja konkreettisia, jotta kaikki asianosaiset ymmärtäisivät niiden sisällön. Käyttötapausten suunnittelun yhteydessä vältetään ottamasta kantaa itse toteutukseen.

Käyttötapaukset kattavat vain tärkeimmät osat järjestelmästä; niihin ei kannata ottaa mukaan kaikkia yksityiskohtia. Tarkemmat toimintokuvaukset tehdään yleensä vielä erikseen. Missään tapauksessa kuvaus ei saa olla yhtä A4: ta isompi. Mukaan kaavioon otetaan käyttäjän kannalta mielekäs kokonaisuus.

Käyttötapaukset muodostavat perustan järjestelmätestaukselle; ts. käyttötapauksen on muodostettava kokonaisuus, joka voidaan ajaa testivaiheessa yhtenä tai useampana peräkkäisenä testitapauksena. Käyttötapaukset toimivat perustana myös käyttöohjeiden laadinnalle.

Usein skenaariolla tarkoitetaan samaa kuin käyttötapauksella. UML:ssä skenaario (scenario) merkitsee käyttötapauksen yhtä ehdollista vaihtoehtoa. Esimerkiksi tilausten käsittelyn eri skenaarioita voisivat olla normaali tilaus, luottorajan ylitys ja tuotteen loppuminen/jälkitoimitus.

Kaaviossa käytetään seuraavia symboleja:



**Esimerkki: Ajoneuvojen vuokrausjärjestelmä**

Esimerkki: Ajoneuvojen vuokrausohjelmiston käyttötapauksia ovat ajoneuvon varaus, käyttöönotto, palautus ja huolto. Ulkoisia toimijoita ovat asiakas ja virkailija. Huolto sisältyy aina palautukseen. Virkailija osallistuu kaikkiin käyttötapauksiin ja asiakas muihin paitsi huoltoon.

include

Asiakas

Virkailija

Kaavio kuvaa käyttötapausten ulkoiset suhteet, mutta ei kerro mitään yksittäisen käyttötapauksen sisällöstä. Se on kätevintä kuvata sanallisella selostuksella.

Sanallinen kuvaus voi koostua seuraavista osista: (ei standardoitu UML:ssä)

* Käyttötapaus
* Heräte, joka käynnistää käyttötapauksen
* Käyttäjä/käyttäjärooli
* Toiminta eli tarkempi kuvaus
* Lopputulos

Sanalliset kuvaukset voi joissakin tapauksissa pyytää järjestelmän tulevia käyttäjiä tekemään.

MS Visiossa käyttötapauskaaviot saa näppärästi piirrettyä toiminnolla Software and Database/ UML Model Diagram / USE CASE.

**Esimerkki: Luentosalinvarausjärjestelmän käyttötapaukset (Haikala-Märijärvi)**

****

**Nimi:** Luentosalin varaaminen

**Suorittajat**: Kurssin vastuuhenkilö

**Esiehdot**: Vastuuhenkilö ja kurssi on syötetty järjestelmään

**Kuvaus**: Vastuuhenkilö seuraa WWW-linkkiä, joka johtaa järjestelmän pääsivulle. Hän syöttää järjestelmään käyttäjätunnuksensa ja salasanansa. Käyttäjä pyytää järjestelmää näyttämään salin varaustilanteen haluamaltaan aikaväliltä. Hän saa eteensä salin lukujärjestysnäytön.

Käyttäjä näkee näytöstä vapaat ajat sekä myös, mille kursseille sali on milloinkin varattu ja kuinka monelle viikolle. Käyttäjä tekee varauksen joltain vapaaksi havaitsemaltaan ajankohdalta.

**Poikkeukset**: Varaus ei onnistu: Varaustilanne on voinut muuttua sillä aikaa kun varaaja tekee varausta. Järjestelmä ilmoittaa tilanteesta käyttäjälle ja käyttäjä yrittää uudelleen.

**Lopputulos**: Varaukset kurssin luentoajoiksi on tehty.

**Muut** vaatimukset: Päivittäin käsitellään kiireisimpänäkin aikana enintään n. 100 varausta. Vastausajan on oltava alle 1 sekuntia, lukujärjestysnäytön päivitys saa kestää 5 sekuntia.

**Esitutkimusraporttirunko**

1. Tuoteidea
   1. Tuotteen tausta
   2. Ryhmän tausta
2. Projektin organisointi
3. Nykyinen järjestelmä
   1. Tekninen ympäristö
      1. Tulosteet
      2. Tietokannan rakenne ja kentät
      3. Käytössä olevat ohjelmistot
4. Havaitut ongelmat ja riskit
   1. Riskit ja niihin varautuminen
      1. Toimintakyky

Ongelma

Varautuminen

* + 1. Ammattitaito ja ajankäyttö

Ongelma

Varautuminen

* + 1. Odotukset

Ongelma

Varautuminen

* 1. Ongelmista toipuminen

1. Tavoitteet ja vaatimukset
   1. Toiminnalliset vaatimukset
   2. Ei-toiminnalliset vaatimukset
2. Tehtävä ja rajaukset
3. Hyödyt ja kustannukset
   1. Projektin tehtävä
   2. Projekti tekee
   3. Projekti ei tee
4. 8. Aikataulu
5. Toteutusvälineet

**Liitteet** (käyttötapauskaavio, johon viitataan toiminnallisten vaatimusten luvussa sekä käsitteistökuvaus)