

Vesiputousmalli:

- Vesiputousmallissa suunnittelu- ja toteutusprosessi etenee yhdestä vaiheesta toiseen.
- Suurien projektien vaiheita mahdotonta suunnitella etukäteen.
- Ongelmien ilmetessä tau asiakkaan muuttaessa vaatimuksiaan suunnitelma menee uusiksi.
- Ongelmia on vaikeata korjata jälkikäteen.

Ketterä Kehitys:

- Agile Manifesto, 17 ketterän kehityksen puolesta puhujaa. 4 arvoa, 12 periaatetta. Saatavilla monella kielellä.
- 12 Periaatetta:
 - o Tyydyttää asiakas toimimalla tämän tarpeet täyttäviä versioita säännöllisesti.
 - o Täyttää muuttuvat vaatimukset myös kehityksen myöhäisessä vaiheessa.
 - o Toimittaa toimivia versioita ohjelmistosta säännöllisesti.
 - o Edustajien ja kehittäjien työskentelyä yhdessä päivittäin.
 - o Projektit rakennetaan motivoituneiden yksilöiden ympärille.
 - o Toimivin tapa tiedon välittämiseksi kehitystiimille ja tiimin jäsenten kesken.
 - o Plerb
- 4 Arvoa
 - o Yksilöitä ja vuorovaikutusta
 - o Toimivaa sovellusta
 - o Asiakasyhteistyötä
 - o Muutokseen reagoimista

Scrum:

- Projektihallinnan viitekehys.
- Scrumissa ryhmä tähtää toimimaan yksikkönä ja toimimaan tiiviissä yhteistyössä.
- Scrum toimii erilaisissa tilanteissa ja on sopeutuva, sekä itse ohjautuva.
- Ytimessä toimii palaverit. Suunnittelupalaveri ennen projektin alkua ja päivittäin palaveri joka kestää n. 15min.
- Scrummasteri vastaa tiimin puskemisesta oikeaan suuntaan valmentamalla ja esteiden eliminoimisella.
- Kehitystiimi vastaa suunnittelusta ja toteutuksesta.
- Kehitystiimiä saatetaan kutsua toimitusryhmäksi, välttääkseen sekaannuksen, että ryhmään kuuluisi pelkkiä ohjelmoijia.

XP:

- XP painottaa mukautuvuutta enemmän kuin ennustettavuutta.
- Painotus yhteistyössä.
- Tuodaan asiakas mukaan kehitystyöhön.
- Tiimit julkaisevat pieniä versioita ohjelmistoista, näyttääkseen asiakkaalle työn edistymisen.
- Pariohjelmointi, eli kaksi henkilöä samalla tietokoneella. Toinen navigoi ja huomaa virheitä ja toinen luo ja korjaa koodia.

- Kaikki kehittäjät omistavat kaiken koodin, eli kuka tahansa voi päivittää ja huoltaa koodia, vaikka se ei olisi alun perin heidän. Tämä vaatii kommunikointia ja versiohallintaa.

Lean:

- Leanissa tähdätään jätteen minimointi ja prosessi tuotannon maksimointi. Jätteiksi katsotaan epäolennaista- ja turhaa koodia.
- MVP (Minimum Viable Product) eli prototyyppi versio jossa on kaikki ominaisuudet, jota asiakas tarvitsee tuotteelta.
- Kanban, eli visuaalinen työkalu jolla osoitetaan milloin tuotannon tulisi alkaa ja loppua.

TDD:

- Kehitystä johtaa testitapaus, jokainen ohjelman osa testataan ennen käyttöönottoa.
- Testikoodin pohjalta on helppo lisätä uusia ominaisuuksia ja korjata virheitä.

RUP:

- Ohjelmistokehityksen prosessikehys.
- Voimaantulovaihe, tutkitaan järjestelmää ja katsotaan onko järjestelmä kannattava toteuttaa.
- Kehittelyvaihe, vähentää pahimmat havaitut riskit.
- Rakennusvaihe, make a guess!
- Muutostavoite, tuotanto ja loppukäyttäjä.
- Työtä kannattaa suunnitella visuaalisesti.
- Valvo laatua, tee testejä projektin aikana.
- Hallinnoi muutoksia via Versiohallinta.
-