

Web-არქიტექტურის მიმოხილვითი ანალიზი

იოსებ ძმანაშვილი, ზურაბ ბოსიკაშვილი

საქართველოს ტექნიკური უნივერსიტეტი

რეზიუმე

ბოლო ათწლეულის მანძილზე, განსხვავებული ტიპის გამოთვლითი პლატფორმებისა და მოწყობილობების სწრაფად მზარდი სპექტრის ფონზე განსაკუთრებული მნიშვნელობა შეიძინა მათი ინტეგრაციისა და თანათავსებადობის უზრუნველყოფის ამოცანებმა. თანამედროვე ინფორმაციული სამყაროს უკიდურესად ჰეტეროგენული ბუნება და კოლოსალური მასშტაბები მკაფიოდ მიანიშნებს მდგრადი, მასშტაბირებადი და ადაპტირებადი ეველუციის თვისებების მქონე, ქსელურ არქიტექტურაზე დაფუძნებული დისტრიბუციული გადაწყვეტილებების შემუშავების მნიშვნელობასა და აუცილებლობას. მიმდინარე სტატიის ფარგლებში წარმოვადგენთ ვებ არქიტექტურის საკვანძო პრინციპების მიმოხილვას, მათი წარმოშობის განმარტებულ ფაქტორებსა და მოდელს, რომელზე დაყრდნობითაც შესაძლებელია არა რომელიმე კონკრეტული პლატფორმისკენ მიმართული, არამედ ზოგადი, ზიარ აღქმაზე დაფუძნებული, შესაბამისი სინტაქსური და სემანტიკური მახასიათებლების მქონე თვითაღწერადი გზავნილების მიმოცვლაზე დაფუძნებული ურთიერთქმედებისა და ინტეგრაციის თანათავსებადი მოდელის შემუშავება.

საკვანძო სიტყვები: ინტერნეტი. ვებ-არქიტექტურა. მოდულები. ინტეგრაცია. კლიენტ-სერვერ არქიტექტურა. ინფორმაციის გადაცემა. რესურსი. რეპრეზენტაცია. ჰიპერტექსტი.

1. შესავალი

ბოლო წლების ტექნოლოგიურმა ცვლილებებმა და პროგრესმა მკვეთრად შეცვალა ინტერნეტის ლანდშაფტი. მობილური პლატფორმებისა და ტექნოლოგიების გაენითარებამ ხელი შეუწყო ახალი თაობის ურთიერთდაკავშირებული მობილური მოწყობილობების, აპლიკაციებისა და მათი მომსახურებისა და ინტეგრაციისთვის განკუთვნილი სერვისების ეკოსისტემების წარმოქმნასა და განვითარებას. ტრადიციული, ვებ ბრაუზერებისთვის განკუთვნილი აპლიკაციების პარალელურად მნიშვნელოვანი როლი შეიძინა მობილური მოწყობილობებისთვის განკუთვნილმა მკვიდრ ტექნოლოგიებზე დაფუძნებულმა სპეციალიზებულმა აპლიკაციებმა, ხოლო მათი ჩართულობისა და ინტეგრაციის უზრუნველყოფის ამოცანამ გამოიწვია სრულიად ახალი თაობის, ტრადიციული RPC[1,2] სტილის ვებ სერვისებისგან განსხვავებული ვებ არქიტექტურასა და პროტოკოლებზე დაფუძნებული მიდგომების განვითარება. მიდგომის პოპულარობა ძირითად განპირობებულია ვებ პროტოკოლების, სპეციფიკაციების, მონაცემთა სერიალიზაციის ფორმატების, ინფრასტრუქტურის ფართო ხელმისაწვდომობითა, პლატფორმებს შორის არსებული კონსისტენტური მხარდაჭერით და შედეგის მიღწევის შედარებითი სიმარტივით.

2. კვლევის საგანი

მიუხედავად ტექნოლოგიური პროცესებით გამოწვეული ტენდენციებისა, ახალი თაობის სერვისების რეალიზაციის გავრცელებული მიდგომები არასაკმარის თანხვედრაში იმყოფება ვებ არქიტექტურის ძირითად პრინციპებთან, და ხშირ შემთხვევაში შედეგი მიიღწევა ვების ფუნდამენტური - ჰიპერტექსტის მიმოცვლის პროტოკოლის საკვანძო სემანტიკური და არქიტექტურული პრინციპების უგულვებლყოფით, რაც დიდწილად გამოიხატება პროტოკოლის როგორც მონაცემების უბრალო ტრანსპორტირების მექანიზმად გამოყენებაში.

თუმცა, გამოდინარე იქედან, რომ მხოლოდ მონაცემების არსებობა და მათი ტრანსპორტირების შესაძლებლობა ავტომატურად არ გულისხმობს მონაცემებში ინკაფსულირებული ინფორმაციის დანიშნულებისა და ამ ინფორმაციით მანიპულირების შესახებ

ზიარ ცოდნას ურთიერთქმედების მონაწილე მხარეთა შორის, როგორც წესი შედეგად ვლდებულობთ ევოლუციის შესაძლებლობებს მოკლებულ[3], მჭიდროდ ურთიერთდაკავშირებული კომპონენტებისგან შედგენილ მონოლითურ სისტემებს, რაც ბუნებრივ წინააღმდეგობაში მოდის ვების - როგორც პლატფორმის დანიშნულებასთან.

3. ვებ არქიტექტურის ძირითადი პრინციპები

ვებ არქიტექტურის პრინციპების ჩამოყალიბებაზე მნიშვნელოვანი ზეგავლენა იქონია სხვა ისეთმა დისციპლინებმა როგორებიცაა: სამშენებლო არქიტექტურა, ლიტერატურა, ვიზუალური პერცეფცია, ინდუსტრიული დიზაინი და ა.შ. აღნიშნულ დისციპლინებში აღმოცენებულმა პრინციპებმა მატერიალიზაცია კპოვა ვების ფორმირებაში და მნიშვნელოვანწილად განსაზღვრა მისი როგორც პლანეტარული მასშტაბისა და მნიშვნელობის მქონე ფენომენის მომავალი.

ვებ არქიტექტურა ეფუძნება ფუნდამენტურ პრინციპს სადაც, აგენტებს შორის ურთიერთქმედება იმართება წინასწარ განსაზღვრული სინტაქსისა და სემანტიკის მქონე გზავნილების მიმოცვლის მეშვეობით, და თავისთავად გულისხმობს ზიარ მოლოდინს თითოეულ გზავნილში განთავსებული სამართავი მონაცემებისა და შიგთავსის(რეპრეზენტაციის მონაცემები და მეტა მონაცემები) ინტერპრეტაციის შესახებ. კომუნიკაციის პროტოკოლის მიერ მხარდაჭერის შემთხვევაში, გამოყენებული რეპრეზენტაციული მეტა მონაცემების ერთობლიობა განსაზღვრავს გამგზავნის მიზანს და მიუთითებს მიმღებს თუ როგორ უნდა განხორციელდეს გზავნილის ინტერპრეტირება[4]. ნახსენები პრინციპები რეალიზებულია ჰიპერტექსტის მიმოცვლის პროტოკოლში(HTTP), რომელიც განსაზღვრავს რესურსთან ურთიერთქმედების ერთგვაროვან ინტერფეისს რესურსის რეპრეზენტაციის მიმოცვლისა და მანიპულირების გზით[5,6].

3.1.რესურსი

ნებისმიერი ინფორმაცია, რომლის იდენტიფიცირებაც შესაძლებელია რესურსის იდენტიფიკატორის(URI[10]) მეშვეობით, შესაძლებელია განვიხილოთ როგორც რესურსი: ნებისმიერი ფორმატის დოკუმენტი, მოთხოვნის პასუხად დაბრუნებული დინამიური რეზულტატი, სხვა რესურსების კოლექცია და ა.შ[7]. ვებში ნებისმიერი მოთხოვნის საბოლოო ადრესატს წარმოადგენს რესურსი და მასთან ურთიერთქმედების ინტერფეისი განსაზღვრულია HTTP პროტოკოლის მიერ[5].

რესურსი არის კონცეპტუალური შესაბამისობა ეკვივალენტურ ინფორმაციულ ერთეულთა სიმრავლესთან და არა კონკრეტული ინფორმაციული ერთეული, რომელიც შეესაბამება რესურსს დროის რომელიმე მომენტში[7]. რესურსი შესაძლოა იყოს სტატიკური იმ თვალსაზრისით, რომ დროდადრო ინსპექტირებისას იგი შეესაბამებოდეს ერთსა და იმავე მდგომარეობას(ან მნიშვნელობას). ზოგიერთი რესურსი შესაძლოა გამოირჩეოდეს შესაბამისი მდგომარეობის მაღალი ცვალებადობით დროის ნებისმიერ მომენტში. თუმცა, მნიშვნელოვანია, რომ რესურსის შესაბამისობის სემანტიკა იყოს სტატიკური და დარჩეს უცვლელი დროთა განმავლობაში, რადგან სწორედ სემანტიკაა რესურსებს შორის მთავარი განმასხვავებელი ფაქტორი.

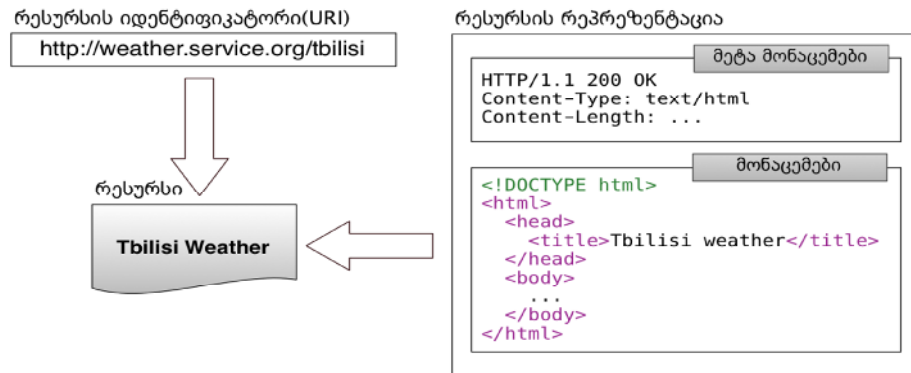
მაგალითად, „წერტილი რუკაზე“ შესაძლებელია განვიხილოთ როგორც კონკრეტული სემანტიკის მქონე რესურსი, რომელსაც გააჩნია მახასიათებლები გრძედისა და განედის სახით და აღწერს არა რომელიმე კონკრეტულ, არამედ ნებისმიერ წერტილს რუკაზე. შესაბამისად, რესურსთან ურთიერთქმედებისას საკმარისი იქნება მხოლოდ იმის ცოდნა, თუ რა ტიპის რესურსთან გვაქვს საქმე და არა ის თუ რომელი კონკრეტული წერტილის რეპრეზენტაციას მივიღებთ რესურსთან ასოცირებული იდენტიფიკატორის განმისამართების შედეგად.

დამატებით ანალოგიად შესაძლოა გამოვიყენოთ ნებისმიერ სტატიკურად ტიპიზებულ დაპროგრამების ენაში არსებული კონკრეტული ტიპის ცვლადი. მაგალითად, მთელი ტიპის ცვლადი განსაზღვრავს შესაბამისობას არა რომელიმე კონკრეტულ, არამედ ნებისმიერ მთელი ტიპის რიცხვთან. ცვლადის არსებობის მანძილზე მისი მნიშვნელობა შესაძლებელია იცვლებოდეს ან დარჩეს უცვლელი, თუმცა კოდის ნებისმიერ ფრაგმენტისთვის, რომელიც მანიპულირებს აღნიშნული ცვლადით, დიდწილად მნიშვნელოვანია მხოლოდ მისი სახელის (იდენტიფიკატორი) და ტიპის (სემანტიკური შესაბამისობა) ცოდნა და არა მისი კონკრეტული მნიშვნელობა.

3.2. რესურსის რეპრეზენტაცია

რესურსის რეპრეზენტაცია ეწოდება ინფორმაციას, რომელიც ასახავს რესურსის წარსულ, მიმდინარე ან მოთხოვნილ მდგომარეობას, ფორმატში, რომლის მიმოცვლაც შესაძლებელია პროტოკოლის მიერ და შედგება რეპრეზენტაციის მონაცემებისა და რეპრეზენტაციის მეტა მონაცემების ერთობლიობისგან [8]. რეპრეზენტაციის ფორმატი ცნობილია როგორც მედია ტიპი. გზავნილში განთავსებული რეპრეზენტაცია შესაძლებელია დამუშავდეს მიმღების მიერ, გზავნილში განთავსებული სამართავი მონაცემებისა და მედია ტიპის ბუნებიდან გამომდინარე. მედია ტიპი შესაძლოა განკუთვნილი იყოს ავტომატიზებული დამუშავებისა ან ვიზუალური წარმოდგენისთვის ან თავის თავში აერთიანებდეს ორივე შესაძლებლობას [4,7].

1-ელ ნახაზზე სქემატურად არის წარმოდგენილი რესურსის იდენტიფიკატორის, რესურსისა და რესურსის რეპრეზენტაციას შორის არსებული დამოკიდებულება.



ნახ.1. რესურსი, რესურსის იდენტიფიკატორი და რეპრეზენტაცია

რესურსი არის იდენტიფიკატორის მქონე კონცეპტუალური შესაბამისობა, ხოლო რეპრეზენტაცია რესურსის განმათავსებელი სერვერის მიერ მისადაგებული მდგომარეობა რესურსის მიმართ გაგზავნილი მოთხოვნის მიღების შემდეგ. რეპრეზენტაცია განკუთვნილია მოთხოვნის პასუხის მიმღებისთვის და შეიცავს მისი დამუშავებისთვის აუცილებელ ყველა საჭირო ინფორმაციას. რეპრეზენტაციის ფორმატი არ არის შეზღუდული და გამომდინარე კლიენტის მოთხოვნიდან, სერვერმა შესაძლოა გამოიყენოს განსხვავებული ფორმატი. გამომდინარე იქედან, რომ რესურსის სემანტიკური შესაბამისობა და მასთან ურთიერთქმედების ინტერფეისი რჩება უცვლელი, ხოლო მედია ტიპი დამოუკიდებლად განსაზღვრავს მისი დამუშავების მოდელს, აუმჯობესებს განსხვავებული ტიპის კლიენტების ეფექტურად მომსახურების შესაძლებლობებს.

3.3. ჰიპერტექსტი

ჰიპერტექსტის მიმოცვლის პროტოკოლის მიერ განსაზღვრული რესურსებთან ურთიერთქმედების ერთგვაროვანი ინტერფეისის და სხვა სემანტიკური დეტალების გარდა,

მნიშვნელოვანია ასევე ჰიპერტექსტი, რომელსაც საკვანძო როლი აკისრია პროტოკოლის ფარგლებში და ერთერთი განმარტებით იგი არის, სხვადასხვა მონაცემებში გაბნეული ასოცირებულ კავშირების ერთობლიობა, რომელიც განსაზღვრავს ამ მონაცემებში ჩაქსოვილ ინფორმაციაზე წვდომისა და მანიპულირების შესაძლებლობებს[19].

ზემოთ მოცემული ჰიპერტექსტის განმარტება არასაკმარისად ცხადია ვებ არქიტექტურის კონტექსტში. გამომდინარე იქედან, რომ ჰიპერტექსტი იყო აქტიური განხილვის საგანი მრავალი ათწლეულის მანძილზე, მის ფორმირებაზე ზეგავლენა იქონია არაერთმა დისციპლინამ და დროსთან ერთად მან შეიძინა არაერთი დამატებითი დატვირთვა, რამაც საბოლოო ასახვა ჰპოვა ვებში.

მაგალითად, ფილდინგი ჰიპერტექსტს განმარტავს როგორც, ნიშნთვისებათა ერთობლიობად გარდაქმნილი ინფორმაციისა და სამართავი ელემენტების ერთდროული ასახვა, რომელთაგანაც მომხმარებელს გააჩნია ქმედების შერჩევისა და განხორციელების შესაძლებლობა“ [10].

შესაბამისად ჰიპერტექსტში, ასოცირებული კავშირის გარდა ჩნდება ისეთი მნიშვნელოვანი ცნებები, როგორებიცაა ქმედება და ნიშნთვისება, რომლებიც უკავშირდება კოგნიტიურ ფსიქოლოგიის დარგს და მნიშვნელოვანწილად განსაზღვრავს ვების, რესურსებთან ურთიერთქმედების პროცესის ირგვლივ აგებულ მოდელს. ინფორმაციის სისრულისთვის, მომდევნო ქვეთავებში განვიხილავთ ასოცირებული კავშირის, ქმედებისა და ნიშნთვისების ცნებებსა და იმ ფაქტორებს, რომლებმაც ზეგავლენა იქონია მათ ფორმირებაზე.

3.3.1. ასოცირებული კავშირი

ასოცირებული კავშირი ვებში ცნობილია როგორც ჰიპერმიმითებელი(ან ბმული)[11,12] და მისი სტრუქტურა განსაზღვრულია რესურსის იდენტიფიკატორის სპეციფიკაციის მიხედვით[13].

სტრუქტურული მახასიათებლების გარდა მნიშვნელოვანია უშუალოდ კავშირის სემანტიკა რამდენადაც სწორედ კავშირის სემანტიკა განსაზღვრავს როგორც ცალკეული რესურსის, ასევე რესურსებს შორის ურთიერთდამოკიდებულების ხასიათს. ეს კავშირები შესაძლოა იყოს სხვადასხვა ტიპის - შინაარსობრივი ან კონტექსტუალური, სტრუქტურული, შემცველობითი და ა.შ.

3.3.2. ქმედება

ჰიპერტექსტის ფარგლებში ქმედების ხასიათს განსაზღვრავს კონკრეტული ტიპიზებული ბმულისა ან ბმულის აღწერისთვის გამოყენებული სტრუქტურის სემანტიკა. სემანტიკიდან გამომდინარე ქმედების მოსალოდნელი შედეგი შესაძლოა დავახასიათოთ როგორც წაკითხვა, ტრანსკლუზია[9], მონაცემის გადაცემა, შექმნა, წაშლა, რედაქტირება და ა.შ. თუმცა, როგორც სხვა ნებისმიერ ობიექტთან, ასევე რესურსთან ურთიერთქმედებისას განხორციელებული კონკრეტული ქმედება არ არის იზოლირებული აქტი და შესაძლოა მოიცავდეს რამდენიმე ეტაპს[14]. ეს ეტაპებია: 1. მიზნის დასახვა; 2. განზრახვის ფორმირება მიზნის მისაღწევად; 3. ქმედების შერჩევა; 4. ქმედების განხორციელება; 5. ქმედების შედეგის შეცნობა; 6. ქმედების შედეგის ინტერპრეტირება; 7. შედეგის შეფასება მიზნის მიმართებაში.

არაა აუცილებელი, რომ ქმედება შედეგობდეს ყველა ზემოთ ჩამოთვლილ ეტაპისგან, მაგრამ, ხშირ შემთხვევაში საბოლოო მიზნის მიღწევას შესაძლოა დაჭირდეს ქმედებათა მიმდევრობის შესრულება, სადაც ყოველი მომდევნო ქმედებას განსაზღვრავს წინა ქმედების შესრულების შედეგი. ობიექტთან ურთიერთქმედების ასეთ ფორმას მიყვებით გამოკვლევადი სისტემის იდეამდე, რომელშიც ყოველი წარმატებული ქმედება შესაძლებელია დავახასიათოთ როგორც სწორი მიმართულებით გადადგომული ნაბიჯი, შეცდომა კი უბრალოდ არასწორად შერჩეული ან არასრულფასოვნად განხორციელებული ქმედება [14].

3.3.3. ნიშანთვისება

ნიშანთვისების ცნება ფართოდ არის გავრცელებული ადამიანურ-კომპიუტერული ურთიერთქმედების დისციპლინაში [15] და განიმარტება როგორც, ობიექტის ან გარემოს მახასიათებელი, რომელიც მომხმარებელს უქმნის გარკვეულ ქმედებათა არჩევასა და განხორციელების შესაძლებლობას [14,16]. თავისთავად ნიშანთვისება განეკუთვნება ობიექტის ღიზანის ასპექტს, რომელიც მიგვანიშნებს თუ როგორ უნდა იქნას გამოყენებული ობიექტი და მიუთითებს მის იმ უშუალო და აღქმად თვისებებს, რომლებიც განსაზღვრავს ობიექტთან ურთიერთქმედების შესაძლებლობებს. ნიშანთვისება თავის თავში აერთიანებს არა მხოლოდ ობიექტის მიმართებაში შესაძლო განხორციელებადი ქმედების, არამედ ამ ქმედების არსებობისა და მისი დანიშნულების წარმოჩენისა და მომხმარებელამდე მიტანის შესაძლებლობასაც [17].

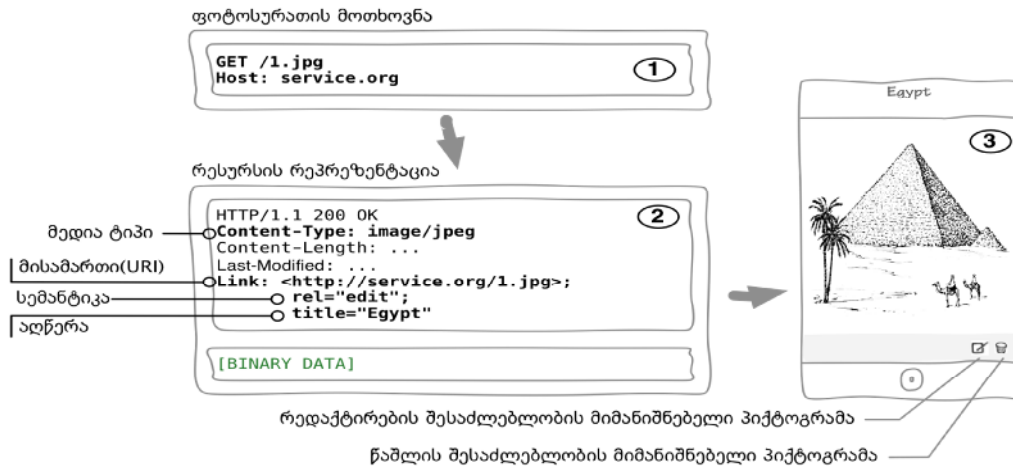
გამომდინარე ზემოთქმულიდან, რესურსის რეპრეზენტაცია შესაძლებელია განვიხილოთ როგორც სასარგებლო ერთეული, რომელიც წარმოადგენს სასარგებლოობის წარმოჩენისა და ამ ინფორმაციის გამოყენებლამდე მიტანის შესაძლებლობათა შერწყმას. ხოლო ნიშანთვისება მახასიათებელი, რომელიც მიანიშნებს მომხმარებელს ამა თუ იმ ქმედების განხორციელების შესაძლებლობის არსებობას მოცემული რესურსის მიმართებაში.

4. ურთიერთქმედებათა კოორდინაცია

ნებისმიერი სერვისის ემსახურება გარკვეული მიზნის მიღწევის შესაძლებლობის უზრუნველყოფას წინასწარ განსაზღვრული და შეთანხმებული ურთიერთქმედებების მიმდევრობის შესრულების მეშვეობით. ტრადიციული ვებ სერვისების შემთხვევაში, აღნიშნული ქმედებები წინასწარ არის განსაზღვრული სტატიკური კონტრაქტის მეშვეობით, ხოლო ქმედებათა სემანტიკა და მიზნის მისაღწევად საჭირო განსახორციელებელ ქმედებათა მიმდევრობა აღწერილი შესაბამის დოკუმენტაციაში. მიდგომა განსაზღვრავს გარკვეულ წინასწარ მოლოდინს სერვისის კლიენტისა და მისი შესაძლებლობების შესახებ და მოითხოვს კლიენტის დაპროგრამებას უშუალოდ ამ სერვისის ინტერფეისის მიმართებაში. იგი არ ეყრდნობა რაიმე ტიპის ზიარ ცოდნას მიმოცვლილ მონაცემებში ინკასფულირებული ინფორმაციის და ქმედებათა სემანტიკის შესახებ. აღწერილი მიდგომის საპირისპიროა რეპრეზენტაციული მდგომარეობის მიმოცვლის არქიტექტურულ სტილზე დაფუძნებული სერვისები, რომლებიც სრულად არის მართული ზიარ ცოდნაზე დაფუძნებული თვითაღწერადი, ნიშანთვისებებით გაჯერებული გზავნილების მიმოცვლის მეშვეობით, სადაც ყოველი შესაძლო მომდევნო ქმედება ამოცნობადია მიმოცვლილი გზავნილებიდან [7].

უფრო მეტი თვალსაჩინოებისთვის განვიხილოთ ჰიპოტეტიური სერვისი, რომელიც მომხმარებლებს სთავაზობს ფოტოსურათების განთავსების, დათვალიერების და ამ ფოტოსურათებით მანიპულირების შესაძლებლობას (მაგალითად, რედაქტირება ან წაშლა და ა.შ.). სადემონსტრაციოდ საკმარისი იქნება თუ შევჩერდებით ერთ კონკრეტულ ფოტოსურათთან მუშაობისთვის აუცილებელ ფუნქციონალზე: ა) ფოტოსურათის წაკითხვის და ჩვენების; ბ) მისი გადამუშავების (ხელახლა ატვირთვა, განსხვავებული ფილტრებით იერსახის შეცვლა, ზომის შეცვლა და ა.შ.); და/ან გ) წაშლა.

მე-2 ნახაზზე წარმოდგენილია ერთ-ერთი ფოტოსურათის წაკითხვის ნიმუში, რომელიც შეიცავს ურთიერთქმედებისას მიმოცვლილი მოთხოვნა/პასუხის გზავნილებსა და პასუხად მიღებული რესურსის რეპრეზენტაციის დამუშავების შედეგად ეკრანზე გამოტანილ ფოტოსურათს. გარდა ფოტოსურათის ვიზუალური წარმოდგენისა ეკრანზე ასევე გამოტანილია სურათის მიმართებაში დამატებითი ოპერაციების განხორციელების შესაძლებლობის მიმანიშნებელი პიქტოგრამები - ამ კონკრეტულ შემთხვევაში რედაქტირება და წაშლა.

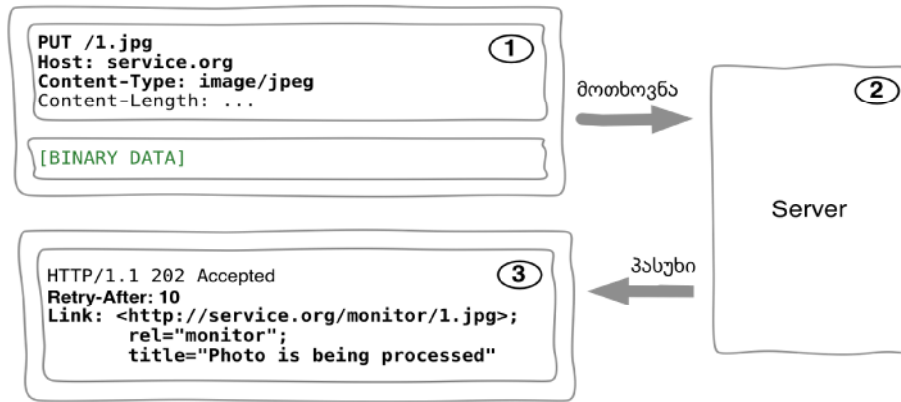


ნახ.2. აგენტის მიერ ინტერპრეტირებული რესურსის რეპრეზენტაცია

ზემოთ მოცემული ერთი შეხედვით ბანალურ ნიმუშზე დაკვირვებით შესაძლებელია უამრავი მნიშვნელოვანი დასკვნის გაკეთება. მაგალითად, რეპრეზენტაციული მეტა მონაცემებიდან გამომდინარე, დამუშავებელ აგენტს(ამ შემთხვევაში მობილურ მოწყობილებაში განთავსებული აპლიკაცია) შეუძლია არა მხოლოდ ვიზუალურად წარმოაჩინოს ინფორმაცია მითითებული ფორმატის სპეციფიკაციის მიხედვით(მედია ტიპი), არამედ მისცეს მომხმარებელს ფოტოსურათის მიმართებაში დამატებითი ოპერაციების განხორციელების შესაძლებლობა. აღნიშნული ინფორმაცია მითითებულია გზავნილში „Link” თავსართისა და მისი „rel” ატრიბუტის მნიშვნელობით[11]. მაგალითში ნაჩვენებია „edit” ანოტაციას გააჩნია წინასწარ განსაზღვრული მკაფიო და ცალსახა სემანტიკა[18] და თავის თავში აერთიანებს რესურსის განმისამართების, რედაქტირებისა და/ან წაშლის შესაძლებლობებს გამოყენებული კომუნიკაციის პროტოკოლის მიერ განსაზღვრული მეთოდების მეშვეობით.

ნაჩვენები ნიმუშიდან მკაფიოდ ჩანს, რომ გზავნილში განთავსებული ინფორმაციის სრულყოფილად ექსპლუატირების შემთხვევაში შესაძლოა მითითებული რესურსის მიმართებაში ქმედებათა მთელი კასკადის აგება, რასაც კვლავ მივეყვართ წინამორბედ თავებში განხილული გამოკვლევადი სისტემების იდეამდე[14], რაც თავის მხრივ შესაძლებელია თვითაღწერადი, ნიშანთვისებებით გაჯერებულ გზავნილებზე დაფუძნებული ურთიერთქმედების მეშვეობით.

აპლიკაციის მიერ შემოთავაზებული რედაქტირების ფუნქციის შესაძლებლობების გამოყენების შემთხვევაში, მომხმარებელს შეუძლია განხორციელოს დამატებითი მანიპულაციები ეკრანზე წარმოდგენილი ფოტოსურათის მიმართებაში. მაგალითად, თუ განვიხილავთ ფოტოსურათის რედაქტირების ოპერაციას, იგი შესაძლებელია დავეყთ შემდეგ ქმედებათა მიმდევრობის ერთობლიობად: 1) ლოკალურად მოდიფიცირება; 2) განმთავსებელი სერვერისთვის გადაგზავნა პერმანენტულად შესანახად. აღნიშნულ ოპერაციათა განხორციელების შემდეგ კლიენტი აპლიკაცია ვალდებულია მომხმარებელს შეატყობინოს ოპერაციის წარმატებით დასრულებისა ან წარუმატებლობის შესახებ. მაგრამ, ზოგიერთ შემთხვევაში ასევე დასაშვებია, რომ მოდიფიცირებული სურათის გადაგზავნის შემდეგ სერვერმა მიიღოს იგი წარმატებით, თუმცა მის დამუშავებას მოანდომოს განუსაზღვრელი რაოდენობის დრო. ასეთ შემთხვევაში მიზანშეწონილია, რომ სერვერმა კლიენტს დაუდასტუროს მოთხოვნის წარმატებით მიღების შესახებ. სცენარის ამ მიმართულებით განვითარების შემთხვევაში მნიშვნელოვანია კლიენტი აპლიკაციის შესაბამისად ინფორმირება სერვერის მხრიდან. მომდევნო ნახაზში მოცემულია ურთიერთქმედების აღნიშნული სცენარის კოორდინაციის ნიმუში:



ნახ.3. სერვერის მიერ მოთხოვნის დასამუშავებლად მიღების დადასტურება

მომხმარებლის მიერ ლოკალური ცვლილებების განხორციელების შემდეგ, კლიენტი აპლიკაცია აგზავნის მოთხოვნას, რომლის სტრუქტურაც წინასწარ არის განსაზღვრული „edit“ ანოტაციის სპეციფიკაციით(1). სერვერი ღებულობს მოთხოვნას(2) და უდასტურებს კლიენტს, რომ მან წარმატებით მიიღო მოთხოვნა, თუმცა მისი დამუშავება ჯერ არ განხორციელებია(3).

აღნიშნულ ინფორმაციას შეიცავს პასუხის სტატუსის კოდი - 202. გარდა ამ ინფორმაციისა გზავნილი შეიცავს „Link“ თავსართს ერთი ბმულით რომლის „rel“ ატრიბუტის მნიშვნელობა „monitor“ მიანიშნებს კლიენტს, რომ მოთხოვნის დამუშავების მონიტორინგის განსახორციელებლად მას შეუძლია ბმულში მითითებული მისამართის გამოყენება და ასევე „Retry-After“ თავსართის მნიშვნელობის მეშვეობით განსაზღვრავს თუ რა სიხშირით უნდა განხორციელდეს მონიტორინგის მისამართზე მოთხოვნის გაგზავნა(ამ კონკრეტულ შემთხვევაში მითითებულია 10 წამი). ამ ინფორმაციაზე დაყრდნობით კლიენტ აპლიკაციას უჩნდება უამრავი ახალი შესაძლებლობა მომხმარებელთან ურთიერთქმედების წარსამართად, თუმცა მნიშვნელოვანია აღვნიშნოთ, რომ რეალიზაციის არჩევანი სრულიად დამოუკიდებელია სერვისის ინტერფეისისგან.

5. დასკვნა

როგორც ვხედავთ, ნიშანთვისებებით გაჯერებული გზავნილების მიმოცვლის მეშვეობით უმჯობესდება ზიარი აღქმა გადაცემული მონაცემების შესახებ, რამდენადაც ასეთი ტიპის გზავნილები შეიცავს ინფორმაციას როგორც მონაცემებთან ასოცირებული სემანტიკის, ასევე ინფორმაციას შესაძლო განხორციელებად ქმედებათა შესახებ მოცემული რესურსის მიმართებაში. შესაბამისად, ურთიერთქმედება, რომელიც არ ეყრდნობა კომუნიკაციის დიაპაზონს მიღმა არსებულ ინფორმაციას, საგრძნობლად ზრდის ურთიერთქმედების ხარისხს მონაწილე მხარეთა შორის, რადგან კომუნიკაციის სტრუქტურის შიგნით, სინტაქსური და სემანტიკური შემადგენლობის მაღალი გარჩევადობა უზრუნველყოფს არა მხოლოდ კლიენტისა და სერვერის ურთიერთქმედებათა კოორდინაციის მაღალ ხარისხს, არამედ ამ ურთიერთქმედებაში ჩართულ სხვა შუალედურ მონაწილეთა სრულყოფილ კოორდინაციას მოთხოვნა პასუხის სრული ციკლის გასწვრივ.

ლიტერატურა:

1. A High-Level Framework for Network-Based Resource Sharing. <http://tools.ietf.org/html/rfc707>
2. Remote procedure call(RPC) http://en.wikipedia.org/wiki/Remote_procedure_call
3. Interoperability. <http://en.wikipedia.org/wiki/Interoperability>>
4. Authoritative Metadata <http://www.w3.org/2001/tag/doc/mime-respect.html>

5. Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content <http://tools.ietf.org/html/draft-ietf-httpbis-p2-semantic-22>
6. Architecture of the World Wide Web, Volume One. <http://www.w3.org/TR/webarch>
7. Fielding, Roy Thomas. Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, University of California, Irvine, 2000.
8. Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies <<http://tools.ietf.org/html/rfc2045>>
9. Transclusion <<http://en.wikipedia.org/wiki/Transclusion>>
10. REST APIs must be hypertext-driven <<http://roy.gbiv.com/untangled/2008/rest-apis-must-be-hypertext-driven#comment-718>>
11. Web Linking <<http://tools.ietf.org/html/rfc5988>>
12. Hyperlink. <<https://en.wikipedia.org/wiki/Hyperlink>>
13. Uniform Resource Identifier (URI): Generic Syntax <<http://tools.ietf.org/html/rfc3986>>
14. The Psychology Of Everyday Things. Donald A. Norman.
15. Human-computer interaction <http://en.wikipedia.org/wiki/Human-computer_interaction>
16. The Ecological Approach To Visual Perception by James J. Gibson.
17. McGrenere, Joanna and Ho, Wayne (2000): Affordances: Clarifying and Evolving a Concept. In: Proceedings of Graphics Interface 2000 May 15-17, 2000, Montreal, Quebec, Canada.
18. Hypermedia and the Web: An Engineering Approach. David Lowe, Wendy Hall. Wiley; 1 edition (January 22, 1999).
19. The Atom Publishing Protocol <http://tools.ietf.org/html/rfc5023>

ANALYSIS OF THE WEB ARCHITECTURE

Dzmanashvili Ioseb, Bosikashvili Zurab
Georgian Technical University

Summary

Increased number of new computational platforms and devices over the past decade changed the landscape of the internet and proved the need in robust, interoperable and high scalable network-based distributed services. Heterogenous nature of the modern information world brought new challenges to existing services ecosystem from both - provider and consumer perspectives while forcing development of whole new generation of services, service infrastructures and APIs. This paper focuses on analysis of design principles behind the Web Architecture and benefits and advantages when applied to services.

АНАЛИЗ WEB АРХИТЕКТУРЫ

Дзманашвили И., Босикашвили З.
Грузинский Технический Университет

Резюме

Рост числа новых вычислительных платформ и устройств за последнее десятилетие изменил ландшафт интернета что доказывает нужду разработки более стабильных, интероперабельных и масштабируемых сетевых систем. Гетерогенная природа современного информационного мира создала новые задачи для существующей сервисной экосистемы как с точки зрения поставщиков так и пользователей в тоже время ускоряя разработку сервисов и сервисных инфраструктур нового поколения. В статье рассмотрены принципы Веб архитектуры и пользы при применении в разработке сервисов.