Tarkvara metoodikad ja mudelid.

1. Loetle tarkvara arenduse etapid

* Vajaduste kirjeldamine ja nende analüüs
* Tarkvaratoote disain
* Teostamine
* Testimine
* Toote väljalase
* Toote hooldus

1. Millises etapis tehtud vead on kõige kallimad kõrvaldada

Toote väljalase

1. Peale, millist etappi on mõistlik alustada programmeerimisega (koodi kirjutamisega)

Peale tarkvaratoote disaini

1. Tarkvara arenduses mõistetakse abstraktsiooni all …

Selle all mõeldakse lähenemisviisi, kus mingi nähtuse analüüsimisel jäetakse mängust välja selle ebaolulised küljed, mis ei mängi rolli antud probleemi lahendamise kontekstis.

1. Mis on avatud lähtekoodiga arenduse eelised?

Avatud lähtekoodiga arenduse eeliseks on see, et programm on üldjuhul vabalt kättesaadav, mugav rakendust luua vabalt ligipääsetavas keskkonnas. On lootust, et huvilised on valmis lõike omalt poolt täiendama ja kontrollima: Üldjuhul on õigus kasutada lõike juba valminud avatud koodiga rakenduste juurest

1. Mis on suletud lähtekoodiga arenduse eelised?

Suletud koodiga tarkvaraarenduse juures on lõppkasutajale kättesaadav vaid installeerimisvalmis toode, tarkvara loomisega seonduv jääb tarkvarafirma siseasjaks.

1. Mis on monumentaal ja agiilse arendus metoodika peamine erinevus?

Monumentaalsed metoodikad leiavad rakendust selliste projektide realiseerimisel, kus algusest peale on täpselt teada, mida tahetakse, millistele standarditele ja spetsifikatsioonidele tulemus vastab ja rajaneb ning mille korral lahenduskäik on selgelt teada ja kus (tavaliselt) on realiseerijaks suurem firma.

Agiilseid meetodeid kasutatakse juhul, kui arendajate grupp on väikesearvulisem, projekti lõpptulemus täpsustub töö käigus, koodi kirjutatakse (ja kirjutatakse ümber) praktiliselt projekti esimesest päevast peale.

1. Loetle süsteemiarenduse mudeleid (4 tk.)

* Koskmudel (waterfall model)
* Spiraalmudel (spiral model)
* Inkrementaalmudel (incremental model)
* Prototüüpimine (prototyping)

1. Monumentaal mudelite eelised agiilsete ees

Eeliseks on see, et algusest peale teatakse, mida tehakse. Süsteem ei muutu liigselt keerukaks.

1. Miks on dokumentatsioon oluline ja mida tuleb dokumenteerida?

Dokumentatsioon on oluline, et vältida vigu, täiendavat ajakulu, konflikte. Dokumentatsioon on vajalik ka selleks, et kui süsteemiga tegelev inimene vahetub, siis saaks ta ennast vajaliku infoga kurssi viia.

Dokumenteerida tuleks :

* defineerida projekti jaoks vajalikud komponendid (nõuete kataloog),
* IT võrgu struktuur
* opereerimise juhend