

The Functional Architecture Modeling Method applied on Web Browsers

Kristian Wahlroos - 014417003

Referaatti
HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Helsinki, 31. tammikuuta 2016

Wilbert Seelen, Shaheen Syedin ja Sjaak Brinkkemperin julkaisemassa tutkimuksessa “The Functional Architecture Modeling Method applied on Web Browsers“ vuodelta 2014 ottaa analysoitavaksi funktionaalisen arkkitehtuurin mallinnuksen (Functional Architecture Modelling), jonka pohja luotiin Sjaak Brinkkemperin ja Stella Pachidin julkaisussa “Functional Architecture Modeling for the Software Product Industry“ vuodelta 2010. Tutkimuksessa myös kerrotaan menetelmä luoda funktionaalinen malli järjestelmästä. Itse analysointi julkaisussa tapahtuu tapaustutkimuksen kautta, jossa analysoidaan mallin toimivuutta luomalla funktionaalinen malli kuvitteellisesta web-selaimesta, johon on lisätty sama toiminnallisuus kuin oikean maailman selaimessa olisi. Tätä mallia verrataan toiseen tutkimukseen, jossa koodianalyysi-työkalulla luotiin korkean tason malli oikean maailman web-selaimen lähdekoodista. Funktionaalinen malli koettiin onnistuneeksi, koska mallit vastasivat loppuanalyysissä toisiaan.

Tutkimus “The Functional Architecture Modeling Method applied on Web Browsers“ alkaa kertomalla, että arkkitehtuurin design ja dokumentointi on projekteissa usein keskeneräistä tai puuttuu jopa kokonaan. Tähän ongelmaan he esittävät ratkaisuksi funktionaalisen arkkitehtuurin mallinnuksen. He väittävät, että tämä arkkitehtuurin mallinnustapa on virtaviivaistettu tyyli, joka antaa tilaa nopeille arkkitehtuurin iteraatiolle, koska keskittyminen on ollut mallin helppokäyttöisyydessä ja kommunikaatioon tärkeydessä. Tämä uusi malli myös yhdistää funktionaalista ja teknistä arkkitehtuuria ja antaa visualisaation järjestelmän ydin toiminnallisuuteen ja sen avulla nähdään järjestelmän eri osien lopulliset tarkoitukset.

Tutkimuksen tarkoituksena on ollut pienentää dokumentoinnin ja lopullisen toiminnallisuuden välistä kuilua tämän uuden mallin avulla. He jatkavat kertomalla, että itse mallilla on kaksi päätarkoitusta: mahdollistaa selkeä korkean tason näkymä projektiin sekä ylläpitää hyviä käytänteitä ohjelmoinnissa, koska se pakottaa jakamaan toteutettavan järjestelmän toiminnallisiin moduuleihin jo projektin alkuvaiheessa. He määrittelevät seuraavaksi vastattavan tutkimuskysymyksen muodossa “Miten toiminnallisuutta voidaan ilmaista ohjelmistoarkkitehtuurien malleissa?“, johon he vastaavat, että adaptoimalla Brinkkemperin ja Pachidin luomaa metodia.

Nykyisestä kirjallisuudesta he löytävät materiaalia tutkimuksen tueksi funktionaalisen arkkitehtuurin mallinnuksesta ja vaatimuksista ohjelmistoarkkitehtuurin mallinnuksessa. He ottavat esille ADL-mallien (Architecture Description Language) ongelmia; ADL yrittää mallintaa koko järjestelmän arkkitehtuuria laajamittaisesti, ADL ei täytä suunnittelijoiden vaatimuksia, vaikeasti ymmärrettävä sekä ADL tukee vain yhtä näkymää. He nostavatkin esille UML-mallin, josta on tullut nykypäivänä niin sanottu “de facto“-standardi arkkitehtuurin kuvauksessa. Syy tähän on, että UML on kevyempi käyttää ja visuaalisen muotonsa takia helpommin lähestyttävä.

Seuraavaksi tutkimuksessa avataan itse menetelmä arkkitehtuurin kuvaukseen, joka on muokattu hieman aiemmasta “Functional Architecture Modeling for the Software Product Industry“-julkaisusta. Uudessa menetelmässä suunnittelu ei enää keskity moduläärisyyteen, vaihtelevuuteen tai yhteentoimivuuteen, vaan on haluttu korostaa helppokäyttöisyyttä, kommunikoinnin tärkeyttä, tukea useille näkymille, rajoittavuuden vähentämistä sekä nykyisiä laatuvaatimuksia malleissa. Itse luotu malli koostuu neljästä eri vaiheesta: vaatimusten keräys, ominaisuuksien mallinnus (feature modelling), toiminnallisen arkkitehtuurin suunnitteleminen ja lopuksi teknisen arkkitehtuurin suunnitteleminen. Vaatimusten keräyksessä kerätään toiminnalliset ja tekniset vaatimuksen osakkailta, ominaisuuksien mallinnuksessa vaatimuksen yhdistetään aiheiden mukaan, toiminnallisen arkkitehtuurin suunnittelemisessa edellisen vaiheen tuotokset yhdistetään erillisiksi moduuleiksi, joiden välille määritellään yhteydet ja niiden tyypit, toiminnallisen arkkitehtuurin suunnittelemisessa moduuleita tarkennetaan teknisten vaatimusten avulla. Lopullinen dokumentaatio, joka on käytännössä UML-tyylinen malli, näytetään osakkaille. Tätä mallia pystytään näin tarkentamaan iteraatioittain sekä ohjelmointi pystytään jo aloittamaan.

He analysoivat metodin toimintaa tapaustutkimuksen kautta, jossa mallinnettiin web-selainta ja jonka vaatimuksista mukaan otettiin vain ne vaatimukset, jotka ovat ydintoiminnallisuutta web-selaimelle. Tapaustutkimuksessa luotiin ensin korkean tason näkymä, josta jokaista luotua moduulia sitten tarkennettiin tarpeen mukaisesti. Yhteensä kolme eri mallia luotiin tutkimuksessa: korkean tason toiminnallinen arkkitehtuuri, tarkennettu kuvaus eräästä moduulista sekä korkean tason tekninen arkkitehtuuri. Analysointivaiheessa tutkittiin, että miten tekninen arkkitehtuuri vastaa koodianalyysin avulla tuotettua arkkitehtuuria. Tuloksena saatiin, että ainoa erilaisuus löytyi vain yhdestä moduulista, joka oli ylimääräinen. Syy tälle löytyi heidän mukaansa siitä, että moduuli on todennäköisesti oikeissa selaimisissa sulautettu muihin moduuleihin.