

OptiLift

Tietotekniikan Sovellusprojekti

Lauri Laasala
Olli Lukkarinen
Ville Räisänen
Vesa Tanhua-Tyrkkö

Testausraportti
Versio 1.0
7.6.2004

Jyväskylän yliopisto
Tietotekniikan laitos

- Tekijät:** Laasala Lauri
Lukkarinen Olli
Räisänen Ville
Tanhua-Tyrkkö Vesa
- Projektin tiedot:** OptiLift-projekti
Kotisivu: <http://sovellusprojektit.it.jyu.fi/optilift/>
- Työn nimi:** OptiLift-projektin testausraportti.
- Työn kuvaus:** Testausraportti tietotekniikan Sovellusprojektiin.
- Tilaaaja:** Kilpa- ja huippu-urheilun tutkimuskeskus KIHU.
- Teettäjä:** Jyväskylän yliopisto, tietotekniikan laitos.
Vastaava ohjaaja: Markus Inkeroinen.
Tekninen ohjaaja: Ville Tirronen.
- Tiivistelmä:** Tämä dokumentti on Jyväskylän yliopiston keväällä 2004 toteutetun OptiLift-Sovellusprojektin testausraportti. Dokumentissa kuvataan projektin toteuttaman sovelluksen testausta testaus suunnitelman pohjalta.
- Avainsanat:** KIHU, kilpa- ja huippu-urheilun tutkimuskeskus, testausympäristö, testitapaus.

Versiohistoria

Versio	Päiväys	Tehnyt	Muutokset
0.1	27.4.2004	Ville Räisänen	Pohja dokumentille.
0.2	26.5.2004	Ville Räisänen	Lisätty yhteenvedot testi- ja virheraporteista. Kirjoitettu luku 4.
1.0	31.5.2004	Ville Räisänen	Lopullinen versio, yhdistetty testi- ja virheraporttien yhteenvedot ja lisätty versiot.

Sisältö

1	Johdanto	1
2	Termistöä	2
3	Testaaminen	3
3.1	Testilaitteistot.....	3
3.2	Sovelluksen testatut versiot.....	3
3.3	Yhteenveto testi- ja virheraporteista	4
4	Kommentteja.....	6
5	Lähteet.....	7
	Liitteet:	8

1 Johdanto

OptiLift-projekti suunnitteli ja toteutti Kilpa- ja huippu-urheilun tutkimuskeskukselle painonnoston levytankoharjoitteluun nostotekniikan automaattisen mittaus- ja analysointisovelluksen. Järjestelmän avulla analysoidaan urheilijan levytankoharjoittelua, joka on olennainen osa monien eri urheilulajien voimaharjoittelua. Oikean tekniikan hallitseminen on tärkeää niin vammattoman kuin mahdollisimman tehokkaankin harjoittelun kannalta. Monilla valmentajilla ei kuitenkaan ole riittävästi tietoa oikeista nostotekniikoista, jotta mahdollisimman hyödyllisen palautteen antaminen olisi mahdollista [3].

Projekti toteutettiin Jyväskylän yliopiston tietotekniikan sovellusprojektina kevään 2004 aikana. Projektiin liittyvistä käytännöistä ja toimintatavoista sekä projektiin kuuluvista henkilöistä on kerrottu tarkemmin projektisuunnitelmassa [4].

Tässä dokumentissa kuvataan sovelluksen testauksen tulokset. Sovellusta testattiin testaussuunnitelmassa [5] esiteltyjen testitapausten avulla.

2 Termistöä

Tässä luvussa selitetään yleisimmät sovelluksen testaamiseen liittyvät termit.

Alfa-testaus	Alfa-testauksessa tutkitaan täyttääkö esimerkiksi tilaustyönä tehty ohjelmisto asiakkaan asettamat vaatimukset. Tätä vaihetta jatketaan, kunnes systeemin kehittäjä ja asiakas hyväksyvät tuotteen ominaisuudet [2].
Beta-testaus	Yleensä laaja testausjakso ohjelmistolle ennen sen varsinaista käyttöönottoa ja jakelua [1].
FireWire (IEEE 1394)	Apple Computerin rekisteröimä tuotemerkki lisälaitteiden liitäntäteknikaksi mikrotietokoneille. FireWire tunnetaan paremmin nimellä IEEE 1394. Se on nopea (100, 200, 400 ja 800 Mbps), digitaalista sarjamuotoista dataa välittävä väylä [1].
Koodekki	Ohjelma tai laite, jota käytetään informaation pakkaamiseen ja purkamiseen. (engl. <i>code & decode= codec</i>).

3 Testaaminen

Tässä testausraportissa on testattu OptiLift-projektin toteuttamaa sovellusta testaussuunnitelman [5] mukaisesti. Kappaleessa 3.1 on esitelty testaamiseen käytetyt testilaitteistot. Kappaleessa 3.2 on esitelty sovelluksen versiot, joilla testaus suoritettiin ja kappaleessa 3.3 on yhteenveto testauksen yhteydessä kirjatusta testi- ja virheraporteista.

3.1 Testilaitteistot

Testilaitteistoina käytettiin kahta projektiryhmän käytössä ollutta pöytäkoneita (1, 3) niiden käyttötapausten osalta, joihin ei tarvittu kameraa. Muut testitapaukset testattiin kannettavalla tietokoneella (2) sekä siihen liitetyllä kameralla. Kannettavalla testattiin erityisesti myös sovelluksen ikkunointiin ja näytön resoluutioon liittyvät testitapaukset.

Nro	Laitteisto:
1	pöytäkone AMD Athlon XP 1600+ 1,41 GHz, 256Mb RAM, näytönohjain GeForce2 MX 400, näytön resoluutio 1280x1024 pikseliä, ei kameraa
2	kannettava P3 1,2 GHz, 512Mb Ram, näytönohjain Intel® 82830M, FireWire-liitäntä, näytön resoluution 1024x768 pikseliä, kamera Sony MiniDV HandyCam Vision
3	pöytäkone AMD Athlon XP 2600+, 1,91 GHz, 512MB RAM, näytönohjain, GeForce FX5200, näytön resoluution 1280x1024, ei kameraa

3.2 Sovelluksen testatut versiot

Taulukossa 3.1 on mainittu sovelluksen versiot, joilla testausta suoritettiin. Yleensä yhden testikierroksen suorittamisen ja testiraportin kirjoittamisen jälkeen siinä löydetty virheet korjattiin ja sovellus versioitiin uudelleen. Poikkeuksen muodostivat testiraportit 2 ja 3 sekä niihin liittyvät testikierrokset, jotka suoritettiin samalla sovelluksen versiolla.

Testiraportti, pvm	PVM	Sovelluksen versio
Liite 1, raportti #1	5.5	0.4
Liite 2, raportti #2	6.-7.5.	0.5
Liite 3, raportti #3	6.-7.5.	0.5
Liite 3, raportti #4	13.-14.5.	0.6
Liite 3, raportti #5	25.5.	0.7

Taulukko 3.1: Sovelluksen testatut versiot.

3.3 Yhteenveto testi- ja virheraporteista

Taulukossa 3.2 on esitetty yhteenveto testi- ja virheraporteista. Testiraportteja kirjoitettiin kaikkiaan 5 kappaletta. Raportit numero 3, 4 ja 5 on kirjattu samaan lomakkeeseen (Liite 3). Testitapausten 30 ja 31 toteuttaminen jää tilaajan tehtäväksi. Myöskään testitapausta 7 ei suoritettu, koska kameraan ei ollut manuaalia eikä kellään projektin jäsenistä ollut kokemusta digitaalisten videokasettien käytöstä.

Virheraportteja kirjoitettiin kaikkiaan 4 kappaletta. Raportit numero 1, 3 ja 4 on kirjattu samaan lomakkeeseen (Liite 4). Tapauksen 21 suorituksen yhteydessä tapahtunut sovelluksen kaatuminen oli ainoa kirjattu virhe, jota ei saatu korjattua.

Sovelluksen versiolla 0.7 suoritettu testaus onnistui lukuun ottamatta tapausta 21 sekä tapauksia 7, 30 ja 31, joita ei testattu. Sovellus siis täytti sille vaatimusmäärittelyssä [6] asetetut ensimmäisen luokan toiminnalliset vaatimukset, eli vaatimukset jotka sen on täytettävä.

ID	Testikerrat:	Virheitä kpl:	Ilmoitettu:	Korjattu:	Virheiden kuvaukset:	OK PVM:
1	2	0				6.5.
2	2	1	6.5.	10.5.	kaatuminen	10.5.
3	2	0				6.5.
4	3	1	6.5.	7.5.	painike piilossa, vanha kuva	14.5.
5	2	1	6.5.	7.5.	piste erottimena ei toimi	7.5.
6	2	1	6.5.	7.5.	kameran kuva katoaa, kaatuminen	7.5.
7	0	EI TEST.				

8	1	0				7.5.
9	2	1	6.5.	17.5.		25.5.
10	1	0				13.5.
11	3	2	7.5. , 5.5.	10.5. , 6.5.	kaatuminen, painikkeet piilossa	10.5.
12	3	1	5.5.	6.5.	eivät pysyneet synkronissa	6.5.
13	3	1	5.5.	5.5.	jää minimoiduksi	6.5.
14	1	0				14.5.
15	2	2	5.5. , 7.5.	5.5. , 10.5.	-merkit nimissä sotkevat, eri erottimet desimaaleissa	10.5.
16	2	1	5.5.	6.5.	ei korjaudu automaattisesti	6.5.
17	2	0				6.5.
18	2	0				7.5.
19	2	1	13.5.	14.5.		7.5.
20	1	1	7.5.	7.5.	ei huomioitu	7.5.
21	2	1	14.5.	-	sovelluksen kaatuminen	-
22	3	1	5.5.	5.5.	rekisteritiedostossa ongelmia	5.5.
23	3	1	5.5.	6.5.	'file not found'	6.5.
24	3	1	5.5.	5.5.	vertailuikkuna ruman näköinen, virheilmoitus	14.5.
25	3	0				7.5.
26	2	1	7.5.	tulostus: 11.5. , help: 25.5.	tulostus ja program help ei toimi	25.5.
27	2	1	13.5.	24.5.	värienvaihtaminen ei toimi	24.5.
28	2	0				14.5.
29	1	0				14.5.
30	0	EI TEST.				-
31	0	EI TEST.				-
32	2	0				14.5.
33	3	1	14.5.	19.5.	x ja y-distanceissa väärää arvoja	19.5.

Taulukko 3.2: Yhteenveto testi- ja virheraporteista.

4 Kommentteja

Sovelluksen testaaminen päästiin aloittamaan hieman myöhässä siinä tarvittavien laitteistojen vuoksi. Kannettavan tietokoneen, jossa on FireWire-liitäntä, metsästäminen vei aikansa ja sen saaminen Agoranettiin ja muutenkin käyttökuntoon vei vielä lisää aikaa. Toinen tärkeä osa testilaitteistoa, eli digitaalinen videokamera, oli KIHU:ltä lainassa aika ajoin, mutta sitäkään ei yleensä saatu pitää lainassa hyvin pitkiä aikoja kerrallaan. Näin ollen testaamista jouduttiin suorittamaan sopivissa rakosissa, kun aikataulut antoivat periksi ja käytössä oli sopivat laitteistot.

Sovelluksen testaamiselle tunnusomaista olivat pitkät toimosarjat. Esimerkiksi jotta pääsi testaamaan nostojen poistamista tai tallentamista, täytyi lisätä nostajia suoritusryhmään, kalibroida ja tallentaa nostosuorituksia. Näin pientenkin asioiden testaaminen saattoi vaatia paljon toimintoja.

Virheitä testattaessa löytyi kuitenkin yllättävän vähän. Niiden korjaaminen oli helppoa ja yleensä pystyi heti sanomaan, missä kohtaa koodia virhe ja korjaustarve oli.

Sovelluksen testaamista Beta-testauksena jatkaa tästä eteenpäin tilaaja. Sovellus versioidaan projektin päättyttyä 1.0:ksi.

5 Lähteet

- [1] Jaakohuhta, Hannu, IT Ensyklopedia, Sanasto, Edita Oyj, Helsinki, 2001.
- [2] Kautto Tuomas, Ohjelmistotestaus ja siinä käytettävät työkalut, Ohjelmistotekniikan seminaariesitelmä, Jyväskylän yliopisto, saatavilla WWW-muodossa <URL: <http://www.mit.jyu.fi/opiskelu/seminaarit/ohjelmistotekniikka/testaus/>>, 21.11.1996.
- [3] Keränen Tapani, Viitasalo Jukka ym., Nostotekniikan automaattinen mittaus- ja analysointijärjestelmä-esittelymoniste, KIHU, Jyväskylä, 2003.
- [4] Laasala Lauri, Lukkarinen Olli, Räisänen Ville ja Tanhua-Tyrkkö Vesa, OptiLift-Sovellusprojektin projektisuunnitelma, Jyväskylän yliopisto, Tietotekniikan laitos, 2004.
- [5] Laasala Lauri, Lukkarinen Olli, Räisänen Ville ja Tanhua-Tyrkkö Vesa, OptiLift-Sovellusprojektin testausuunnitelma, Jyväskylän yliopisto, Tietotekniikan laitos, 2004.
- [6] Laasala Lauri, Lukkarinen Olli, Räisänen Ville ja Tanhua-Tyrkkö Vesa, OptiLift-Sovellusprojektin vaatimusmäärittely, Jyväskylän yliopisto, Tietotekniikan laitos, 2004.

Liitteet:

Liite 1: Testiraportti 1

Liite 2: Testiraportti 2

Liite 3: Testiraportit 3, 4, 5

Liite 4: Virheraportit 1, 3, 4

Liite 5: Virheraportti 2