

Kursuseprogramm

| | | | |
|--|--|-----------------|-------|
| IFI6054.DT | Agiilne tarkvaraarendus | | |
| 3 EAP | Kontakttundide maht: 28 | Õppesemester: K | Eksam |
| Eesmärk: | Ülevaate andmine kaasaegsetest tarkvaraarendusmetoodikatest, tarkvaraarenduse trendidest, samuti protsessis kasutatavatest vahenditest. | | |
| Aine lühikirjeldus: (sh iseseisva töö sisu kirjeldus vastavuses iseseisva töö mahule) | Tarkvara arendamise meetodikad: monumentaal- ja agiilsed meetodikad, nende kontseptuaalsed erinevused. Agiilse tarkvaraarenduse eeldused. Meeskond agiilses tarkvaraarenduses. Inkrementaalsus ja iteratiivsus. Agiilsete meetodikate arengulugu ja erinevad kehasused. Ekstreemprogrammeerimise ja teiste enamlevinud agiilsete meetodikate rõhuasetused. Agiilse tarkvaraarenduse põhipraktikad. Töökorraldus ja -protsess agiilses tarkvaraarendusmeeskonnas. Tarkvaraarendusmeeskonna poolt kasutatavad vahendid (tarkvara). | | |
| Õpiväljundid: | <ul style="list-style-type: none"> • Teadmised tänapäevastest agiilsetest tarkvaraarenduse meetodikatest: õppija teab ning kirjeldab agiilse tarkvaraarenduse aluspõhimõtteid ning oskab selgitada erinevusi monumentaal- ja agiilsete meetodikate vahel. • Arusaamine agiilse tarkvaraarenduse praktikatest, nende rakendamise eesmärkidest: õppija oskab loetleda põhipraktikaid, selgitada nende olemust, mõju ning praktikate vahelisi seoseid, põhjendada nende vajalikkust. • Suutlikkus liituda agiilset arendusmeetodikat kasutava meeskonna töösse: õppija tunneb arenduse võtmetegevusi, mõistab nende sisu ning tunneb tarkvaraarenduse protsessis kasutatavat terminoloogiat; teab, milliseid vahendeid ning millistel eesmärkidel tarkvaraarendusmeeskond kasutab; mõistab koodistandardi kasutamise vajalikkust ning kasutab seda; suudab iseseisvalt kasutada versioonihaldussüsteeme. | | |
| Hindamismeetodid: | Hinde panekul arvestatakse kirjalikku eksamitööd (50%), tööd kontakttundides (40%) ja referatiivset kodutööd (10%). Eksamile pääsemise eelduseks on osalemine vähemalt 80% kontakttundides. Praktiseerivatel agiilsete arendusmeeskondade juhtliikmetel (näiteks <i>XP Coach</i> , <i>agile team lead</i> jt) ning sertifitseeritud spetsialistidel (näiteks CSM, CSPO, CSD, CSP jt) on võimalik vastava kokkuleppe korral eksamile pääseda muudel tingimustel. | | |
| Õppejõud: | Marek Kusmin | | |

| | |
|---|---|
| Ingliskeelne nimetus: | Agile Software Development |
| Eeldusaine: | IFI6069.DT - Programmeerimise põhikursus |
| Kohustuslik kirjandus: | Robert C. Martin, 2008. Clean Code: A Handbook of Agile Software Craftsmanship. Pearson Education, Inc. |
| Asenduskirjandus: (üliõpilase poolt läbi töötatava kirjanduse loetelu, mis katab ainekursuse loengulist osa) | Asenduskirjanduse alusel pole võimalik antud kursust läbida. |
| Õppetöös osalemise ja eksamile/arvestusele pääsemise nõuded | Eksamile pääsemise eelduseks on osalemine vähemalt 80% kontakttundides. Praktiseerivatel agiilsete arendusmeeskondade juhtliikmetel (näiteks <i>XP Coach, agile team lead</i> jt) ning sertifitseeritud spetsialistidel (näiteks CSM, CSPO, CSD, CSP jt) on võimalik vastava kokkuleppe korral eksamile pääseda madalama kontakttundide osalusprotsendiga. |
| Iseseisva töö nõuded | Referatiivse töö koostamine või essee kirjutamine õppejõu poolt antud teemal. |
| Eksami hindamiskriteeriumid või arvestuse sooritamiseks vajalik miinimumtase | <p><i>Teadmised tänapäevastest agiilsetest tarkvaraarenduse meetodikatest:</i></p> <p>A – Õppija omab selget ülevaadet monumentaal- ja agiilsetest meetodikatest, oskab neid analüüsida, tuua välja mõlemat tüüpi meetodikate eelised ja puudused erinevates olukordades. Loetleb erinevaid agiilseid meetodikaid, teab nende rõhuasetusi ning oskab analüüsida nende erinevusi. Tunneb põhjalikult agiilsete meetodikate aluspõhimõtteid, agiilse tarkvaraarenduse manifesti. Valdab vabalt teemakohast terminoloogiat. Suudab argumenteeritult diskuteerida erinevate meetodikate rakendamise teemadel erinevates olukordades ja projektides, kaitsta oma seisukohti. Kui õppijale kirjeldatakse reaalseid või hüpoteetilisi olukordi tarkvaraarendusprojektis või -meeskonnas, siis teeb ettepanekuid erinevate olukordade lahendamiseks ning põhjendab neid.</p> <p>B – Õppija omab selget ülevaadet monumentaal- ja agiilsetest meetodikatest, oskab neid analüüsida, tuua välja mõlemat tüüpi meetodikate eelised ja puudused erinevates olukordades. Loetleb erinevaid agiilseid meetodikaid, teab nende rõhuasetusi ning oskab analüüsida nende erinevusi. Tunneb põhjalikult agiilsete meetodikate aluspõhimõtteid, agiilse tarkvaraarenduse manifesti. Valdab teemakohast terminoloogiat. Suudab argumenteeritult diskuteerida erinevate meetodikate rakendamise teemadel erinevates olukordades ja projektides, kaitsta oma seisukohti.</p> |

C – Õppija omab ülevaadet monumentaal- ja agiilsetetest metoodikatest, loetleb mõlemat tüüpi metoodikate eelised ja puudused erinevates olukordades. Loetleb erinevaid agiilseid metoodikaid, teab nende rõhuasetusi. Tunneb agiilsete metoodikate aluspõhimõtteid, agiilse tarkvaraarenduse manifesti. Valdab teemakohast terminoloogiat.

D – Õppija omab ülevaadet monumentaal- ja agiilsetetest metoodikatest, loetleb agiilsete metoodikate eelised. Loetleb erinevaid agiilseid metoodikaid. Teab agiilse tarkvaraarenduse manifesti sisu. Teab teemakohaseid põhimõisteid.

E – Õppija omab ülevaadet agiilsetetest metoodikatest. Oskab nimetada paari levinumat agiilset metoodikat. Teab agiilse tarkvaraarenduse manifesti sisu. Teab teemakohaseid põhimõisteid.

Arusaamine agiilse tarkvaraarenduse praktikatest, nende rakendamise eesmärkidest:

A – Õppija teab kõiki agiilse tarkvaraarenduse manifestis nimetatud agiilset praktikat, selgitab nende sisu, analüüsib nendevahelisi seoseid. Põhjustab erinevate praktikate olulisust. Kirjeldab praktikate realiseerimist arendusmeeskonnas. Kui õppijale kirjeldada arendusmeeskonnas ette tulevaid olukordi, oskab teha argumenteeritud ettepanekuid praktikate rakendamise osas.

B – Õppija teab kõiki agiilse tarkvaraarenduse manifestis nimetatud agiilset praktikat, selgitab nende sisu, analüüsib nendevahelisi seoseid. Põhjustab erinevate praktikate olulisust.

C – Õppija teab kõiki agiilse tarkvaraarenduse manifestis nimetatud agiilset praktikat, selgitab nende sisu. Põhjustab erinevate praktikate olulisust.

D – Õppija teab kõiki agiilse tarkvaraarenduse manifestis nimetatud agiilset praktikat, selgitab nende sisu.

E – Õppija suudab loetleda kõiki agiilse tarkvaraarenduse manifestis nimetatud agiilset praktikat.

Suutlikkus liituda agiilset arendusmetoodikat kasutava meeskonna töösse:

A – Õppija teab tarkvaraarenduse võtmetegevusi ja tunneb meeskonnatöö vahendeid, selgitada korrektset terminoloogiat kasutades võtmetegevuste sisu ja kodeerimisstandardi kasutamise vajalikkust. Kasutab versioonihaldussüsteeme. Oskab kirjutada koodi vastavalt etteantud kodeerimisstandardile. Oskab luua ühikteste. Oskab luua koodi testjuhitud programmeerimise alusel.

| | |
|--|---|
| | <p>B – Õppija teab tarkvaraarenduse võtmetegevusi ja tunneb meeskonnatöö vahendeid, selgitada korrektset terminoloogiat kasutades võtmetegevuste sisu ja kodeerimisstandardi kasutamise vajalikkust. Kasutab versioonihaldussüsteeme. Oskab kirjutada koodi vastavalt etteantud kodeerimisstandardile. Oskab luua ühikteste.</p> <p>C – Õppija teab tarkvaraarenduse võtmetegevusi ja tunneb meeskonnatöö vahendeid, selgitada korrektset terminoloogiat kasutades võtmetegevuste sisu ja kodeerimisstandardi kasutamise vajalikkust. Kasutab põhioperatsioonide ulatuses versioonihaldussüsteeme. Oskab kirjutada koodi vastavalt etteantud kodeerimisstandardile.</p> <p>D – Õppija teab tarkvaraarenduse võtmetegevusi ja tunneb meeskonnatöö vahendeid, selgitada korrektset terminoloogiat kasutades võtmetegevuste sisu ja kodeerimisstandardi kasutamise vajalikkust. Oskab põhioperatsioonide ulatuses kasutada versioonihaldussüsteeme. Oskab selgitada koodistandardi järgimise olulisust.</p> <p>E – Õppija oskab loetleda tarkvaraarenduse võtmetegevusi ja meeskonnatöö vahendeid, tunneb vastavat terminoloogiat. Oskab loetleda versioonihaldussüsteemide kasutamise põhioperatsioone.</p> |
| <p>Informatsioon kursuse sisu kohta, kursuse jaotumine teemade kaupa sh kontaktundide ajad</p> | <p>E 30.01.2017- Sissejuhatus. Tarkvaraarenduse eesmärgid. Tarvara elutsükkel. Arendussammud. Eesti kutsestandard – noorem tarkvaraarendaja.</p> <p>E 06.02.2017- “Kõik korraga” vs inkrementaalsus. Inkrementaalsus vs iteratiivsus. Inkrementaalsus + iteratiivsus. Metoodikad. Monumentaalmetoodikad vs agiolsed metoodikad - helikopterivaade. Väärtuse loomine vs üldkulud. Loojad ja kiibitsejad.</p> <p>E 13.02.2017- Agiilse tarkvaraarenduse manifest. Agiilse tarkvaraarenduse 12 põhiprintsiipi. Agiilsed metoodikad. Scrum vs XP - ülevaade.</p> <p>E 20.02.2017- Kihiline mudel. XP väärtused. XP kriitika.</p> <p>E 27.02.2017- Agiilse tarkvaraarenduse praktikad. Seosed praktikate vahel.</p> <p>E 06.03.2017- User story. Story telling. Hindamine. Story board. Prioritiseerimine.</p> <p>E 13.03.2017- Arendustempo. Burn-up ja burn-down statistika.</p> <p>E 27.03.2017- Infojagamine. Stand-up koosolek. Progressi jälgimine. Agiilne skoobihaldus.</p> <p>E 03.04.2017- Clean code. Koodistandard. Koodibaas. Versioonihaldussüsteemid.</p> <p>E 10.04.2017- Automatiseerimine.</p> <p>E 17.04.2017- The Wisdom of Crowds. Teenuse sisseostmine vs body leasing. Projekti maksumus.</p> <p>E 24.04.2017- Retrospektiiv.</p> <p>E 08.05.2017- Eksam.</p> |

| | |
|---------------------------|-----------------------------|
| Õppeainet kureeriv üksus: | Digitehnoloogiate Instituut |
| Kursuseprogrammi koostaja | Marek Kusmin |
| Allkiri: | |
| Kuupäev: | 15.01.2017 |