

შესაძლებლობათა სიმწიფის მოდელის შესახებ

ლოლიტა ბეჟანიშვილი
საქართველოს ტექნიკური უნივერსიტეტი

რეზიუმე

მომხმარებელთა პრეტენზიებმა პროგრამული პროდუქტების ხარისხზე გამოკვლევების წინა პლანზე წამოწიეს დამუშავებული სისტემის ხარისხის გარანტიები. შექმნილი სისტემის ხარისხი გამოცხადებული იქნა სისტემის თვისებად და უნდა გამხდარიყო მართვადი და დასაგეგმი. აღმოჩნდა, რომ პროდუქტის ხარისხი პირდაპირ დამოკიდებულია პროცესების ხარისხზე, რომელთა გამოყენებაც საჭიროა ამ პროდუქტის შესაქმნელად. ამ ფაქტის გააზრება გახდა ფუძემდებლური CMM/CMMI-ის მოდელის შექმნისათვის, რომლის დანიშნულებაც დამუშავების პროცესების სიმწიფის შეფასება და დამუშავების პროცესების ძირითადი ჯგუფების განსაზღვრა. საქართველოში CMM/CMMI-ის შესწავლა და მით უმეტეს სერტიფიცირება ჯერჯერობით არ ხორციელდება. ამიტომ აქტუალურია კოგნიტიური სისტემის შექმნა ორგანიზაციის შესაძლებლობათა შესაფასებლად, ასევე ორგანიზაციული კულტურის პრინციპების შესწავლის მიზნით.

საკვანძო სიტყვები: შესაძლებლობათა სიმწიფის მოდელი, CMM, ორგანიზაციული კულტურა, სიმწიფის დონე, საკვანძო არეები, საკვანძო პრაქტიკები.

1. შესავალი

80-ან წლებში სიტუაცია პროგრამული უზრუნველყოფის დარგში იყო შეფასებული როგორც “დაპროგრამების კრიზისი” იმ კატოსტროფული შეუსაბამობების გამო, რომლებიც გამოვლინდა პროგრამული სისტემების საჭიროებასა და მათზე მოთხოვნილ ხარისხს, ერთი მხრივ, და შეთუშავეების გეგმიურ და რეალურ ვადებსა და აგრეთვე რეალიზაციების ხარისხს შორის, მეორე მხრივ. გამოკვლევების წინა პლანზე წამოიწია სისტემის დამუშავების პროცესების ორგანიზაციული კულტურის გაუმჯობესების და ხარისხის გარანტიის კონცეფცია. მენეჯმენტის თეორია ხაზს უსვამს, რომ არსებობს პირდაპირი დამოკიდებულება პროდუქტის შექმნის პროცესების დონის და შედეგის ხარისხს შორის. მსოფლიოში იზრდება **Capability Maturity Model (CMM)** მოდელის პოპულარობა, რომლის დანიშნულებაც დამუშავების პროცესების სიმწიფის შეფასება.

ხარისხის უნივერსალური კონცეფცია Total Quality Management (TQM) ჩამოყალიბდა 50-ან წლებში იაპონიაში, ხოლო დასავლეთის ინდუსტრიაში მისი გავრცელება დაიწყო 80-ან წლებში. TQM ღალადებს ზოგად პრინციპებს, რომლების თანმიმდევრული დაცვა ეხმარება კომპანიებს ორგანიზაციული კულტურის ფორმირებაში, რაც უზრუნველყოფს ხარისხიანი პროდუქტების და სერვისების შექმნას. მაგრამ იმის გამო, რომ მოთხოვნები ხარისხის მართვის სისტემისადმი უნივერსალურია, საჭირო ხდება მათი გარკვეული ინტერპრეტაცია, როდესაც საქმე მიღის კონკრეტულ კომპანიებში დანერგვამდე.

მომხმარებელთა პრეტენზიებმა პროგრამული პროდუქტების ხარისხის მიმართ განაპირობეს ამ დარგისთვის ხარისხის მართვის სპეციფიური მოდელის შექმნის საჭიროება. პირველი დამკვეთის სახით მოგვევლინა აშშ-ს თავდაცვის სამინისტრო, რომელმაც ატეხა განგაში იმ პრობლემების გამო, რომლებიც წარმოიქმნა სამხედრო დანიშნულების პროგრამული სისტემების გამოყენების დროს. მოდელის დამუშავებას ხელი მოკიდეს Carnegie Mellon-ის უნივერსიტეტის Software Engineering Institute-ში და Mitre Corporation-ში, რის შედეგადაც 1991 წელს შეიქმნა პროგრამული უზრუნველყოფის დამუშავების პროცესების სიმწიფის მოდელი (Capability Maturity Model for Software, CMM-SW) პირველი ვერსია.

ხარისხის მენეჯმენტის ზოგადმა თეორიამ შემოიღო ე.წ. “ხარისხის სამკუთხედი”, რომლის წვეროებია ადამიანები, ტექნოლოგიები და პროცესები. ნათელია, რომ ხარისხის მიღწევისთვის საჭიროა კვალიფიციური სპეციალისტები და ეფექტური ინსტრუმენტარეზი, მაგრამ არც ერთი და არც მეორე არ იძლევა სასურველ შედეგს თუ არ იქნა გამართულად ორგანიზებული პროცესი – ბიჯთა თანამიმდევრობა, რომელთა შესრულებაც მიგვიყვანს დასმული მიზნის რეალიზაციამდე. პროდუქტის ხარისხი პირდაპირ დამოკიდებულია პროცესთა ხარისხზე, რომელთა გამოყენებაც საჭიროა ამ პროდუქტის შექმნისთვის.

ამ ფაქტის გააზრებამ ითამაშა ფუძემდებლური როლი CMM მოდელის შექმნისას – იმ მოდელისა, რომელიც განსაზღვრავს დამუშავების პროცესების ძირითად ჯგუფებს, აყალიბებს მახასიათებლებს ამ პროცესების სიმწიფის სხვადასხვა დონეებისათვის და იძლევა პრაქტიკულ რეკომენდაციებს პროცესების სრულყოფისათვის გარკვეული დონის მიღწევის მიზნით.

2. ძირითადი ნაწილი

CMM უზრუნველყოფს პროგრამული უზრუნველყოფის შექმნის პროცესის გაზომვის გარემოს (framework) და მისი ევოლუციურად სრულყოფის ორგანიზებას. CMM-ის ბაზაზე შეიქმნა პროგრამული უზრუნველყოფის მწარმოებელი კომპანიების შეფასების და სერტიფიცირების ინფრასტრუქტურა სათანადო სიმწიფის დონის მიღწევის თვალსაზრისით. სერტიფიცირების პროცესების სტიმულატორად გადაიქცა პენტაგონის მოთხოვნა მისთვის პროგრამული უზრუნველყოფის პოტენციურ შემქმნელებთან – მათ უნდა მიაღწიონ სიმწიფის დონის ფორმალურ შეფასებაში მინიმუმ მე-3 დონეს.

წარმოების ორგანიზაციის თვალსაზრისით მომზადებულ და ნაკლებად მომზადებულ ორგანიზაციებს შორის განსხვავებების დახარისხების მიზნით შეიქმნა პროგრამული უზრუნველყოფის შემუშავების პროცესების სტრუქტურა “მწიფე” ორგანიზაციისთვის. ეს სტრუქტურა გვიჩვენებს ევოლუციურ გზას შემთხვევითი პროცესების ქაოსიდან დისციპლინირებულ ტექნოლოგიამდე გაწონასწორებული და დოკუმენტირებული პროცესებით. სტრუქტურა დაფუძნებულია პროგრამული უზრუნველყოფის შექმნის პროცესების, შექმნის შესაძლებლობების, მეთოდების და საწარმოო კულტურის კონცეფციების გაერთიანებაზე.

"პროცესი არის ოპერაციათა სისტემა რაიმეს შექმნის დროს,... მოქმედებების, ცვლილებების ან ფუნქციების სერია, რომელიც მიგვიყვანს საჭირო შედეგამდე" – Webster-ის ლექსიკონი. "პროცესი არის ბიჯთა თანამიმდევრობა, რომელთა შესრულების შემდეგ მივაღწევთ სათანადო მიზანს" – IEEE-STD-610 სტანდარტი. პროგრამული უზრუნველყოფის შექმნის პროცესი შეიძლება იყოს განსაზღვრული როგორც ოპერაციების, მეთოდების, ხერხების და გარდაქმნების კრებული (და არა თანამიმდევრობა!), რომლებიც გამოიყენება პერსონალის მიერ პროგრამული უზრუნველყოფის და მასთან დაკავშირებული სხვა პროდუქტების დაპროექტებაში, დამუშავებაში და თანხლებაში. სხვა პროდუქტების როლში შესაძლო კანდიდატებია გეგმები (დამუშავების, ტესტირების, ვერიფიკაციის, სერტიფიკაციის, კონფიგურაციებით მართვის, ხარისხით მართვის,...), დოკუმენტაცია, ტესტები, პროგრამები და დამატებითი სასწავლო მასალა და სხვა.

პროცესების სიმწიფე არის მათი მართვალობის, რაოდენობით შეფასებათა შესაძლებლობის, გაკონტროლების და ეფექტიანობის ხარისხი. სიმწიფის ზრდა ნიშნავს პროცესების მგრადობის ზრდის პოტენციურ შესაძლებლობას და მიუთითებს პროგრამული უზრუნველყოფის შექმნის/თანხლების პროცესების ეფექტურობის და შეთანხმებულობის ხარისხზე მთელი ორგანიზაციის ფარგლებში. პროცესების რეალური გამოყენება შეუძლებელია მათი დოკუმენტირების და დაინტერესებული პერსონალის ინფორმირების გარეშე, მუდმივი კონტროლის და გაუმჯობესების გარეშე. კარგად მოფიქრებული პროცესების შესაძლებლობები სრულიად განსაზღვრულია.

ორგანიზაციებში, რომლებმაც მიაღწიეს სიმწიფეს, პროგრამული უზრუნველყოფის შექმნის/თანხლების პროცესები ღებულობენ სტანდარტის სტატუსს, ფიქსირდებიან ორგანიზაციულ სტრუქტურებში და განსაზღვრავენ საწარმოო ტაქტიკას და სტრატეგიას. კანონის სტატუსის მინიჭება აუცილებელს ხდის ინფრასტრუქტურის აშენებას და საჭირო კორპორატიული საწარმოო კულტურის შექმნას, რომელიც უზრუნველყოფს საქმის წარმოების შესაბამისი მეთოდების, ოპერაციების და პროცედურების მხარდაჭერას.

CMM-ის განვითარების და მისი გამოყენების არეების გაფართოების შედეგად 2001 წელს მოხდა მოდელის ახალი ვარიანტის გამოჩენა — Capability Maturity Model Integration (CMMI). ამ მოდელით ხდება პროცესების დონის სიმწიფის აღწერა არა მხოლოდ პროგრამული, არამედ სისტემური ინჟინერიის, პროდუქტების და პროცესების ინტეგრირებული დამუშავების და მიმწოდებლების შერჩევის კუთხით. ამჟამად SEI უკვე აღარ უჭერს მხარს CMM-ის მოდელს და კომპანიების სერტიფიკაცია ტარდება მხოლოდ CMMI-ში.

მოდელი გვთავაზობს დამუშავების პროცესის ხუთ სიმწიფის დონეს, რომლებიც ქმნიან ბაზისს იმისათვის, რომ თანამიმდევრულად გაუმჯობესდენ დამუშავების პროცესის

უმნიშვნელოვანესი მახასიათებლები. ისინი აყალიბებენ კონკრეტულ ორგანიზაციაში პროცესების სიმწიფის გაზომვის თავისებურ სკალას და ემსახურებიან ორგანიზაციის შესაძლებლობების შეფასებას.

სიმწიფის დონე (maturity level) – არის კარგად განსაზღვრული ევოლუციური საფეხური დამუშავების პროცესის სრულყოფის გზაზე. ყოველი დონე არის საბაზო შემდეგი დონის მისაღწევად. ამ მნიშვნელობით დონის გამოტოვებით გადახტომა, თუ საერთოდ შესაძლებელია, პრაქტიკულად არაეფექტურია და ამიტომ არ არის მიზანშეწონილი.

ყოველი დონე შეიცავს მიზნების ნაკრებს. მიზნის მიღწევა ნიშნავს დამუშავების მნიშვნელოვანი კომპონენტის (თვისების) გამოვლენას და მდგრად რეალიზაციას. დონის მიღწევა ნიშნავს დამუშავების პრაქტიკაში, რომელიც ტარდება ორგანიზაციაში, კომპონენტების ნაკრების დანერგვას, რაც გამოიწვევს ორგანიზაციის შესაძლებლობათა გაზრდას.

CMM/CMMI რეკომენდაციების მიხედვით კომპანიები აშენებენ მოთხოვნების მართვის და პროექტის დაგეგმვის, მონიტორინგის და პროექტის კონტროლის, ხარისხის და კონფიგურაციის მართვის, ტექნიკური გადაწყვეტილებების მიღების და სხვა პროცესებს. გარდა ამისა, ისინი ღებულობენ პროცესების სრულყოფის ინსტრუმენტს: პროცესების მახასიათებლების სხვადასხვა სიმწიფის დონეებზე განსაზღვრის შედეგად ასეთი მოდელები გვაძლევს ორგანიზაციის განვითარების კექტორს უფრო მაღალი დონეების მიღწევისათვის. საბოლოოდ სერტიფიცირებული კომპანიის პროდუქცია იქნება უფრო იაფი, ვიდრე ორგანიზაციისა, რომლის პროცესების დონე შეესაბამება პირველ დონეს, რათგანაც დამუშავების პროცესების ეფექტური ორგანიზება იძლევა შედეგის მიღწევის საშუალებას ნაკლები ძალების დახარჯვით.

CMMI სერტიფიკაცია საჭიროა კომპანიებისთვის, რომლებსაც აქვთ სურვილი თავისი პროდუქციის გამოტანის ამერიკულ ან დასავლურ ბაზარზე. რამდენიმე ათასზე მეტმა კომპანიამ მიიღო CMM/CMMI სერტიფიკატი, ოღონდ მე-5 დონე ჯერჯერობით რჩეულთა ზვედრია.

საქართველოში CMM/CMMI-ის სწავლება და სერტიფიცირება ჯერჯერობით ატარებს ეპიზოდურ ხასიათს. პრობლემას შეადგენს ის, რომ ოფიციალური სწავლების უფლებას ღებულობენ მხოლოდ ინსტრუქტორები, რომლებთაც აქვთ SEI-ის ავტორიზაცია და სერტიფიკაციისთვის საჭიროა მიმართვა სათანადო ორგანიზაციებში.

ამიტომ ძალიან აქტუალური ხდება კოგნიტიური სისტემის შექმნა ორგანიზაციის შესაძლებლობების შეფასებისთვის და ასევე ორგანიზაციული კულტურის პრინციპების შესწავლის მიზნით. ცოდნის შიდა წარმოდგენისთვის დამუშავდება კონტექსტურად მართვადი სემანტიკური ქსელებით წარმოდგენის მეთოდები და შესაბამისი აღწერები XML-ის ბაზაზე. ამ სისტემის საშუალებით შესაძლებელი გახდება ორგანიზაციის წინასწარი შეფასება სერტიფიკაციის საგანზე. ორგანიზაციები ასევე მიიღებენ რეკომენდაციებს ორგანიზაციული პროცესების გაუმჯობესებისთვის.

CMMI-ის სიმწიფის დონეები

განვიხილოთ დონეების მოკლე დახასიათება.

1) Initial (დაწყებითი)

პროცესების მკაფიო ორგანიზაცია არ არსებობს, პროცესები გაუთვალისწინებელნი არიან, არ არსებობს დამუშავების გარკვეული წესები და პროცედურები, პროცესები უნიკალურნი არიან, მათი რეალიზაცია ყველა დეტალში ხდება მხოლოდ კონკრეტულ დამუშავებაში (შემდეგი დამუშავება, თუ კი ბევრად მსგავსია, მაინც სრულდება სხვანაირად). ფიქსირებული მეთოდიკებით სრულდება დამუშავების პროცესის მხოლოდ პატარა რაოდენობის ელემენტარული მოქმედებები. დამუშავების წარმატება დამოკიდებულია შემსრულებლების ენთუზიაზმზე, მათ ფაქტიურ (ამ კონკრეტული პროექტის პირობებში) კვალიფიკაციაზე, შრომისმოყვარეობაზე და სხვა პროფესიონალურ და პირად თვისებებზე.

2) Repeatable (გამეორებითი)

დამუშავებულია თითოეული პროექტის საბაზო მართვის პროცესები ფაქტიური შეფასების, გეგმა-გრაფიკის და დამუშავების მიღებული ხარისხის შედეგის დაფიქსირების მიზნით. პროცესები განსაზღვრულნი არიან და ხდება მათი დოკუმენტირება. მიღწეულია პროცესის საჭირო დისციპლინა, რომელიც გვაძლევს დამუშავების ძირითადი მახასიათებლების გამეორების ნებას ახალ (მსგავს) პროექტებში.

3) Defined (განსაზღვრული)

პროგრამული უზრუნველყოფის შექმნის პროცესი როგორც მისი მართვის თვალსაზრისით, ასევე საწარმოო ტექნოლოგიის თვალსაზრისით დოკუმენტირებულია, სტანდარტიზებულია და ინტეგრირებულია სტანდარტულ (ამ ორგანიზაციისთვის) პროცესში. ნებისმიერ პროექტში გამოყენებულია პროგრამული უზრუნველყოფის დამუშავების და თანხლების პროცესის დამტკიცებული ვერსია.

4) Managed (მართვადი)

რეგულარულად იკრიბება პროგრამული უზრუნველყოფის შექმნის და მისი ხარისხის მახასიათებლების დეტალური გაზომვების შედეგები. როგორც უშუალოდ პროცესი, ასევე მისი მოღვაწეობის პროდუქტი შეისწავლება და მოწმდება მათი რაოდენობითი შეფასებების თვალსაზრისით. შესაძლებელია პროცესების წინასწარმეტყველება და ხდება შესაძლებელი ისეთი პარამეტრების მართვა, როგორიც არიან შეცდომების რაოდენობა, შრომატევადობა, გადამუშავების მოცულობა და ა.შ.

5) Optimizing (მაოპტიმიზირებელი)

შესაძლებელია პროგრამული უზრუნველყოფის შექმნის პროცესის მუდმივი და მიზანმიმართული გაუმჯობესება პროცესთან რაოდენობრივი უკუკავშირის ხარჯზე და ასევე ახალი იდეების და ტექნოლოგიების დანერგვის ხარჯზე. პროცესები იმყოფებიან მუდმივი გაუმჯობესების მდგომარეობაში, კომპანიას შეუძლია მნიშვნელოვანი ინოვაციების დანერგვა პროცესებში რაოდენობითი მაჩვენებლების ანალიზის საფუძველზე, პროექტების პრობლემების ძირეული მიზეზების იდენტიფიცირება და მათი გავლენისაგან თავის არიდება.

სიმწიფის დონების შიდა სტრუქტურა

თითოეული სიმწიფის დონე შედგება რამდენიმე საკვანძო არისგან (key process areas). თითოეული საკვანძო არე შეიცავს საკვანძო პრაქტიკების (key practices) 5 ჯგუფს (ჯგუფებს ეწოდება common features – საერთო მახასიათებლები). ყველა საკვანძო პრაქტიკის განხორციელება მოცემულ საკვანძო არისთვის უზრუნველყოფს ამ არის მიზნების (goals) მიღწევას. ქვემოთ ეს ცნებები განიხილება უფრო დაწვრილებით.

საკვანძო არეები (Key Areas)

ეს არის ურთიერთდამოკიდებული მოქმედებების (საკვანძო პრაქტიკების) კრებული, რომლების ერთობლივი შესრულებაც მიგვიყვანს მიზნების მიღწევამდე მოცემულ არეში.

თითოეული სიმწიფის დონე (გარდა პირველისა, რომელიც არის გამონაკლისი) შეიცავს თავის საკვანძო არეების ნაკრებს. თუ პროექტების მიხედვით, რომლებიც სრულდებიან რომელიმე ორგანიზაციაში, მიღწეულია მოცემული საკვანძო არის მიზნები, ეს ნიშნავს: დამუშავების პროცესის შესაძლებლობები, რომლებიც დამახასიათებელი არიან მოცემული საკვანძო არისათვის, მიღწეულია მოცემულ ორგანიზაციაში. თუ ასეთი დონე მიღწეულია მოცემული დონის ყველა საკვანძო არეებისთვის, ეს ნიშნავს: პროცესი (ორგანიზაცია) შეასაბამება ამ სიმწიფის დონეს.

საკვანძო არეები გამოყოფილია იმიტომ, რომ მათთან დაკავშირებული პრობლემების ამოხსნა უფრო ეფექტურად გააუმჯობესებს პროგრამული უზრუნველყოფის შექმნის პროცესს. თავად ეს პრობლემები წარმოადგენენ მოთხოვნებს სიმწიფის დონის მისაღწევად. რომ “დაფვაროთ” საკვანძო არე საჭიროა ყველა მისი მიზნის მიღწევა.

საკვანძო პრაქტიკები, რომლებიც შეასაბამებიან მოცემულ საკვანძო არეს, მდოვრედ იცვლიან სახეს უფრო მაღალი დონეების მიღწევის პროცესში. მაგალითად, Integrated Software Management (დონე 3) მიღებულია Software Project Planning და Software Project Tracking and Oversight-დან (დონე 2) მე-2 დონედან მე-3 დონემდე ევოლუციის გზით.

საერთო თვისებები (Common Features)

ზოგიერთი საკვანძო არეები შეიცავენ ათობით საკვანძო პრაქტიკას. მუშაობა ასეთი დიდი რაოდენობის პრაქტიკებთან მოუხერხებელია. მოხერხებულობისთვის ყველა საკვანძო პრაქტიკა, რომელიც ეკუთვნის თითოეულ საკვანძო არეს, დაყოფილია 5 ტიპურ ჯგუფად (common feature).

ვალდებულებები შესრულებაზე (Commitment to perform)

მოქმედებები, რომლებიც უნდა იყვნენ წამოწყებული იმისათვის, რომ უზრუნველყონ პროგრამული უზრუნველყოფის შექმნის პროცესის, დაყენების და შემდგომი მხარდაჭერის

გარანტია. ჩვეულებრივ, ასეთი მოქმედებები შეიცავენ ორგანიზაციული პოლიტიკის შემუშავებას და ხელმძღვანელობის თავდებობას.

შესრულების შესაძლებლობა (Ability to perform)

წინაპირობები, რომლებიც უნდა არსებობდნენ იმისათვის, რომ პროცესის რეალიზება შესაძლებელი იყოს. იგულისხმება, რესურსები, ორგანიზაციული სტრუქტურები, პერსონალის სწავლა.

შესასრულებელი მოქმედებები (Activities performed)

მოქმედებები, რომლებიც არიან საჭირო საკვანძო არის განსახიერების ("დაფარვის") მიზნით: გეგმების მომზადება, სამუშაოების შესრულება, მათთვის თვალისდევნება, საჭირო კორექტივების განხორციელება. ეს მოქმედებები ძირითადად კატეგორიას განეკუთვნებიან. ზუსტად ისინი ფარავენ საკვანძო არეს და უზრუნველყოფენ კოლექტივის/პროცესის შესაძლებლობების გაფართოებას. სხვა ჯგუფების საკვანძო პრაქტიკები (Commitment to Perform, Ability to Perform, Measurement and Analysis, Verifying Implementation) მხოლოდ ქმნიან ბაზის Activities Performed-თვის.

გაზომვები და ანალიზი (Measurement and Analysis)

პროგრამული უზრუნველყოფის შექმნის პროცესის გაზომვების შესრულება და მათი შედეგების ანალიზი. ეს აუცილებელია იმის გასაგებად, თუ როგორ არიან შესრულებული საკვანძო არის განსახიერებისთვის საჭირო მოქმედებები და როგორია ამის ეფექტი.

შესამოწმებელი რეალიზაცია (Verifying Implementation)

ბიჯები, რომლებიც სრულდებიან იმისათვის, რომ დაერწმუნდეთ: პროგრამული უზრუნველყოფის შექმნის პროცესის მოქმედებები სწორედ სრულდებიან. ჩვეულებრივ, ამისათვის ხელმძღვანელობა და პროგრამული პროდუქტების ხარისხის სამსახური (Software Quality Assurance - SQA) აწვებენ ინსპექციებს და შემოწმებებს.

საკვანძო პრაქტიკები (Key Practices)

თითოეული საკვანძო არე განსაზღვრულია საკვანძო პრაქტიკების ტერმინებში, რომლებიც აუცილებელია არიან ამ საკვანძო არის მიზნების მისაღწევად. საკვანძო პრაქტიკები განსაზღვრავენ ინფრასტრუქტურებს და მოქმედებებს.

საკვანძო პრაქტიკები განსაზღვრავენ რა უნდა იყოს გაკეთებული, მაგრამ არ არიან გადღებული განსაზღვრონ როგორ უნდა მოხდეს საკვანძო არის მიზნების მიღწევა. მიზნის მიღწევისთვის შეიძლება იყვნენ გამოყენებული ალტერნატიული პრაქტიკები. ამ შემთხვევაში მიზნის მიღწევის გზის შერჩევის საკითხი უნდა იყოს გადაწყვეტილი საღი აზრის საფუძველზე.

3. დასკვნები

კოგნიტიური სისტემა CMM მოდელის ბაზაზე შეიძლება იყოს გამოყენებული შემდეგი 4-ი კატეგორიის სპეციალისტების მიერ:

- ორგანიზაციის შეფასების ჯგუფი (Assesments Team) იყენებს CMM-ს ორგანიზაციის მაგარი და სუსტი მხარეების აღმოჩენისთვის;
- კონტრაგენტების შეფასების ჯგუფი (Evaluation Team) იყენებს CMM-ს კონტრაგენტების შერჩევისას რისკის გამოვლენისთვის და კონტრაქტებისთვის თვალისდევნისთვის;
- ხელმძღვანელები და ტექნიკური პერსონალი იყენებს CMM-ს პროცესის სრულყოფის პროგრამის დაგეგმის და ცხოვრებაში დანერგვის საქმიანობის გასააზრებლად;
- პროცესის გაუმჯობესების ჯგუფი იყენებს CMM-ს, როგორც თავისი მოქმედებების სახელმძღვანელოს.

4. ლიტერატურა.

1. Technical Report CMU/SEI-87-TR-23. Preliminary Version. September 1987. "A Method for Assesing the Software Engineering Capability of Contractor". W.S.Humphrey, W.L.Sweet. Software Engineering Institute. Carnegie Mellon University. Pitsburg, Pensilvania 15213
2. Rubin, Howard. "Software Process Maturity". *American Programmer*, January, 1991.
3. Humphrey, Watts. *Managing The Software Process*. Reading, MA: Addison-Wesley, 1989

4. Pressman, Roger. *Making Software Engineering Happen*. Englewood Cliffs, NJ: Prentice Hall, 1988.
5. Yourdon, Edward. *Decline & Fall of the American Programmer*. Englewood Cliffs, New Jersey: YOURDON PRESS, Prentice Hall, 1993
6. IFPUG (International Function Points User Group), *Counting Practices Manual Release 3.3*, 1992. 5008 Pine Creek, Westerville, Ohio.
7. Jones, T.C., *Applied Software Measurement*, McGraw-Hill, 1991.
8. Dreger, J.B., *Function Point Analysis*, Prentice-Hall, 1989.
9. Chrissis, Mary Beth; Konrad, Mike, Shrum, Sandy. *CMMI : Guidelines for Process Integration and Product Improvement*. Addison-Wesley Professional, 2003.
10. Kulpa, Margaret K.; Kent A. Johnson. *Interpreting the CMMI : A Process Improvement Approach*. Auerbach Publications, 2003.
- 11..16. სტანდარტების სერია ISO 9000.
17. Perry, William E., *Effective Methods of EDP Quality Assurance*. Q.E.D. Information Services, 1981.
18. Schulmeyer, G. Gordon, McManus, James I., *Handbook of Software Quality Assurance*. Van Nostrand Reinhold Company, 1987.
19. ANSI/IEEE Std 730-1984, Software Quality Assurance Plans.
20. ANSI/IEEE Std 983-1986, Software Quality Assurance Planning.
21. ANSI/IEEE Std 1012-1986, Software Verification and Validation Plans.
22. ANSI/IEEE Std 1028-1988, Standard for Software Reviews and Audits.

О МОДЕЛИ ЗРЕЛОСТИ ВОЗМОЖНОСТЕЙ

Бежанишвили Л.

Грузинский Технический Университет

Резюме

Претензии потребителей к качеству программных продуктов выдвинули на передний план исследований гарантии качества программных продуктов. Качество разрабатываемой системы объявлялось свойством системы и должно было стать планируемым и управляемым. Выяснилось, что качество продукта находится в прямой зависимости от качества процессов, которые используются для его создания. Понимание этого факта стало основополагающим при создании модели СММ/СММИ, которая предназначена для оценки зрелости процессов разработки и определения основных групп процессов разработки. В Грузии изучение, и тем более сертификация, по СММ/СММИ пока относится к малоизвестной категории. Поэтому очень актуальна проблема создания когнитивной системы для оценки возможностей организации, а также обучения принципам организационной культуры. Организации также получают рекомендации по улучшению организационных процессов.

ABOUT CAPABILITY MATURITY MODEL

Bejanishvili Lolita

Georgian Technical University

Summary

User's claim pretension to the programming products quality brings to the investigations forefront the insurance arrangements of the programm products quality. A quality of the workable system is announced as system property and should be planned scheduled and managed. It came to light that the product quality is in direct dependence of a quality of the process, which is used for its creation. Understanding this fact stands fundamental for the creation CMM/CMMI model, which is assigned for the maturity estimation of the working process. In Georgia learning and certification on CMM/CMMI have episodic character yet. Therefore the cognitive system creation is very important for estimation of the organization's abilities and also for learning organizational culture. An organization takes also the recommendation for the organizational culture improvement.