

Vaatimusdokumentti

Ketutus-ryhmä

Helsinki 17.7.2007

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Kurssi

581260 Ohjelmistotuotantoprojekti (6 ov)

Projektiryhmä

Aleksanteri Aaltonen

Sari Kajantie

Jani Koivukoski

Onni Koskinen

Ilkka Kosunen

Risto Sandvik

Asiakas

Kai Kuikkaniemi

Johtoryhmä

Kimmo Simola

Jaakko Saaristo

Kotisivu

<http://www.cs.helsinki.fi/group/ketutus>

Versiohistoria

Versio	Päiväys	Tehdyt muutokset
0.1	13.6.2007	Ensimmäinen versio
0.2	26.6.2007	Vaatimusten luokittelu uudistettu
0.3	27.6.2007	Käyttötapaukset lisätty
0.4	29.6.2007	Ryhmäpalaverin muutokset
0.5	3.7.2007	"Elinkaari"-luvun hionta
0.6	5.7.2007	Asiakastapaamisen kommentit
0.7	17.7.2007	1. Julkinen julkaisu

Sisältö

1	Johdanto	1
2	Sanasto	1
3	Järjestelmän yleiskuvaus	2
3.1	BitTorrent-laajennoksen kuvaus	2
3.2	Käyttäjryhmät tarpeineen	3
3.3	Eräitä käyttötapauksia	3
4	Vaatimukset	4
4.1	Järjestelmävaatimukset	4
4.1.1	Käyttöympäristövaatimukset	4
4.1.2	Toiminnalliset järjestelmävaatimukset	5
4.1.3	Ei-toiminnalliset järjestelmävaatimukset	5
4.2	Käyttäjävaatimukset	5
4.2.1	Toiminnalliset käyttäjävaatimukset	6
4.2.2	Ei-toiminnalliset käyttäjävaatimukset	7
5	Järjestelmän elinkaari	7

1 Johdanto

Ketutus on Helsingin yliopiston tietojenkäsittelytieteen laitoksen kesän 2007 ohjelmistotuotantoprojekti, jonka tavoitteena on tuottaa BitTorrent-laajennos Mozilla Firefox-selaimeen. Laajennuksen avulla käyttäjä voi ladata torrent-tiedostoja joko samaan tapaan kuin tavallisella BitTorrent-asiakalla tai ikkunoidusti sivulle upotetusta torrentfile-resurssista. Laajennuksella ei sen sijaan voi luoda torrent-tiedostoja, eikä siten laittaa uutta materiaalia jakoon.

Projektissa tuotettava laajennos toteutetaan FusionFox-BitTorrent-moottorin päälle. Asiakas on toteuttanut moottorin Helsinki Institute for Information Technologyssä (HIIT) osana laajempaa EU-rahoitteista eurooppalaisten yliopistojen yhteistä P2P-Fusion-hanketta. Hankkeessa on kehitetty esimerkiksi ikkunoitua BitTorrent-protokollaa, jolla tiedostoja voidaan ladata reaaliaikaisena lähetyksenä (engl. *streaming*), BitTorrentin tavallisesta sirpaleittaisesta lataustavasta poiketen.

Tässä vaatimusdokumentissa kuvataan BitTorrent-laajennukselle asetettavat vaatimukset. Siten dokumentissa myös määritellään ne laajennuksen ominaisuudet sekä ominaisuuksien toteutusprioriteetit, joista asiakas ja Ketutus-ryhmä sopivat keskenään.

2 Sanasto

BitTorrent	Vertaisverkkoprotokolla, jossa jaettava data lähetetään ja vastaanotetaan pienissä paloissa vähiten jaossa olevia paloja priorisoiden.
DOM-malli	Tekniikka, joka mahdollistaa HTML-sivujen muokkauksen selaimessa esimerkiksi JavaScriptillä.
Download	Tämän dokumentin puitteissa tietojen tai tiedon lataamista verkosta.
Download Manager	Mozilla Firefox -selaimen mukana toimitettava tiedostoja lataava komponentti käyttöliittymineen.
Firefox	Mozilla Foundationin julkaisema avoimeen lähdekoodiin perustuva selain, virallisesti Mozilla Firefox.
Ikkunoitu BitTorrent	Perinteisen sirpaloituneen lataamisen sijaan tiedostoa ladataan selkeissä erissä tiedoston alkupäästä lähtien. Kunkin erän sisällä lataaminen tapahtuu sirpaloidusti.

JavaScript	Lähinnä WWW-ympäristössä käytettävä komentosarjakieli.
Libtorrent	Avoimella lähdekoodilla toteutettu BitTorrent-moottori.
Streaming	Tämän dokumentin puitteissa: reaaliaikainen jakaminen, jonka seurauksena tiedostoja tai tiedon osia voi käyttää ennen kokonaisuuden saapumista perille.
Torrent	Tiedosto, joka sisältää ladattavien tiedostojen tarkistussummia sekä tiedon siitä, mistä osoitteesta löytyy BitTorrent-asiakkaista kirjaa pitävä keskuspalvelin.
Upload	Tämän dokumentin puitteissa: tietojen tai tiedon lähettämistä verkkoon.
XPCOM	Cross Platform Component Object Model - Mozilla Foundationin kehittämä alustariippumaton komponenttimalli, jota käytetään myös Firefoxin komponenteissa.
XUL	XML User-interface Language - Mozilla Foundationin kehittämä XML-pohjainen käyttöliittymien kuvauskieli, jota käytetään myös Firefoxin käyttöliittymissä.

3 Järjestelmän yleiskuvaus

3.1 BitTorrent-laajennoksen kuvaus

Asiakkaan tavoitteena on saada BitTorrent-laajennos, jossa on siinä määrin kehittyneet ominaisuudet, ettei käyttäjä tarvitse enää erillistä BitTorrent-asiakasohjelmaa tiedostojen lataamiseen.

Järjestelmän tulee asiakkaan näkemyksen mukaan tarjota vähintään Opera-selaimen BitTorrent-laajennoksen mukainen toiminnallisuus kuitenkin mahdollisimman huolellisesti integroituna Firefox-selaimeseen. Tällöin BitTorrent-lataukset näkyisivät ensisijaisesti käyttäjälle ”Download Managerin” käyttöliittymässä, kuten muutkin lataukset, joskin esimerkiksi tunnistettavalla värillä. Ikkunoiden ja asetusten säätövalikoiden tulisi noudattaa Firefoxin yleistä ulkoasuteemaa. Asiakas myös toivoo rauhallista ulkoasua.

Laajennos rakennetaan asiakkaan tarjoaman XPCOM-komponenttina toteutetun BitTorrent-moottorin päälle JavaScript-ohjelmointikielellä käyttäen Douglas Crockfordin ohjelmointityyliä ja XUL-käyttöliittymänkuvauskieltä.

Optionaaliseksi vaatimusmäärittelyvaiheessa jätetty XPCOM paketointi mahdollistaisi sisäisen BitTorrent moottorin vaihtamisen toiseen, esimerkiksi vapaassa levityksessä olevaan libtorrent-kirjastoon. Se toteutetaan aikataulun salliessa.

3.2 Käyttäjärhyhmät tarpeineen

Ketutus-ryhmä tunnisti laajennukselle kaksi erillistä käyttäjäryhmää:

1. peruskäyttäjät
2. tehokäyttäjät

Peruskäyttäjille olennaista on mahdollisimman selkeä ja helppokäyttöinen käyttöliittymä. WWW-sivun linkin kautta BitTorrentilla jaettavaa videoresurssia tavoitteleva peruskäyttäjä ei välttämättä edes halua tietää käyttävänsä BitTorrentia. Toiminnan on oltava peruskäyttäjän näkökulmasta mahdollisimman läpinäkyvää. Niinpä peruskäyttäjälle tarjotaan käyttöliittymäksi lähtökohtaisesti Download Manageria. Edistymispalkki kuitenkin eroaa ulkonäöltään muiden protokollien latauksissa käytettävistä latauspalkeista, sillä peruskäyttäjänkin on tarpeellista tietää, ettei lataus ole samanlainen kuin muilla protokollilla. Muilla protokollillahan lataus on kokonaan lineaarinen.

Tehokäyttäjän on sen sijaan kyettävä kontrolloimaan siirtotapahtumaa mahdollisimman tarkasti. Tehokäyttäjälle tarjottava käyttöliittymä sallii latauksen asetusten tarkemman kontrollin sekä lataustapahtuman yksityiskohtaisemman seurannan.

Kummallekin käyttäjäryhmälle tarjotaan sekä ikkunoitua että tavallista BitTorrentia.

3.3 Eräitä käyttötapauksia

K1:

Lähtötilanne: Peruskäyttäjä klikkaa WWW-sivulla olevaa linkkiä nähdäkseen P2P-Fusion-hankkeen yhteydessä jaettavan opetusvideon.

Tavoite: Käyttäjä haluaa saada videon mahdollisimman nopeasti auki ja vaivattomasti käyttöönsä.

K2:

Lähtötilanne: Peruskäyttäjä haluaa ladata verkosta Fedora Core 7:n asennus-imagen. Käyttäjä käynnistää latauksen yöksi.

Tavoite: Käyttäjä haluaa saada imagen mahdollisimman vaivattomasti käyttöönsä. Kulutetun kaistan määrällä tai latauksen käynnistymisen nopeudella ei ole käyttäjälle merkitystä.

K3:

Lähtötilanne: Tehokäyttäjä haluaa ladata yliopiston videojakelusta eläinlääketieteen kurssivideot, joista hän haluaa nähdä ensin lehmän endokrinologiaa käsittelevän jakson. Käyttäjä työskentelee samalla interaktiivisesti yliopiston unix-palvelimella.

Tavoite: Tehokäyttäjä haluaa vaikuttaa latausoperaatioon: Ladattavasta joukosta tulee voida priorisoida yksittäisiä kohteita ja latauksen käyttämän kaistan määrää tulee voida säätää, jotta latenssia huonosti sietävät palvelut olisivat samaan aikaan käytettävissä.

4 Vaatimukset

Vaatimukset on koottu haastatteleamalla asiakasta Ketutus-ryhmän kokouksissa sekä BitTorrentin tehokäyttäjiä kokousten ulkopuolella.

Järjestelmävaatimusten konkretisoinnin apuna on käytetty libtorrentin API-kuvausta, sillä se tarjoaa hyvän yleiskuvan torrent-tiedostojen käsittelyyn liittyvistä toiminnoista.

4.1 Järjestelmävaatimukset

4.1.1 Käyttöympäristövaatimukset

1. Ohjelmiston tulee toimia Firefox 2.0 -sarjan selaimissa.
Korkea prioriteetti.
2. Ohjelmiston tulee osata ladata sivulle upotetut torrentfile-resurssit.
Korkea prioriteetti.
3. Ohjelmisto tulee voida asettaa torrent-MIME-tyypin oletuskäsittelijäksi.
Korkea prioriteetti.
4. Ohjelmiston tulee olla asennettavissa XPI-paketista.
Korkea prioriteetti.
5. Ohjelmiston tulee käyttää asiakkaan tarjoamaa, XPCOM-komponenttina toteutettua, BitTorrent-moottoria.
Korkea prioriteetti.
6. Libtorrent paketoidaan XPCOM-komponentiksi.
Matala prioriteetti.

4.1.2 Toiminnalliset järjestelmävaatimukset

1. Ohjelmiston tulee voida avata torrent ja ladata sen viittaamat tiedostot.
Korkea prioriteetti.
2. Ohjelmiston tulee osata ladata sivulle upotetut torrentfile-resurssit.
Korkea prioriteetti.
3. Ohjelmistossa tulee voida valita ladataanko tiedostot ikkunoimatonta vai ikkunoitua BitTorrent-protokollaa hyödyntäen ja myös määritellä ikkunan koko.
Korkea prioriteetti.
4. Torrent-latauksia tulee voida keskeyttää ja jatkaa. Jatkamisen tulee onnistua myös selaimen oltua välillä suljettuna.
Korkea prioriteetti.
5. Libtorrent paketoidaan XPCOM-komponentiksi.
Matala prioriteetti.

4.1.3 Ei-toiminnalliset järjestelmävaatimukset

1. JavaScript-koodin tulee noudattaa Douglas Crockfordin JavaScript-ohjelmointityyliä.
(<http://www.crockford.com/javascript/>)
Korkea prioriteetti.
2. Käyttöliittymä hyväksytetään asiakkaalla.
Korkea prioriteetti.
3. Ohjelmiston kieli on englanti.
Korkea prioriteetti.
4. Merkkijonoliteraaleja ei sijoiteta ohjelmakoodiin vaan ulkoiseen tiedostoon, jotta kääntäminen muille kielille on helppoa.
Korkea prioriteetti.
5. Käyttöohje ja ylläpitodokumentti kirjoitetaan englanniksi.
Korkea prioriteetti.

4.2 Käyttäjävaatimukset

Peruskäyttäjää koskevat vaatimukset tulevat pääsääntöisesti asiakkaalta kun taas teho-käyttäjävaatimuksia on saatu asiakkaan lisäksi ryhmän jäseniltä sekä ryhmän haastattelemilta aktiivisilta BitTorrent-käyttäjiltä.

4.2.1 Toiminnalliset käyttäjävaatimukset

1. Upload- ja download-nopeuksille tulee pystyä asettamaan rajat, joita torrent-lataukset eivät saa yhteensä ylittää.
Korkea prioriteetti.
2. Käyttäjän tulee voida säätää torrentin viittaamien tiedostojen latausprioriteettia tiedostokohtaisesti; vähintään prioriteetella *korkea*, *matala* ja *ei ladata*.
Korkea prioriteetti.
3. Käyttäjän tulee voida määritellä ohjelmiston käyttämät tietoliikenneportit.
Korkea prioriteetti.
4. Käyttäjän tulee voida säätää ohjelmiston asetuksia liitännäisellä asetusdialogilla ja Firefoxin about:config-liittymällä.
Korkea prioriteetti.
5. Käyttäjän tulee voida asettaa yhtäaikaan ladattavien torrentien maksimimäärä
Keskitason prioriteetti.
6. Käyttäjän tulee voida määritellä asetuksissa kuinka kauan torrentia jaetaan latauksen valmistuttua. Jakaminen tulee voida säätää Loppumaan myös ennalta-asetetun jakosuhteen täytyttyä.
Keskitason prioriteetti.
7. Käyttäjän tulee voida asettaa upload- ja download-nopeuksille torrent-kohtaiset maksimit.
Keskitason prioriteetti.
8. Käyttäjän tulee voida määrätä torrentien latausjärjestys, kun torrenteja on enemmän kuin yhtäaikaaisesti halutaan ladata.
Keskitason prioriteetti.
9. Käyttäjän on voitava säätää latauksen tallennuspaikkaa.
Keskitason prioriteetti.
10. Latauksen ei tule alkaa ennen kuin käyttäjä on hyväksynyt tallennuspaikan.
Keskitason prioriteetti.
11. Käyttäjän tulee voida asettaa lähetyspaikkojen (kohdekoneiden) määrä torrentia kohden.
Matala prioriteetti.

4.2.2 Ei-toiminnalliset käyttäjävaatimukset

1. Latauksen tulee näkyä Firefoxin Download Managerissa.
Korkea prioriteetti.
2. Torrent-latauksien tulee erottua Download Managerissa tavallisista latauksista.
Korkea prioriteetti.
3. Torrent-latausten oletushakemiston tulee olla sama kuin Firefoxin tavallisten latausten oletushakemisto.
Keskitason prioriteetti.

5 Järjestelmän elinkaari

Ketutusryhmä käyttää ohjelmiston kehityksessä vesiputousmallia ja testaus tapahtuu V-mallin mukaan.

Ryhmä kartoittaa vaatimukset haastatteleamalla asiakkaan edustajaa sekä BitTorrent-käyttäjiä. Kohderyhmä valikoidaan ryhmän jäsenten tuntemien tehokäyttäjien joukosta sekä lähettämällä kyselyt Mozilla-kehitysfoorumeille. Lisäksi ryhmä itse kerää ryhmäkokouksissa ominaisuuksia, joita hyvällä BitTorrent-asiakaskäyttöliittymällä tulisi olla. Vaatimukset kootaan tähän vaatimusdokumenttiin, joka hyväksytetään asiakkaalla. Asiakkaalla hyväksytetään myös ehdotus järjestelmän käyttöliittymäksi.

Vaatimusmäärittelyn ja käyttöliittymäehdotuksen valmistuttua ryhmä suunnittelee, toteuttaa sekä testaa kattavasti ohjelmiston, joka vastaa vaatimusmäärittelyä.

Kaikki vaiheet dokumentoidaan. Asiakkaalle toimitetaan käyttöohje sekä ylläpitodokumentti, jossa muuta dokumentaatiota täydentäen käsitellään ohjelmiston ylläpitoon liittyviä seikkoja.

Ylläpitovaihe ei kuulu ohjelmistotuotantoprojektin aikatauluun, joten ylläpito jää asiakkaan sekä mahdollisten muiden jatkokehittäjien vastuulle. Jotta jatkokehitys onnistuisi järkevästi, ryhmä tuottaa jatkokehitykseen vaikuttavat dokumentit englanniksi.