

2. Laskarit

Toinen sykli

Syklin teemat:

- *Laadunvarmistussuunnitelma*
- *Projektissa käytettävät mittarit*

Syklin tuotos: Toinen versio ohjelmistosta

Kukin ryhmä tekee ohjelmistolleen laadunvarmistussuunnitelma. Siitä käy ilmi, miten tehtävän tuotteen laatua valvotaan ja varmistetaan.

Kukin ryhmä valitsee mittarit, joita ryhmässä käytetään valvottaessa prosessin ja tuotteen laatua. Mittarit ovat osa laadunvarmistussuunnitelmaa.

Laadunvarmistussuunnitelma

1 Testausstrategia

Kaikki koodi yksikkötestataan paitsi käyttöliittymän osalta (85 % testikattavuus). Käyttöliittymän toiminta varmistetaan järjestelmätestauksessa.

Yksikkötestaus tehdään samaan aikaan koodauksen kanssa. Järjestelmätestaus tehdään jokaisen uuden toiminnallisuuden valmistuttua.

Yksikkötestien kattavuutta seurataan. (ECLEMMMA(.org))

2 Aikataulu

Aikataulu määritetään jokaisen sprintin alussa. Sprintin lopussa määritetään tehty act-työmäärä ja tarkistetaan jäljellä oleva työmäärä.

3 Projektinhallinnan laadunvarmistus

3.1 Projektin seuranta ja ohjaus

Säännölliset suunnittelukokoukset viikon alussa. Viikon lopulla pidetään tarvittaessa lyhyet tarkistuskokoukset.

3.2 Proseduurit ja työohjeet

Työohjeet tallennetaan google.docs-sivustolle.

Lähdekoodit dokumentoidaan käyttäen runsaasti koodin toimintaa kuvaavia kommentteja.

3.3 Versionhallinta ja dokumenttien valvonta

Versionhallinta toteutetaan git-versionhallintasovelluksella. gitHub:ssa sijaitsee keskusrepository.

Dokumentit ylläpidetään GoogleDocs:n avulla. Sivuilla sijaitsevat mm. työohjeet ja backlogit.

3.4 Projektin laadunvarmistuksen metriikat

Backlogin avulla seurataan kunkin resurssin taskeja, työmääriä ja valmiusastetta.

3.5 Katselmointi

Kaikille luokille tehdään vertaisarviointi, jossa kolleega käy läpi versionhallintaan laitettuja luokkia.

4 Standardointi

Sovelluskehityksessä käytetään junit-testimallia. Sovellus rakennetaan arkkitehtuurisesti käyttäen MVC-mallia.

Ohjelmistossa käytettävät laadun mittarit

1 Toiminnallisuus (Functionality)

Backlog ylläpitää toivottuja toiminnallisuuksia, joiden tulee olla kaikkien toteutettuna tai kuitattuna (esim. toteutetaan versiossa X.x).

2 Luotettavuus (Reliability)

Kattavat yksikkötestit (85 %, paitsi käyttöliittymäkomponenteille).

3 Käytettävyys (Usability)

Käytettävyyttä mitataan asiantuntija-arvioinnilla.

4 Tehokkuus (Efficiency)

Järjestelmätestauksen yhteydessä mitataan operatioiden vasteajat.

5 Ylläpidettävyys (Maintainability)

Java-tyylin mukaista koodia.
Yritetään välttää spagettikoodia ja purkkaratkaisuja.

6 Siirrettävyys (Portability)

Ohjelmisto testataan useilla eri alustoilla jokaisen sprintin aikana. Alustoina ovat ainakin Linux, Windows ja MacOSX.

Prosessin laadun mittarit

Tuottavuus koodirivien määrä viikoittain.

Asiakkaan raportointien virheiden määrä per 1000 riviä kohden.

Työtehokkuuden osalta mitataan työaikaa suhteessa kokonaistyöaikaan.

Sprintissä sovittujen taskien toteutuminen aikataulussa.