

**ტენდერი აპლიკაციის მენეჯმენტ/მონიტორინგის სისტემის ე.წ. APM(****Application Performance Monitoring)-ის შეძენაზე**

# **ტერმინოლოგია**

დოკუმენტში გამოყენებული ტერმინების განმარტება:

ბიზნეს ტრანზაქცია: არის სისტემური, Software ტრანზაქცია. იგი თავის თავში აერთიანებს ისეთ Software სპეციფიურ პროცესებს როგორებიცაა მაგ,: Call, request და ა.შ.

Baseline: ნიშნავს სისტემის, მისი კომპონენტების ან მეტრიკების ზღვრულ მაჩვენებლებს, რომლებიც გამოიყენება მათი ნორმალური ქცევის შეფასების კრიტერიუმად, გადახრებისა და ანომალიების იდენტიფიცირებისათვის.

# **ზოგადი პირობები**

VTB ბანკი „ჯორჯია“ ინტერნეტბანკის, ქორბენკინგის და შიდა სერვისების მონიტორინგისათვის აცხადებს ტენდერს APM(Application Performance Monitoring) გადაწყვეტილების შესაძენად.

# **ტექნიკური მოთხოვნები APM მონიტორინგის სისტემის მიმართ**

1. მონიტორინგის სისტემა თავსებადი უნდა იყოს შემდეგ ოპერაციულ სისტემებთან:
	1. Microsoft Windows Server 2012 R2;
	2. Microsoft Windows Server 2016;
	3. Microsoft Windows Server 2019;
	4. RedHat Enterprise Linux 7+ (64 bit);
	5. CentOS 7 (64 bit);
	6. Ubuntu 14+;
2. მონიტორინგის სისტემა თავსებადი უნდა იყოს POP3 და IMP პროტოკოლებთან;
3. მონიტორინგის სისტემას უნდა გააჩნდეს API სხვა სისტემებთან ინტეგრაციისათვის
4. მონიტორინგის სისტემას უნდა გააჩნდეს VMware Vsphere და Microsoft Hyper-V ვირტუალურ გარემუოში მუშაობის მხარდაჭერა;
5. მონიტორინგის სისტემას უნდა შეეძლოს კორექტულად დაასრულოს ან აღადგინოს ქსელური სესიები, ქსელში მომხდარი წყვეტების შემთხვევაში;
6. მონიტორინგის სისტემას უნდა შეეძლოს, მისი კომპონენტებისა და საინფორმაციო ნაკადების ლოგირება, როგორც ლოკალურ ისე ცენტრალიზირებულ ლოგირების სანახ ჟურნალებში;
7. მოთხოვნები ინფორმაციული უსაფრთხოების მიმართ
	1. მონიტორინგის სისტემას უნდა გააჩნდეს Microsoft Active Directory-სთან ინტეგრაციის მხარდაჭერა სტანდარტული საშუალებებით, რათა დაადგინოს მოხმარებლებს შესახებ ინფორმაცია, მათ შორის ჯგუფების ინფორმაცია;
	2. მონიტორინგის სისტემას უნდა გააჩნდეს, სისტემის მომხმარებლების შამოწმების ფუნქციონალი Active Directory-ის საშუალებით, თუ რომელ ჯგუფს მიეკუთვნებიან ისინი და ჯგუფის შესაბამისი უფლებების განსაზღვრა, მათ შორის სისტემის ცალკეულ ფუნქციებზე ან ინტერფეისზე წვდომა.
	3. მონიტორინგის სისტემას უნდა გააჩნდეს აუტენტიფიკაციის მექანიზმი დომენური აუტინტიფიკატორების გამოყენების მეშვეობით. ამასთანავე AD-ს მომხმარებლის დაბლოკვის ან წაშლის შემთხვავში აღნიშნულ მომხმარებელს აღარ უნდა მიეცეს მონიტორინგის სისტემასთან და მის ინტერფეისთან წვდომის საშუალება;
	4. მონიტორინგის სისტემას უნდა გააჩნდეს დაერთებების კონფიგურაცია SAML-ის მეშვეობით. ასევე სისტემას უნდა შეეძლეოს დაშვების სისტემებთან ინტეგრირება რათა მოიპოვოს ინფორმაცია მომხმარებლებს შესახებ, მათ შორის როლების და უფლებების;
	5. მონიტორინგის სისტემას უნდა შეეძლოს მომხმარებლის სესიის ავტომატურად დასრულება მისი უმოქმედობის შემთხვევაში (ე.წ. Time out);
	6. მონიტორინგის სისტემასთან მუშაობა უნდა ხორციელდებოდეს როგორც დაკრიფტული (HTTPS) ასევე დაუკრიპტავი (HTTP) არხების საშუალებით, როგორც გრაფიკული ვებ ინტერფეისთან ასევე ქსელურ დონეზე, სერვისების კომუნიკაცია;
	7. მონიტორინგის სიტემას უნდა გააჩნდეს ადვილად კონფიგურირებადი Role Based მოდელი თითოეული კომპონენტისთვის, მათ შორის წვდომები ტრანზაქციებთან, ბრაუზერების სესიებთან, ანალიტიკასთან, დეშბორდებთან და რეპორტებთან;
	8. დაუშვებელია სისტემაზე ადმინისტრაციული წვდომის (ე.წ. Root/sudo) გამოყენება გადაწყვეტილების თუ მისი კომპონენტებისათვის მონიტორინგის საშუალების უზრუნველსაყოფად;
8. მონიტორინგის ობიექტებზე მაქსიმალური Overhead (სისტემაზე დატვირთვა) არ უნდა აღემატებოდეს 3%-ს;
9. მონიტორინგის სისტემას უნდა გააჩნდეს მართვის სისტემასა და აგენტებს შორის აუნტენტიფიკაციის მხარდაჭერა;
10. მონიტორინგის სისტემის კონფიგუყრაცია და მართვა უნდა ხორციელდებოდეს WEB ინტერფეისის მეშვეობით;
11. მონიტორინგის სიტემას უნდა გააჩნდეს კონტეინერების შიგნით მონიტორინგის განხორციელების საშუალება, აპლიკაციის მიკროსერვისული არქიტექრურის შემთხვევაში;
12. მონიტორინგის სისტემას უდა გააჩნდეს Java, .Net, .Net Core, Python, PHP, Node.JS, აპლიკაციების წარმადობის მონიტორინგის საშულება;
13. მონიტორინგის სისტემას უნდა გააჩნდეს აპლიკაციაში მომხდარი შეცდომებისა და გამონაკლისების გამოვლენის და შეტყობინენის ფუნქციონალი
14. მონიტორინგის სისტემას უნდა შეეძლოს ძირითადი აპარატურული მაჩვენებლების როგორებიცაა CPU-ს დატვირთვა, პროგრამული უზრუნველყოფის მეხსიერის გამოყენება/დატვირთვა, დისკურია და ქსელური I/O-ს მაჩვენებლების გაზომვისა და აღნიშნული აპარატურული პარამეტრებისათვის ე.წ. Baseline-ბის ავტომატური განსაზღვრა არანაკლებ ერთი თვის პერიოდით;
15. მონიტორინგის სისტემას უნდა შეეძლოს ახალი კომპონენტებისათვის ე.წ. Baseline-ბის ავტომატური განსაზღვრა ადამინის სიტემაში ჩარევის გარეშე, უნდა გააჩნდეს ე.წ. Alarm Storm-ბის თავიდან აცლიები მექანიზმები;
16. მონიტორინგის სისტემას უნდა შეეძლოს პროდუქტის API-ში, აპლიკაციის აგენტის მეშვეობით გამოვლენილი ტრანზაქციებიდან, მონაცემების და მეტრიკებისა მისაღებად SQL მსგავსი მოთხოვნების ფორმირება;
17. მონიტორინგის სისტემას უნდა გააჩნდეს მონიტორინგის ობიექტიდან log ფაილების შეგროვების და შენახვის საშუალება;
18. მონიტორინგის სისტემას ავტუმატურად უნდა შეეძლოს ნებისმიერი ბიზნეს ტრანზაქციისათვის მეტრიკისათვის დინამიური ზღვრული მაჩვენებლების განსაზღვრა არანაკლებ ერთი თვის პერიოდით;
19. მონიტორინგის სისტემას უნდა გააჩნდეს აპლიკაციის რელიზების შედარების ფუნქციონალი. შესაძლებელი უნდა იყოს ე.წ. Pre-Production სტადიაზე აპლიკაციის ახალი ვერსიის კონვერსიის მაჩვენებლის ანალიზი და შედარების შესაძლებლობა;
20. მონიტორინგის სისტემას უნდა ქონდეს როგორც On-premis ასევე SaaS იმპლემენტაცის შესაძლებლობა;
21. დასამონიტორინგებელ სისტემაზე მონიტორინგის სისტემის აგენტის ინსტალაციის დროს სასურველია არ ჭირდებოდეს სისტემაზე ადმინისტრაციული წვდომის (ე.წ. Root/Sudo) უფლება;
22. მონიტორინგის სისტემისას უნდა ქონდეს სრული On-premise ინსატალციის საშუალება სრულიად იზოლირებულ გარემოში, ინტერნეტთან წვდომის გარეშე
23. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად აღმოაჩინოს და გამოსახოს აპლიკაციის არქიტექტურა და ტოპოლოგია მისი ყველა კომპონენტის მათ შორის ტრანზაქციების ჩათვლით;
24. მონიტორინგის სისტემას კოდში ჩარევის გარეშე, ავტომატურდ უნდა შეეძლოს კლასების და მეთოდების გამოძახებების ხის აგება/ვიზუალიზაცია;
25. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად აღმოაჩინოს ე.წ. request-ები, მაგალითად სისტემაში შესვლა-გამოსვლა, თანხების გადარიცხვები, ბალანსის შემოწმება და ა.შ;
26. ანალიზი და მონაცემთა დამუშავება უნდა ხორციელდებოდეს მონიტორინგის სისტემის სერვერის მხარეს რათა არ მოხდეს მონიტორინგის ობიერქტების რესურსების დატვირთვა/გამოყენება;
27. მონიტორინგის სისტემას უნდა შეეძლოს ნელი და შეფერხებული (დაკიდებული) ბიზნეს ოპერაციების განსაზღვრა ავტომატურად, ადამიანის ჩარევის გარეშე;
28. მონიტორინგის სისტემას უნდა გააჩნდეს მონიტორინგის ობიექტებიდან ბიზნეს ტრანზაქციების ანალიტიკური მონაცემების შეგროვების და შენახვის ფუნქციონალი;
29. მონიტორინგის სისტემას სასურველია შეეძლოს ბიზნეს ტრანზაქციების ფილტრაცია ნებისმიერი მაჩვენებლით;
30. მონიტორინგის სისტემას სასურველია ქონდეს კონკრეტული მომხმარებლის ნებისმიერი ტრანზაქციის დროის კონკრეტულ მონაკვეთებში განსაზღვრის საშუალება;
31. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად განსაზღვროს ზღვრული მაჩვენებლები ბიზნეს ტრანზაქციის ნებისმიერი შიდა კომპონენტისთვის. იმავდროულად მას უნდა შეეძლოს განსაზღვროს არა მხოლოდ ის თუ რომელი ბიზნეს ტრანზაქცია არის ნელი, არამედ მისი რომელი კომპონენტი არის ნორმიდან გადახრილი;
32. მონიტორინგის სისტემას უნდა იძლეოდეს მოქნილი ე.წ. Health Rule-ების შექმნის საშუალებას, როგორც ტექნიკური ასევე ბისზნეს ტრანზაქციისათვის, რაც სისტემას საშუალებას მისცემს დროულად მოახდინოს შეტყობინება წარმოქმნილ ანომალებზე ავტომატურად შექმნილი ზღვრული მაჩვენებლებიდან ე.წ. Baseline-ებიდან გადახრის შემთხვევაში დროის რეალურ ე.წ. Real Time რეჟიმში;
33. მონიტორინგის სისტემა სასურველია იძლეოდეს დეშბორდების შექმნის და კონფიგურირების საშუალებას ბიზნეს მეტრიკებისა და აპლიკაციის წარმადობის კორელაციის ვიზუალიზაციისათვის, მათი მდოგომარეობის (Health) სტატუსის მითითებით;
34. მონიტორინგის სისტემას უნდა შეეძლოს ბიზნეს ტრანზაქციების ინფრასტრუქტურის მონიტორინგთან კოლერაცია (OC JMX და ა.შ);
35. მონიტორინგის სისტემას უნდა შეეძლოს აპლიკაციის ე.წ. "ცხელი წერტილების" იდენტიფიცირება (უნდა შეეძლოს პრობლემურ, არა წარმად და ნელ ბიზნეს ტრანზაქციაში ყველაზე გრძელი მეთოდის სწრაფად განსაზღვრა/გამოვლენა);
36. მონიტორინგის სისტემას უნდა შეეძლოს ყველა (და არა არჩევითად) ტრანზაქციის გაზომვა შეფასება და იდენტიფიცირება;
37. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად განსაზღვროს არასწორი ტრანზაქცია, ხელით ჩარევის გარეშე;
38. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად განსაზღვროს ნელის SQL მიმართვები, ხელით ჩარევის გარეშე;
39. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად განსაზღვროს ნელის სერვერული სისტემები და გარე სერვისები, ხელით ჩარევის გარეშე;
40. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად გაანალიზოს სრული მონაცემები პრობლემის ე.წ. Root cause ის აღმოჩენის უზრუნველსაყოფად;
41. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად აღმოაჩინოს ბიზნეს ტრანზაქციის კლასები და მეთოდები დამატებითი კონფიგურირების გარეშე;
42. მონიტორინგის სისტემას უნდა შეეძლოს ავტომატურად აღმოაჩინოს კოდის ბლოკირებები;
43. მონიტორინგის სისტემას უნდა შეეძლოს ხარვეზებით მომუშავე ბიზნეს ტრანზაქციის კოდის დონეზე დიაგნოსტიკის წარმოება (კლასის და მეთოდის დონეზე ვიზუალიზაცია);
44. მონიტორინგის სისტემას სასურველია ქონდეს ბიზნეს ტრანზაქციებისა და ბიზნეს მაჩვენებლების ავტომატური ამოღების საშუალება კოდში დამტებითი ჩარევის გარეშე;
45. მომავალში მხოლოდ ლიცენზიების დამატების შემთხვევაში, მონიტორინგის სისტემას უნდა ქონდეს სინთეტიკური მონიტორინგის განხორციელების ფუნქციონალი:
46. მარტივი HTTP/HTTPS მიმართვების შესრულება
47. რთული მიმართვების განხორციელება რეალური, მომხმარებლის ქცევის/ქმედების ემულაცია;
48. სინთეტიკური ტესტების განხორციელების ავტომატიზაციის ფუნქციონალი;
49. თითოეული სინთეტიკური ტესტების ანალიზის შესაძლებლობა
50. ტესტირების სტანდარტების გამოყენება როგორიცაა Selenium- სრულფასოვანი საწარმოო სტანდარტი, სინთეტიკური ტესტირების სცენარებისათვის. სისტემას უნდა გააჩნდეს ნებისმიერ ინსტურუმენტთან თავსებადობა რომელიც თავსებადია Selenium ის სცენარებთან (მაგ. Katalon, Selenium IDE, WebDriver)
51. წესების დარღვევის და ნორმალური მდგომარიებიდან გადახრის თრიგერების აღმოჩენის შემთვევაში მონიტორინგის სისტემას უნდა ქონდეს გაფრთხილების ან შეტყობიმნების ფუნქციონალი (ელექტორნული ფოსტა, SMS, Http-მიმართვა, ე.წ. Custom ქმდება);
52. მონიტორინგის სისტემას უნდა ქონდეს ქვესისტემა რომელსაც შეუძლია სხვადასხვა ქმედებების და ინციდნეტების კორელირება რის საფუძველზეც სისტემა აგზავნის შეტყობინებებს უკვე ინდეტიფიცირებული კონკრეტული Root Cause-ის შესახებ და არა მრავალ შეტყობინებას ერთი პრობლემის ირგვლივ;
53. მონიტორინგის სისტემას უნდა ქონდეს ფუნქციონალი გაუგზავნოს შეტყობინებები წინასწარ განსაზღვრულ/დადგენილ ჯგუფებს;
54. მონიტორინგის სისტემას უნდა ქონდეს შეტყობინებების შაბლონების შექმნის საშუალება წინასწარ დადგენილი წესების მიხედვით რეაგირების შესაძლებლობით (ინფრასტრუქტურა, მონაცემთა ბაზები, აპლიკაცია, ბიზნეს მეტრიკა);
55. მონიტორინგის სისტემას უნდა იძლეოდეს აპლიკაციისა და შეერთებების (Connections) ვიზუალიზირების საშუალებას და ასევე უნდა ქონდეს მათი ურთიერთ დამოკიდებულების ვიზუალიზაციის ფუნქციონალი;
56. აპლიკაციის ტოპოლოგიის ვიზუალიზაციიასთან წვდომა შესაძლებელი უნდა იყოს შეზღუდული ან/და განსაზღვრული მომხმარებლის როლების და უფლებების საფუძველზე;
57. მონიტორინგის სისტემას უნდა შეეძლოს ქსელური დაერთებების ვიზუალურად ასახვა;
58. მონიტორინგის სისტემას უნდა შეეძლოს ქსელური ტოპოლოგიის და ურთიერთ დამოკიდებუკლებების განსაზღვრა;
59. მონიტორინგის სისტემას უნდა შეეძლოს ჰიბრიდულ ღრუბლოვან გარემოში ქსელური რესურსების აღმოჩენა და დასურათება;
60. მონიტორინგის სისტემას უნდა შეეძლოს ქსელში პრობლემების ავტომატური აღმოჩენა (latency, Throughput, tcp-loss და სხვა მეტრიკებით);
61. მონიტორინგის სისტემას უნდა შეეძლოს წარმადობის ანომალიების ავტომატური აღმოჩენა და შესწავლა აპლიკაციის ქცევის საბაზო პარამეტრების შესწავლის გზით სასურველია 1 თვის მონაცმების ანალიზის საფუძველზე;
62. მონიტორინგის სისტემას უნდა შეეძლოს ძირითადი აპარატურული მახასაითებლების როგორებიცაა დისკი, მეხსიერება, პროცესორი და დამუშავების გვერდის ხარვეზების მონიტორინგი;
63. სისტემას სასურველია ქონდეს ხელოვნური ინტელექტის ეჯინი AI პრობლემების უკეთ გაანალიზების შესაძლებლობისთვის
64. სასურველია სისტემას შეეძლოს შეეძლოს სესიის განმეორება პრობლემების უკეთ გაანალიზების შესაძლებლობისთვის
65. მონიტორინგის სისტემას უნდა შეეძლოს სერვერული პროცესების და მასთან დაკვაშირებული მეტრიკების მონიტორინგი;
66. მონიტორინგის სისტემის სერვერის წვდომადობის მონიტორინგი:
	1. პროცესორის დატვირთვის მონიტორინგი
	2. დისკის წარმადობის მონიტორინგი
	3. დისკის გამოყენების მოცულობის მონიტორინგი
	4. სერვერის დატვირთვის მონიტორინგი
	5. მეხსიერების ბარათის მონიტორინგი
	6. SWAP-ის მონიტორინგი
	7. ქსელური ადაპტერების მონიტორინგი
	8. ინფრასტრუქტურის მეტრიკების და კონკრეტული აპლიკაციის წარმადობის კორელაცია;
	9. კრიტიკული ივენთების საპასუხოდ სკრიპტების კონფიგურირების და გაშვების შესაძლებლობა;
	10. დისკებზე მაღალი დატვირთვების (წაკითხვა ჩაწერა) შემთხვევაში შეტყობინების გაგზავნის ფუნქციონალი;
	11. სერვერის მონიტორინგის ნორმის წესების ავტომატური განსაზღვრა/აგება
	12. შიდა და გარე სერვისების მონიტორინგი HTTP და HTTPS საშულებით;
67. მონიტორინგის სისტემას უნდა გააჩნდეს მონაცემთა ბაზების მონიტორინგის შესაძლებლობა(სასურველია აგენტის დაყენების გარეშე):
	1. მონაცემთა ბაზის წარმადობის განსაზღვრა რეალურ დროში, მონაცემთა ბაზის PostgresSQL, Sybase, DB2, MS-SQL, My-SQL, Oracle, Mongo DB მიმდინარე აქტიურობის ჩვენებით უშაულოდ მონაცემთა ბაზაზე;
	2. PostgresSQL, Sybase, DB2, MS-SQL, My-SQL, Oracle, Mongo DB ბაზების მონიტორინგის მხარდაჭრა;
	3. ბლოკირების ანალიზის განხორციელების ფუნქციონალი;
	4. მონაცემთა ბაზების დროში განხორციელებული აქტივობის პროფილის შესახებ შეტყობინების ფუნქციონალი (შაბლონების იდენტიფიცირება);
	5. მონაცემთა ბაზების ეგზემპლარების დონის სტატისტიკის შეგროვების და შენახვის ფუნქციონალი (სქემების, ცხრილების, სტრიქონების რაოდენობების, ინდექსების);
	6. მონაცემთა ბაზის/Host-ების წარმადობის ძირითადი მაჩვენებლების შეგროვება და შენახვა (პროცესორი, მეხსიერება და ა.შ.);
	7. SQL-განმარტების და განხორციელების გეგმების შეგროვება;
	8. მომხმარებლის სისტემაში შესვლის შესახებ იმფორმაციის შეგროვება / იდენტიფიცირება;
	9. მონაცემთა ბაზის ობიექტებზე ინფორმაციის შეგროვება, როგორც სტანდარტული ასევე ე.წ. Custome ინფორმაციის;
	10. ავტომატურად განსაზღვროს ნელა მომუშავე მონაცემთა ბაზის მიმართვები, მოახდინოს მათი პირველწყაროს ე.წ. Root Cause-ის იდენთიფიცირება, და აჩვენოს აღმოჩენილი ნელი ტრანზაქციების გავლენა აპლიკაციაზე;
68. Web ბრაუზერის მონიტორინგი გათვლილი უნდა იყო არანაკლებ 10 მილიონ ნახვაზე წელიწდში.
69. სისტემას უნდა გააჩნდეს ბრაუზერში მომხმარებლის სესიის მონიტოინგის შემდეგი ფუნქციონალის მხარდაჭერა:
	1. გვერდის ყველა მოთხოვნის/მიმართვის მონიტორინგი
	2. ყველა AJAX მოთხოვნის/მიმართვის მონიტოინგი
	3. iFrame მოთხოვნის/მიმართვის მონიტოინგი
	4. JavaScript-ის შეცდომების ავტომატური აღმოჩენა
	5. ანალიტიკის განხორციელება მოხმარების საფუძველზე, მოწყობილობის ტიპების და ოპერაციულ სისტემების ასახვა;
	6. თითოეული გვერდისათვის სერვერის Response Time-ს ასახვა
	7. სასურველია ყველა მომხმარებლის სესიების იდენტიფიცირება და აპლიკაციაში მათი გზის განსაზღვრა და კონვერსიის ჩვენება;
	8. ავტომატურად WEB ალიკაციაში ყველა მომხმარების მარშუტის რუქი ასახვა;
	9. ვებ ტრანზაქციებში სერვერულ ტრანზაქციებთან კორელკაციის შესაძლებლობა იმისათვის რომ მოხდეს დეტალებში ჩაღრმავება Root Cause-ის განსზაღვრისათვის;
	10. სასურველია ბრაუზერში მომხმარებლის სესიებიდან ბიზნეს მაჩვენებლების შეგროვების საშუალება;
70. მობილური აპლიკაციის მონიტორინგის ლიცენზიები უნდა შეიცავდეს 40 000 მომხმარებლისთვის საჭირო ლიცენზიებს. ამასთანავე შესვლის რაოდენობა უნდა იყოს არანაკლებ 5 მილიონისა წელიწადში ან ულიმიტო.
71. მობილური აპლიკაციების მონიტორინგის შემდეგი ფუნქციონალი:
	1. iOS/Android მობილური აპლიკაციების მხარდაჭერა
	2. მობილურ აპლიკაციაში ჩაშენებული მობილური ბრაუზერის მონიტორინგის ფუნქიონალი;
	3. სასურველია იყოს შესაძლებლობა ავტომატურად ასახოს მობილურ აპლიკაციაში ყველა მომხმარებლის მარშუტის რუქა;
	4. მობილური აპლიკაციისა და მომხმარებლის აპლიკაციაში განვლილი მარშუტის (განხორციელებული ქმედებების) და კლიენტის გამოცდილების (ე.წ User Experiance) ურთიერთ ქმედებისა და ურთიერთ დამოკიდებულებების ვიზუალიზაციის საშუალება;
	5. ინფორმაციის ასახვა მობილურ აპლიკაციაში მომხდარი შეცდომების შესახებ, მათ შორის HTTP და ქსელური შეცდომების;
	6. სასურველია ინფორმაციის ასახვა მობილური პლიკაციის სერვერის მხარეს განხორციელებული ე.წ. API Call-ების წარმადობის შესახებ;
	7. აპლიკაციის ხარვეზებისა და გამონაკლისების ავტომატური იდენტიფიცირება;
	8. აპლიკაციის ე.წ. Crash-ების ავტომატიზირებული ანალიზი
	9. გადაწყვეტილება უზრუნველყოფს ე.წ. Crash-ების ანალიტიკას
	10. მობილური და სერვერული ტრანზაქციების კორელაციის საშუალება რათა მოხდეს პრობლემებისა და დაბალი წარმადობის გამომწვევი მიზეზების ძირეული ანალიზი;
	11. მობილური მოწყობილობებისა და სერვერების დიაგნოსტიკის ერთიანი მომხმარებლის ინტერფეისი;
	12. მონაცემების ინტეგრაციის და კორელაციის შესაძლებლობა მონიტორინგის, მობილური მოწყობილობებისა, სერვერული და ინფრასტრუქტურულ ნაწილებს შორის;
	13. გეოგრაფიულ (ელექტორნულ) რუკაზე ე.წ. Respose Time-ის მდგომარეობის ვიზუალიზაციის ფუნქციონალი;
	14. სასურველია მობილურ აპლიკაციაში მომხმარებლის სესიასთან დაკავშირებული ეკრანის ე.წ. ScreenShot-ების შეგროვაბის საშუალება;
	15. მობილურ მოწყობილობაში მომხმარებლის სესიის ანალიტიკის უზრუნველყოფა (მიწყობილობის ტიპის, აპლიკაციის ვერსიის, ოპერაციული სისტემი, მობილური ოპერატორის და სხვა);
72. სასურველია იყოს შესაძლებლობა როგორც ტექნიკური ასევე ბიზნეს მეტრიკების ანალიზის საშუალება (აპლიკაციის სერვერების მოანაცემებიდან) მიმართვის ენის საშუალებით - აგრეგატის გათვლა, დაჯგუფება, ამოკრეფა;
73. მონიტორინგის სისტემას უნდა ქონდეს სტანდარტული მზა დეშბორდები
74. სასურველია ბიზნეს მეტრიკებისათვის საბაზო ქცევის ნორმების აგების საშუალება 1 წლის პერიოდით
75. მონიტორინგის სისტემას უნდა ქონდეს მომხმარებლის დეშბორდის კონფიგურირების შესაძლებლობა, იმისათვის რომ შესაძლებელი იყოს როგორც ტექნიკური ასვე სასურველია ბიზნეს მაჩვენებლების გრაფიკულად გამოსახვა (bar-chart, pie-chart, timeseries, treemap);
76. სასურველია გადაწყვეტილებას უნდა ქონდეს, როგორც მომხმარებლის მონიტორინგის მონაცემების ასევე სერვერების მონიტორინგის მონაცემების ანალიზის საშუალება, მონაცემთა ძირითადი ველის მიხედვით კონვერსიული ღრმულის სახით ვიზუალიზაციის/დეშბორდის აგების
77. სასურველია იყოს შესაძლებლობა, აპლიკაციის შიგნით არსებული ინსტრუმენტების გამოყენებით; უნიკალური იდენტიფიკატორის მეშვეობით ბიზნეს პროცესის ვიზუალიზაციის აგების საშუალება, რომლებიც შეიძლება გადანაწილებული იყოს სხვადასხვა სისტემებს შორის;
78. სასურველია იყოს შესაძლებლობა რომ მონიტორინგის სისტემას უნდა ქონდეს მომხმარებლის მიერ გავლილი ე.წ. მომხმარებლის გზის ყოველი გავლილი ნაბიჯის საშუალო დროის გათვლის შესძებლობა (მაგალითისათვის: Login - პირად კაბინეტში ყოფნა - თანხების გადარიცხვა - ტრანზაქციის დადასტურება)
79. სასურველია მომხმარებლის გზის აგების და ვიზუალიზაციის საშუალება მის ყოველ ნაბიჯზე მომხმარებლის გამოცდილების გათვლის შესაძლებლობით;
80. სასურველია ბრაუზერის სესიის ანალიტიკის უზრუნველყოფა (მოწყობილობის ტიპის, აპლიკაციის ვერსიის, ოპერაციული სისტემი, მობილური ოპერატორის და სხვა);
81. სასურველია დროის რეალურ რეჟიმში ნიბეისმიერი ბიზნეს მეტრიკისათვის (გადარიცხვების რადენობა, დაერთბებეის რაოდენობა, აქტიური სესების რაოდენბობა) ნორმის წესის ავტომატიური აგება არანაკლებ ერთი თვის პერიოდისათვის;
82. სასურველია გამოძახებების (Request) მოქნილი ენის გამოყენებით დეშბორდების ან /და რეპორტებსი შექმნის და აგების საშულება ფილტრაციის რთული კრიტერიუმების გამოყენებით როგორც მომხმარებლის გამოცდილების (User Expiriance) ასევ აპლიკაციის მონიტორინგისათვის.
83. PDF ფორმატში რეგულარული რეპორტების შექმნის შესაძლებლობა მათი დაგზავნის ავტომატიზაციის შესაძლებლობით;

# **ტესტირება / დანერგვა**

გამარჯვებულმა პრედენდენტმა, მასთან საბოლოო ხელშეკრულების გაფორმებამდე, უნდა მოახდინოს შემოთავაზებული გადაწყვეტილებით სატესტო სისტემის გამართვა და შემსყიდველისთვის ტესტირების შედეგების წარდგენა. შედეგების წარსადგენი ვადა არაუმეტეს 3 კვირა.

ტესტირების დროს აუცილებელი წესით უნდა მოხდეს ქვემოთ ჩამოთვლილი სცენარების ტესტირება (ასევე, შემსყიდველი იტოვებს უფლებას დაამატოს სატესტო სცენარები, რაზეც წინასწარ მოხდება პრეტენდენტის ინფორმირება):

* + შესაძლებლობა შექმნას custom(მორგებული) მეტრიკები ოპერაციებისთვის BackEnd-ის მხარეს. (მეტრიკა - ამ შემთხვევაში გარკვეული ოპერაციების წარმადობის შეფასებისათვის გამოყენებული მაჩვენებლები, კრიტერიუმები და მნიშვნელობები)
	+ ტრანზაქციების 100%-ის მონიტორინგი და შეფასება საწყისიდან ბოლო ტრანზაქციამდე აგენტების საშუალებით ლინუქს სერვერზე install/configure/start ლინუქს სერვერზე root/sudo უფლებების გარეშე.
	+ სხვადასხვა მონაცემთა ბაზების (მაგ. SQL, PostgreSQL, Mongo DB, Oracle) დეტალური მონიტორინგი და ინფორმაციის შეგროვება ისეთი პარამეტრებისათვის როგორიცაა sql execution plan, სესიის ინდენთიფიკატორები, ბლოკირებების დეტალიზაცია და სხვა.
	+ სასურველია მობილური აპლიკაციის მომხმარებლის სესიის სქრინშოტები შექმნის საშუალება.
	+ IT და სასურველია Business პროცესის მეტრიკზე დინამიურად აგებული ნორმები(ქცევის ნორმებზეა საუბარი) (დღიური, 7 დღის , თვის, წლის , შერჩეული)
	+ სისტემის ინფრასტრუქტურის, მონაცემთა ბაზის რესურსების მონიტორინგი , მეტრიკების დინამიური ნორმის განსაზღვრით (დღიური, 7 დღის , თვის, წლის , შერჩეული)
	+ შეტყობინებების მექანიზმი ფუნქციონალის ტესტირება (ჩვენება) და სასურველია ბიზნეს მეტრიკების ადაპტირებადი საბაზო ქცევის ნორმიდან გადახრის შემთხვევაში.
	+ რთული ე.წ Health Rul-ების გამოყენება სასურველია ნებისმიერი IT მეტრიკებისთვის, და სასურველია ბიზნეს ლოგიკური ოპერაციების გამოყენებით (or,and,not)
	+ ინციდენტების მონიტორინგისთვის ერთიანი ინტერფეისის გამოყენება (war-room)
	+ შეიქმნას ახალი რელიზების ოპრეტიული კონტროლისთვის რეპორტინგი/დეშბორდი, საკვანძო მაჩვენებლების წარმადობისა და სასურველია ბიზნეს შედეგებისათვის. (ახალ რელიზებში საკვანძო ბიზნეს მეტრიკების გამოსახვა/ვიზუალუზაცია ძველ ვერსიებში საბაზო ქცევის ნორმასთან კორელაციით და შედარებით)
	+ ისტორიული მონაცემების ექსპორტი , წარმადობის დინამიკაზე მონოტორიგნის შესაძლებობით ან ნებისმიერი ბიზნეს მაჩვენებლით - PDF ფორმატში
	+ სასურველია ბიზნეს მაჩვენებლებზე ზეგავლენის მქონე, Customer Expirience-ის მრავლობითი პარამეტრების გათვალისწინებით ნორმიდან გადახრის ე.წ. ჭკვიანი შეტყობინების შექმნის შესაძლებლობა.
	+ სასურველია მობილური აპლიკაციიდან სქრინშოტის აღების საშუალება, მომხმარებლის ქმედების შეფასებისთვის სატესტო და პროდუქტიულ გარემოში
	+ შეიქმნას ძირითად ტრანზაქციების დეშბორდი, რომელიც გვიჩვენებს დინამიური ნორმიდან გადახრას Production მეტრიკებზე, მეილზე მიღების შესაძლებლობით
	+ სასურველია შეგროვდეს Business და IT მეტრიკები აპლიკაციიდან, აპლიკაციის კოდის ცვლილების გარეშე. რის საფუძველზეც ავტომატურად უნდა შეიქმნას ე.წ. დინამიური საბაზო მაჩვენებლები (Dynamic Baseline) (დღიური, 7 დღის , თვის, წლის , ე.წ. custom) თვითოეულ მეტრიკზე.
	+ სასურველია მოთხოვნების შესაქმნელად SQL like ენა, ბიზნეს ინფორმაციის მისაღებად მათი კორელაციით შესაძებლობით ინფრასტუქტურის (მათ შორის ე.წ. custom) პროდუქტიულობის მეტრიკებთან (DataBase, Servers, etc)
	+ სასურველია ბიზნეს მეტრიკების შეგოვება, მაგალითად გახსინლი დეპოზიტების საერთო თანხა/ჯამი, მობილური ტელეფონის ანგარიშის შევსება, აპლიკაციის სერვერების ყველა (100%-ვე) ტრანზაქციების გამოყენებით, მათი დეშბორდზე გამოსახვის შესაძლობით. ასევე მათზე წვდომა API-ის მეშვეობით
	+ სასურველია რეპორტების მომზადება იმ ძირითადი ბიზნეს ტრანზაქციების შესახებ რომლებიც ავლენენ დინამიური ნარმებისა და მეტრიკებიდან გადახრას შემდგომში დაინტერესებული ან და წინასწარ განსაზღვრული პირებისათვის მეილზე PDF ფორმატში გაგზავნის საშულებით
	+ სასურველია ძირითად Sql request-ის პარამეტრეებზე ბიზნეს მეტრიკების შექმნა, და დინამიური BaseLine -ის გამოსახვა (დღიური, 7 დღის , თვის, წლის , შერჩეული) თვითოეულ მეტრიკზე.
	+ სასურველია საიტზე მომხმარებლის „გზის“ -ის ავტომატური შემქნა, თვითოეული ეტაპზე პროდუქტიულობის (სისწრაფე, შესრულება) განსაზღვრით და გასული მომხმარებლების რაოდენობის განსაზღვრით.
	+ სასურველია FrontEnd-დან და BackEnd-დან კონვერსიული ღრმულის ვიზუალური რეპორტი, ნელ ბიზნეს ტრანზაქციებში DrillDown-ის (ღრმად გაანალიზების) შესაძლებლობით, იმისათვის რომ შესაძლებელი იყოის პრუდუქტიულობის დაცემის მიზეზის (root cause) განსაზღვრა.
	+ სასურველია ვიზუალური რეპორტი ბიზნეს პროცესების ნაბიჯებზის/ეტაპების გამოსახულებით, როგორც თითოეული ნაბიჯის გავლის დროის მითითებით, ასვე სრული პროცესის საერთო დროის განსაზღვირს შესაძლებლობით.
	+ სასურველია ბიზნეს პროცესის და ნებისმიერ ეტაპზე მათი საბაზო ქცევის ნორმიდან გადახრის ვიზუალიზაცია

# **საგარანტიო პირობები**

* მონიტორინგის სისტემის მხარდაჭერის ღირებულება უნდა შედიოდეს ლიცენზიების ფასში და მისი მოქმედებდეს ვადა არ უნდა იყოს შესყიდული გადაწყვეტილების ლიცენზიების მოქმედების ვადაზე ნაკლები.
* მხარდაჭერა უნდა ხორციელდებოდეს 24X7 წელიწადში 365 დღის განმავლობაში, საჭიროების შემთხვევაში მწარმოებლის ტექნიკური მხარდაჭერის ინჟინრებთან კომუნიკაციის შესაძლებლობით.
* პროგრამული განახლებები ხელმისაწვდომი უნდა იყოს ლიცენზიის და მხარდაჭერის მოქმედების სრული პერიოდის განმავლობაში.
* ე.წ . Support Case-ბის გახსნა შესაძლებელი უნდა იყოს კლიენტის პორტალის მეშვეობით ან შესაბამის საკონტაქტო რესურსზე წვდომით მაგ, ელექტრონული ფოსტა, ტელეფონი და ა.შ.
* მონიტორინგის სისტემის საპორტის და საპორტის ფარგლებში არსებული SLA-ის შესახებ პრეტენდენტის მიერ მოწოდებული ინფორმაციის გადამოწმება შესაძლებელი უნდა იყოს ვენდორის ოფიციალურ საიტზე და უნდა იყოს ღია და წვდომადი.

# **დამატებითი მოთხოვნები**

* პრეტენდენტმა უნდა წარმოადგინოს კომერციული შემოთავაზება 1 და 3 წლიან პირობებზე ცალცალკე ოფციად.
* დასამონიტორინგებელი Production გარემოს სერვისების რაოდენობის, მათ მიერ გამოყენებული რესურსების და მათში გამოყენებული ტექნოლოგიების დასაზუსტებლად გთხოვთ მოგვმართოთ შემდეგ მისამართზე: g.megrelishvili@vtb.ge არანაკლებ 3 სამუშაო დღისა ტენდერის გამოცხადებიდან.
* პრეტენდეტმა კომპანიამ უნდა წარმოადგინოს წინადადება გადაწყვეტილებაზე რომელიც დაფარავს როგორც Production, ასევე ანალოგურ Pre-Production გარემოს.
Pre-Production გარემოს ლიცენზიები წარმოდგენილი უნდა იქნეს ცალკე ოფციად.
* პრეტენდენტმა უნდა წარმოადგინოს მწარმოებლის ავტორიზაციის ფორმა (MAF). აღნიშნულ ფორმაში მითითებული უნდა იყოს მიმდინარე ტენდერის უნიკალური ნომერი ან/და დასახელება.
* წინადადება უნდა მოიცავდეს 20 მომხმარებლისთვის ვენდორის ონლაინ სასწავლო კურსებთან წვდომას. ონლაინ კურსები ხელმისაწვდომი უნდა იყოს არანაკლებ გადაწყვეტილების ლიცენზიების მოქმედების ვადაზე.
* კომერციული წინადადება დამატებით(ცალკე ოფციად) უნდა შეიცავდეს სპეციალიზირებული ავტორიზებული სწავლების ორგანიზების ღირებულებას კონკრეტული სპეციალიზების(დანერგვა, მომსახურება, მართვა, მონიტორინგის წესების აწყობა და ა.შ..) მიხედვით 5 ადამიანზე.

**შენიშვნა:**

ინფორმაცია ანგარიშსწორების შესახებ:

ინფორმაცია მიწოდების ვადის შესახებ:

პრეტენდენტის ხელმოწერა:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_