

Aihe 1

Lue artikkeli:

Buschmann, Frank; Ameller, David; Ayala, Claudia P.; Cabot, Jordi; Franch, Xavier; ,
“**Architecture Quality Revisited**,” Software, IEEE , vol.29, no.4, pp.22-24, July-Aug. 2012:
<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6265077&isnumber=6265068>

ja valmistaudu keskustelemaan siitä. Mieti artikkelissa esitettyjen havaintojen ja johtopäätösten suhdetta luennoilla ja kurssikirjassa esitettyyn

- vaadittujen laatuominaisuuksien ja rajoitteiden (NFR) suhde arkkitehtuuriin
- arkkitehtuurin käyttötavat (arkkitehtuuri on yhdentekevä, arkkitehtuurikeskeinen, vivutus)
- riskit ja niiden hallinta arkkitehtuurin avulla

[NFR = Non-Functional Requirement = Quality Attribute requirements & cConstraints = vaaditut laatuominaisuudet ja taloudellis-teknis-viranomais-organisatoriset rajoitteet]

Aihe 2

Tkt-laitoksen laskuharjoituksien vetäjät haluaisivat helpottaa ja nopeuttaa tehtyjen harjoitustehtävien merkitsemistä laskareissa. Paperilistojen kierrättäminen harjoituskerran aluksi on paitsi antiikkista myös aikaa vievää. Harjoitusten jälkeen vetäjät joutuvat sitten käsin syöttämään tehdyt tehtävät paperilistoista laitoksen KURK I-kurssikirjanpitojärjestelmään, mihin siihenkin kuluu aikaa, ja virheitäkin tapahtuu helposti.

Ryhmien vetäjät ovat visioineet modernin ratkaisun, joka mahdollistaisi seuraavat toiminnalliset skenaariot

- Opiskelijoilla on käytettävissään erilaisilla päätelaitteilla (PC, tabletti, älypuhelin) ohjelma, jonka kautta he merkitsevät tekemänsä tehtävät
- Opiskelijat voivat nähdä kaikilla harjoituskerroilla merkitsemänsä tehtävät
- Ryhmän vetäjä voi päättää aikaikkunan, jolloin tehtäviä voi merkitä ja merkintöjä peruuttaa
- Ryhmän vetäjällä on käytössään oma ohjelma tai käyttöliittymä, josta hän voi reaaliajassa nähdä tietyn harjoituskerran merkintätilanteen
- Ryhmän vetäjä voi omassa ohjelmassaan merkitä harjoituskerran tietoihin, ketkä opiskelijat ovat esittäneen oman ratkaisunsa
- Ryhmän vetäjä voi nähdä tilastotietoa, kuinka monta tehtävää kukin opiskelija on esittänyt koko kurssin aikana
- Ohjelma auttaa vetäjää valitsemaan ratkaisun esittäjän käyttäen erilaisia algoritmeja (esim. satunnainen valinta suodattaen jo aikaisemmin ratkaisuja esittäneet pois). Valinta-algoritmeja pitäisi pystyä lisäämään ohjelmaan.
- Myös ryhmän vetäjän liittymä on käytettävissä erilaisilla päätelaitteilla

- Ryhmän vetäjä voi lisätä yksittäisiin tehtäviin omia merkintöjään (vaikea/liian helppo tehtävä jne.)
- Ryhmän vetäjä voi tehdä muutoksia opiskelijoiden merkintöihin (virheellisen merkinnän korjaus aikaikkunan sulkeutumisen jälkeen)
- Tehdyt merkinnät ja mahdolliset korjaukset päivittyvät automaattisesti KURKI-järjestelmään ilman, että ryhmän vetäjän tarvitsee niitä käsin sinne syöttää

Rajoituksia ratkaisulle

- Järjestelmä pyritään toteuttamaan pääasiassa oppilastyönä (OhTu-projekteissa)
- Halutaan käyttää opensource- tai muuten ilmaiseksi saatavilla olevia kirjastoja/kehyskiä
- Järjestelmän täytyy toimia laitoksen IT-ympäristössä, uutta laitteistoa ei ole tarkoitus hankkia
- Opiskelija- ja opettajaliittymien/aplikaatioiden on toimittava yleisimmillä päätelaitteilla

Pohdi, mitkä yllämainituista vaatimuksista ja rajoitteista vaikuttavat arkkitehtuuriratkaisuihin ja minkälaisia riskejä toteutusprojektiin (tai projekteihin) liittyy. Mistä seikoista haluaisit lisää tietoa tai täsmennyksiä, jotta voisit paremmin arvioida niiden arkkitehtuurisia vaikutuksia?