XP (Extreme programming)

By: Antti, Topi ja Adrian

XP Ydinarvot

- ***Extreme Programming (XP) painottaa muiden ketterien menetelmien tapaan mukautuvuutta enemmän kuin ennustettavuutta. Menetelmän tarkoitus on parantaa sekä ohjelmiston laatua että tiimin kykyä vastata asiakkaan vaatimusten muuttumiseen.***
- Asiakasvaatimukset voidaan käydä läpi välittömästi ja tuoda mukaan kehitystyöhön.
- XP:ssä useiden toistuvien ohjelmistojulkaisujen ja lyhyiden kehityssyklien tarkoitus on parantaa tuottavuutta ja tarjota tarkastuspisteitä kehityssyklien välissä.
- XP perustuu viiteen ydinarvoon sekä niiden pohjalta luotuihin periaatteisiin ja käytäntöihin. XP:n arvot ja käytännöt on suunniteltu siten, että muutoksen kustannus pysyy projektin ajan suunnilleen samana, eikä kasva eksponentiaalisesti ohjelmistoprojektin edetessä.

Suunnittelupelin vaiheet

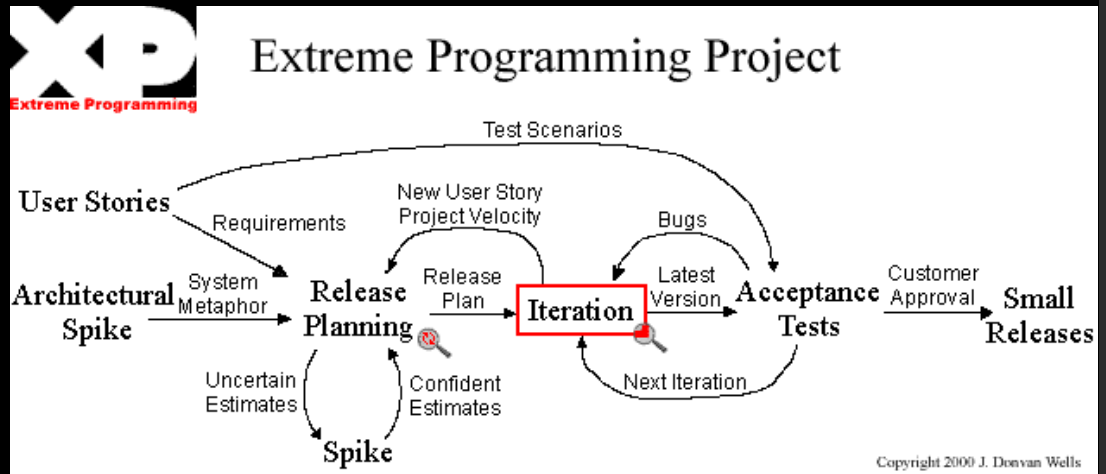
- Pelisuunnittelussa rakennetaan pelille perusmalli.
- Toisin kuin voisi luulla, pelisuunnitteluun ei varsinaisesti liity grafiikan tekeminen tai koodaaminen.
- 1. Konseptointi
- 2. Pelikonseptointidokumentti
- 3. Suunnittelu
- 4. Tuotanto
- 4.1 Budjetti ja aikataulu
- 4.1.1 Kukatekee ja mitätekee
- 5. Testaus
- Tuotantovaihe jolloin toteutettua pelikokonaisuutta testataan.
  Testausvaiheessa eliminoidaan mahdolliset pelissä ilmenevät ongelmat.
  Pääasiassa etsitään ohjelmointivirheitä.
- Pelistä tuotetaan toimiva versio,
  jonka testausta jatketaan. Lopputuotteen täytyy olla virheetön.

Yksinkertaiset vaatimukset

- Tietokone
- Tietokyky
- Tarve tai halu tehdä/käyttää

Pienet julistukset

- Pienet julistukset ovat usein julkaistuja ohjelman versiota, joita julkaistaan asiakkaille, että he näkevät työn edistymisen. Jotkut tiimit julkaisevat, joka päivä uuden version ohjelmasta, mutta 1 julkaisu viikossa on myös hyvä rytmi julkaisuille.
- Jokaisen version täytyy olla testattu ja toimiva ennen kuin ne julkaistaan.
- On myös tärkeää saada asiakkailta ja käyttäjiltä palautetta ajoissa sillä mitä nopeammin tiedät ongelman sitä enemmän aikaa on korjata se.

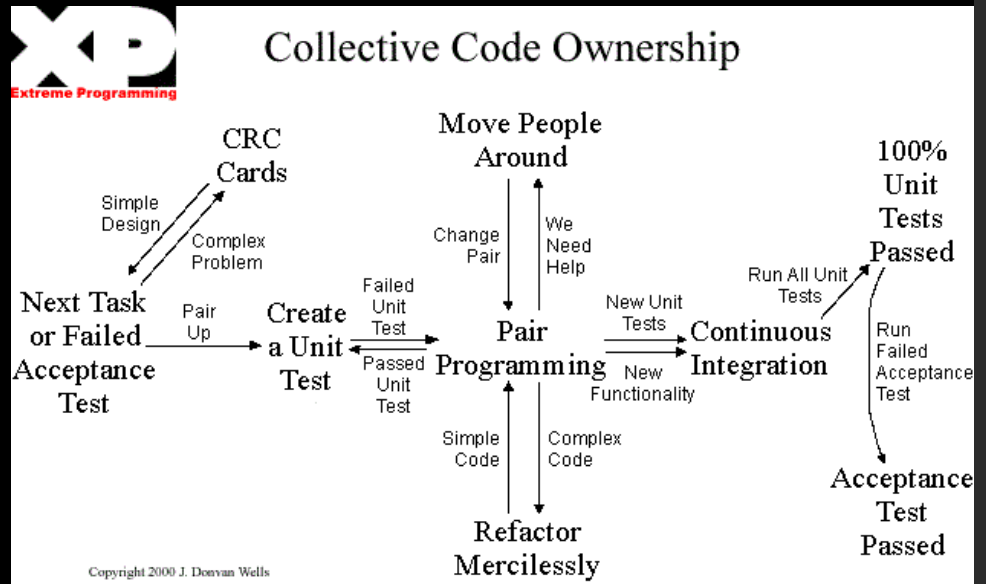


Pariohjelmointi

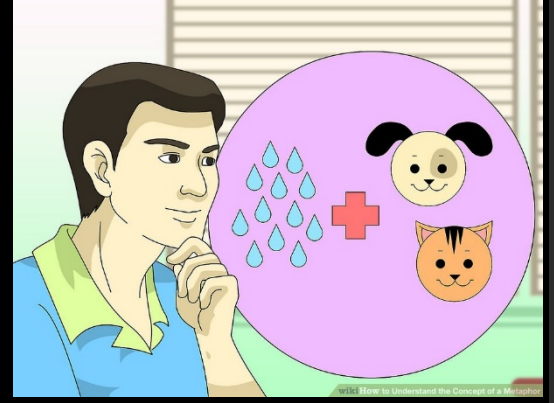
- Pariohjelmoinnissa 2 henkilöä työskentelee yhdellä tietokoneella.
- Tarkoituksena on olla vuorotellen rooleina ns. ”navigoija”, joka auttaa ohjaajaa antamalla hänelle vinkkejä ja sanomalla jos huomaa virheen ja ”ohjaaja”, joka tekee koodia ja pistää navigoijan ideat toteutukseen.
- Tarkoituksena on saada laadukkaampaa koodia, jossa on vähemmän virheitä pistämällä kaksi henkilöä keskittymään samaan asiaan.
- Pariohjelmointi ei ole opetuskäyttöön sillä pariohjelmoinnissa kummankin on tarkoitus osallistua kunnolla ohjelmointiin.

Koodin yhteisomistus

- Koodin yhteisomistus tarkoittaa sitä, että kaikki ohjelmoijista voi muokata mitä vaan koodilohkoista, korjata niiden bugeja jne. ja että kaikilla olisi jonkinlainen tietämys jokaisesta koodin osasta.
- Tämä tarkoittaa, että kuka vaan voi parantaa mitä vaan koodin osaa sen sijaan, että lähettää parannusehdotuksen koodin alkuperäiselle tekijälle
- Yhteisomistus ei toimi ilman hyvää kommunikointia, versionhallintaa ja jatkuvaa integrointia.
- Yhteisomistuksessa pitää myös varmistaa, että muutokset eivät aiheuta ristiriitoja muissa ohjelmiston osissa.



Vertauskuva



- Ohjelmistojen systeemien rakentaminen tarvitsee kommunikointia systeemin vaatimuksista ohjelmoijille. Formaalisissa ohjelmointi metologiassa tämä tehtiin dokumentoinnin kautta.
- XP:n tekniikan suoritustavoissa voidaan katsoa menetelminä rakentaa ja levittää nopeasti institutionaalista tietoa ohjelmoijien kesken. Tavoitteena on antaa kaikille ohjelmoijille yhtenäinen näkemys systeemistä, joka vastaa käyttäjien näkemystä.
- Tähän menetelmään Extreme Programming suosii yksinkertaisia suunnittelua, yleisiä vertauskuvia, käyttäjien ja ohjelmoijien yhteistyö, tiheää verbaalista kommunikointia ja palautetta.
- Vertauskuvien tarkoitus on saada asiakas, ohjelmoijat ja esimiehet ymmärtämään miten systeemi toimii toiminnan nimen kautta. Esimerkiksi: loan_records(class) for borrowers(class). Jos toiminto olisi myöhässä, se tekisi make_overdue operaation catalogue(class) luokalle.

Lähteet

- http://users.jyu.fi/~jorma/kandi/2007/Kandi_Lajunen.pdf
- http://www.extremeprogramming.org/
- http://www.extremeprogramming.org/rules/collective.html
- http://www.extremeprogramming.org/rules/pair.html
- http://www.extremeprogramming.org/rules/releaseoften.html
- https://en.wikipedia.org/wiki/Extreme_programming
- https://en.wikipedia.org/wiki/Extreme_programming_practices#System_metaphor
- https://www.wikihow.com/Understand-the-Concept-of-a-Metaphor#/Image:Understand-the-Concept-of-a-Metaphor-Step-1-Version-2.jpg
- http://trc.utu.fi/embedded/kasikirja/1/3/
- http://mlab.uiah.fi/~jwall/tuotanto/tuotanto.html