ივანე ჯავახიშვილის სახელობის თბილისის სახელმწიფო უნივერსიტეტი ზუსტ და საბუნებისმეტყველო მეცნიერებათა ფაკულტეტი

**ელექტრონული სწავლების პორტალი (EL Study)**

**Education Services Electronic Portal (El Study)**

**პროექტის შემსრულებელი სტუდენტები და მათ მიერ შესრულებული სამუშაო:**

* ბექაური ნათია -  წამყვანი პროგრამისტი (ASP.MVC, C#), პროგრამული უზრუნველყოფის არქიტექტურა, სერვისები, მონაცემთა ბაზის არქიტექტურა და რეალიზება, კიბერ-უსაფრთხოების ელემენტები უზრუნველყოფის ფარგლებში.
* კობაიძე გიორგი - პროგრამისტი, ბაზებთან ურთიერთობა.
* გორგოძე გოგა - პროგრამისტი, სერვის პროგრამირება.
* ასათიანი მარიამ -პროექტ მენეჯერი, დავალებების განაწილება, კონტროლი
* კაჭიური თორნიკე - დიზაინერი, დიზაინის შექმნა. ლოგო, front-end დეველოპმენტი
* ბიჭოშვილი თამარი  - ტესტერი, პროგრამული უზრუნველყოფის ტესტირება, შეცდომების გასწორება.

**ხელმძღვანელი**: მანანა ხაჩიძე- ტექნიკის მეცნიერებათა კანდიდატი

თბილისი 2017

**სარჩევი**

[**ანოტაცია** 3](#_Toc486625195)

[**შესავალი** 5](#_Toc486625196)

[**საერთო გამოყენების მოდული** 6](#_Toc486625197)

[**მონაცემთა დამუშავების მოდული** 6](#_Toc486625198)

[**სერვისების- ბიზნეს ლოგიკის მოდული** 7](#_Toc486625199)

[**კლიენტის მხარის მოდული** 8](#_Toc486625200)

[**დასკვნა** 9](#_Toc486625201)

[**გამოყენებული მასალა** 10](#_Toc486625202)

[დანართები 11](#_Toc486625203)

ანოტაცია

პორტალის მომხმარებლები არიან ადმინისტრატორი, პედაგოგები და სტუდენტები,

**პედაგოგი**  დროითი ბადით (დღეების, საათების, აქტივობის ტიპისა და სასურველი აუდიტორიების მიხედვით) ადგენს სასწავლო კურსის განრიგს თავის მიერ არჩეულ საგანში. საგნის ცხრილის  შედგენა საბოლოოდ ხორციელდება ადმინისტრატორის დასტურის საფუძველზე. პედაგოგი ადგენს სემესტრულ დროით განრიგს და ეტაპობრივად ამდიდრებს მას  სასწავლო მასალით -თემის შესაბამისი დოკუმენტებით, სავარჯიშოებით, საკონტროლო კითხვა/დავალებებით. ვებ-გვერდზე არსებობს რეიტინგების სია, რომელშიც ხვდებიან მხოლოდ ფრიადოსანი სტუდენტები, რაც უფრო დიდ სტიმულს აძლევს ახალგაზრდებს.   საკუთარ საგანში პედაგოგს შეუძლია ასევე ატვირთოს ღია და მხოლოდ საგანზე ჩაწერილი სტუდენტებისთვის განკუთვნილი სიახლეები, უპასუხოს საგნის პროფილზე დასმულ კითხვებს (კითხვის დასმა მხოლოდ საგანზე ჩაწერილ სტუდენტებს შეუძლიათ).

**სტუდენტს** შეუძლია ჩაეწეროს ნებისმიერ საგანზე და Follow -ის შედეგად მიიღოს წვდომა საგნის დროით ბადეზე, სიახლეებსა და კითხვებზე, შეუძლია დასვას კითხვა კონკრეტულ საგანთან დაკავშირებით და ასვე უპასუხოს სხვის მიერ დასმულ კითხვას.        კურსები, რომლებსაც გაივლის სტუდენტი, იწერება მის ანგარიშზე. ვებ გვერდზე შესაძლებელი იქნება მომხმარებლის უნარებისა და გავლილი კურსების შეფასების  დაათვალიერება. ამ გზით დამსაქმებელს გაუადვილდება პოტენციური დასაქმებულის შერჩევა. შესაბამისად, უმჯობესი იქნება, სტუდენტმა ასევე თავის ანგარიშში მიუთითოს უნარჩვევები და არსებული გამოცდილება.

პორტალი არის მარტივად კონფიგურებადი ნებისმიერი ცოდნისა და გამოცდილების მომხარებლისათვის. ასევე, მკაცრად არის განსაზღვრული თითოეული მოხმარებლის უფლებები, რომლებიც, თავის მხვრივ, მართვადია ვებ-გვერდის ადმინისტრატორის მიერ. სისტემა შექმნილია Microsoft-ის თანამედროვე არქტექტურული პატერნის - ASP.NET MVC - გამოყენებით .NET FrameWork პლათმორმაზე. ესაა პროგრამული უზრუნველყოფა, რომელიც შეიცავს სრულიად დამოუკიდებელ დომენურ, ბაზების მართვისა და ბიზნეს ლოგიკის ლეიერებს. პროგრამის ფარგლებში მაღალ დონეზეა გათვალისწინებული უსაფრთხოების საკითხები, დაცულია OOP პრინციპები და გამოყენებულია სოლუშენის ოპტიმიზაციისა და კოდის ხარისხის ამაღლების მიზნით კოდის ორგანიზების (დიზაინ) პატერნები, გამოყენებულია Advanced level programming პრინციპები, როგორიცაა **S.O.L.I.D**, აგრეთვე მსოფლიოში აღიარებული პროგრამირების პრინციპები KISS-**Keep it short and simple** და **D.R.Y.** პროგრამა მორგებადია სამომოავლო Unit Test-ებზეც.პროგრამული უზრუნველყოფა პასუხობს დღეისათვის არსებულ ყველა სტანდარტს და მოთხოვნას, კლიენტს არ უწევს დამატებითი პროგრამული უზრუნველყოფის ინსტალაცია საკუთარ მოწყობილობაზე და შეუძლია მარტივად და დაცულად იყოს ელექტრონული სწავლების პორტალის პროგრამული უზრუნველოყოფის  ვებ- მომხმარებელი.

**Annotation**

Users of the portal are administrators, teachers and students. The **teacher** by using the timetable (according to the type of days, hours, activity type and desired audience) sets the curriculum schedule of the course. The schedule of the subject is based on the administrator's confirmation. The teacher sets up a semester time schedule and gradually enriches it with the training material - relevant documents, exercises, control questions / tasks. There are a list of ratings on the web page where represents only students with higher marks, which encouraged them and gives a far more enthusiasm to young people. The teacher can also upload open news for every students, who are involved in the subject and answer the questions on the subject profile (questions can only be written on the subject).

**Student** can apply to any subject and receive access to subject timetable, news and questions. He can also can ask questions about the specific subject and answer the questions of the others students. All courses, which passed the student, would be written on his / her account. The web page shows the information about the skills of users and he’s score of specific course. In this case, it would be much easier for the employer to select potential employees. Accordingly, the student will also indicate own skills and experience in his report.

The **portal** is easy to use for any user with different knowledge and experience. Also, each user's rights are strictly defined by the site administrator. The system is built by the using of modern arctic paternal - ASP.NET MVC of the Microsoft NET Framework platform. This software contains full independent domain, bases management and business logic layers. In the making process we take on to our consideration all protecting services , based on OOP principles. We used best methods of code organizing (design) ,which gives us the opportunity to improve the efficiency of the our software solution and use of the advanced level programming principles, such as S.O.L.I.D, as well as the universal programming principles of KISS-Keep it short and simple and D.R.Y. The program is also compatible with the Dual Unit Test. The software responds to all existing standards and requirements, there is no need for the client to install the additional software on its own device. Accordingly, client can easily and securely use the software on web-based profile.

შესავალი

ELStudy პორტალი არის პლატფორმა  სტუდენტებისა და ლექტორებისთვის, რომელიც ემსახურება სასწავლო პროცესისა და მომხმარებლების კომუნიკაციის გამარტივებას, სტუდენტების ინტერესის გაღრმავებას სასურველ საგნებსა და სფეროებში და ამით სტუდენტების, როგორც სახელმწიფოსთვის ინტელექტუალური ღირებულების მქონე უმნიშვნეოვანესი ძალის, პროფესიული განვითარების ხელშეწყობას. აღსანიშნავია, რომ ჩვენს პლათფორმას უნიკალურობას სძენს ის ფაქტი, რომ ის მხოლოდ საგნების მშრალად გავლის ნაცვლად საშუალებას აძლევს სტუდენტებს ექაუნთში მითითებული უნარ-ჩვევებით, გამოცდილებით, სასურველი ტექნოლოგიების მიხედვით აპოვნინოს ის დამსაქმებელს, რაც ადმინისტრატორის მიერ შეტყობინებების დაგზავნის გზით განხორციელდება. პორტალზე დარეგისტრირება შეუძლია ნებისმიერს, საბოლოოდ მისი ექაუნთი დადასტურებული იქნება ადმინისტრატორისგან იმ შემთხვევაში, თუ ასეთი სტუდენტი/ლექტორი რეალურად არსებობს უნივერსიტეტის მიმდინარე სასწავლო პროცესში. ნებისმიერ მომხმარებელს საშუალება ექნება დაამატოს სასურველი დეტალები ექაუნთში - უნარჩვევები(skills), ტექნოლოგიები, რომლებსაც ფლობს, სამუშაო გამოცდილება და სხვა, რაც პორტალის ადმინისტრაციას დაეხმარება **დასაქმების კამპანიის** განხორცილებებაში მომხმარებლების სქილებისა და ცოდნის ამორჩევის მიხედვით.

პროგრამული უზრუნველყოფა სტუდენტებს მისცემს საშუალებას თვალი ადევნონთ სასურველ საგნებს, სიახლეებს, იქონიონ კომუნიკაცია ერთმანეთტან და პედაგოგებთან, ჩაერთვნენ შეფასებების რეიტინგში და მოხვდნენ წარატებულ სტუდენტთა სტატისტიკური მონაცემების სათავეებში.

პროგრამული უზრუნველყოფა შექმნილია მოწინავე არქიტეტქურულ პატერნზე- ASP.NET MVC გარემოში. პროგრამის ფარგლებში მაღალ დონეზეა გათვალისწინებული უსაფრთხოების საკითხები, დაცულია OOP პრინციპები და გამოყენებულია ოპტიმიზაციისა და კოდის ხარისხის ამაღლების მიზნითსხვადასხვა პატერნები და პრინციპები, ლეიერების დონეზე დაცულია პასუხისმგებლობათა განაწილებისა და მეთოდების დამოუკდებლობის მკაცრი ტექნიკა. სწორედ ამის მიხედვით შეგვიძლია მიმოვიხილოთ პროექტის არქიტექტურული სტრუქტურა მდულების მიხედვით:

**საერთო გამოყენების მოდული**

**საერთო გამოყენების მოდული - Common layer** არის აღნიშნული პროგრამული უზრუნველყოფის, როგორც მთლიანი სოლუშენის პროექტული ნაწილი - საერთო C# ბიბლიოთეკა, რომელიც შევქმენით პროგრამული არსების - საერთო გამოყენებისა და ლოგიკურად დაყოფილი დომენური მოდელების, ასევე ენუმერაციებისა და საერთო მიკროსერვისების ერთ ფენად გასაერთიანებლად. აღსანიშნავია, რომ დომენური მოდელები არის ის მთავარი ობიექტები, რომლებიც ასევე წარმოადგენს მონაცემთა ბაზის მთავარ არსებს. შესაბამისად, აქ ვიცავთ პროგრამირების DRY პრინციპებს და ვდილობთ ჩვენი მოდელები და მიკროსერვისები იყოს მაქსიმალურად მოქნილი და დამოუკიდებელი, რათა არ მოგვქიწიოს კოდის, კლასების, ფაილების განმეროება და და კოდი იყოს უფრო ნათელი, მარტივი. SOLID პროგრამირების პრინციპებვიდან ამ ლეიერში უმნიშვნელოვანესია Open/Closed Principle, რაც ნიშნავს, რომ დომენური მოდელები გაფართოებადია, მაგრამ დახურულია პირდაპირი წვდომისგან.

**მონაცემთა დამუშავების მოდული**

**მონაცემთა დამუშავების მოდული - Data layer** აგრეთვე წარმოადგენს C# ბიბლიოთეკას, რომლშიც ვაერთიანებთ ბაზასთან ურთიერთობისთვის საჭირო ფრეიმვორქებსა  და დატა -რეპოზიტორებს. აღსანიშნავია, რომ ამ ლეიერის მთავარი შტრიხი პასუხისმეგლობათა მკაცრი განაწილებაა და სრულდება აღიარებული SOLID პროგრამირების პრინციპებიდან უპირველესი - Single responsibility, რაც იმას ნიშნავს, რომ დატა-ლეიერის როლი მხოლოდ მონაცემების შენახვა, წაშლა, რედქტირება და გადმოცემაა. DRY პრინციპებისა და repository pattern გამოყენებით, კლასების სისუფთავისთვის, ვაცალკევებთ დომენურ მოდელებსა და მათთვის ბაზასთან ურთიერთობის საჭირო CRUD მეთოდებს და ვაერთიანებთ ჯენერიკ ან პირდაპირ რეპოზიტორებში დატა - ლეერის შიგნით.

**სერვისების- ბიზნეს ლოგიკის მოდული**

**სერვისების- ბიზნეს ლოგიკის მოდული - BL Service layer,** როგორც ასევე C# Library, წარმოადგენს პროგრამული უზრუნველყოფის გულს, სწორედ ამ ლეიერში თავმოყრილია სუფთად დალაგებული დომენური მოდელების, ბიზნეს მოდელებისა და პროცესების მართვისთვის საჭირო სერვისების ფუნქციონალი. ისევ და ისევ წამყვანია SOLID, DRY და KISS პრინციპები - დაბალი ურთიერკავშირი მეთოდებს შორის, მკაცრი პასუხისმგებლობები, საერთო კოდის მრავალჯერადად გამოყენება და keep it short and simple პრინციპების დაცვა საშუალებას გვაძლევს შევქმნათ იდეურად გამართული, დამოუკიდებელი მეთოდების მქონე ლოგიკური და ფიზიკური ფენა, რომელიც ემსახურება მომხმარებლის მხრიდან მოსული მონაცემების დამუშავებს და დატა-ლეიერისთვის გადაცემას და ასევე დატა-ლეიერიდან მიღებული მონაცემების სპეციფიური ბიზნეს ლოგიკით დამუშავებას და მომხმარებლისთვის მათ ჩვენებას. შესაბამისად, სერვისების ფენის პასუხისმგებლობაა Specific logics driven სტილის დაცვა და მონაცემების ბიზნეს ლოგიკით დამუშავება.

**კლიენტის მხარის მოდული**

**კლიენტის მხარის მოდული - Client side layer** იგივე მომხმარებლის მხარე არის  უკვე განხილული მოდულების მთავარი “მომხმარებელი”.  ვებ-ლეიერი არის პროექტი, რომლიც აღნიშნულ პროგრამულ უზრუნველყოფაში იძახებს და იყენებს აღწეილ ფენებს, კერძოდ კი მას პირდაპირი კავშირი აქვს სერვისების და საერთო ფენასთან, რამეთუ დატა-ლეირთან ურთიერთობის პასუხისმგებლობა მხოლოდ სერვისებს აქვს და ამით გამომძახებელ ლეირს აღარ ვტვირთავთ პასუხისმგებლობებით. შესაბამისად, ისევ და ისევ ვიცავთ დაბალი უერთიერთკავშირის, მაღალი შეჯაჭვულობის და პასუხისმგებლობათა მკაცრი გადანაწილების პრინცპიებს, რათა დავიცვათ პროგრამული უზრუნველყოფის ინჟინერიის მაღალი დონის სტანდარტები და ჩვენი პროგრამა იყოს მარტივად სამართავი, გაფართოებადი, ჩასწორებადი და გარჩევადი. ვებ მხარე წარმოადგენს რეალურ მომხმარებელსა და პროგრამის გულს- სერვისებთან ურთიერთობისთვის  აუცილებელ ხიდს და მისი პასუხისმგებლობაა მონაცემების გადაცემა და მათი მიღება, ვიზუალური გაფორმება და მომხმარებლის მაქსიმალური კომფორტის შექმნა. მნიშნველოვანია ასევე , რომ ჩვენი პროგრამული უზრუნვლყოფის მთავარი სასტარტო ვებ ფაილი ასევე უზრუნველყოფს ისეთი მნიშვნელოვანი პრინციპებისა და პატერნების გამოყენებას, როგორიცაა  Interface segregation და  Dependency inversion, რაც ნიშნავს, რომ ვიყენებთ წმინდა ინტერფეისულ კოდს, რომელიც ამარტივებს კოდის აღქმას და საშუალებას გვაძლევს სამომავლოდ ჩავატაროთ Unit Testing, ხოლო Dependency inversion, როგორც SOLID ის ერთ-ერთი მნიშვნელოვანი პრინციპი, საშუალებას გვაძლევს განვახორციელოთ მეთოდებს შორის დაბალი ურთიერთკავშირისა და დამოუკიდებლობის დონის ამაღლება. ვებ ლეირი მნიშვნელოვანია ასევე Security თვალსაზრისით, სწორედ აქ ხდება აუთენთიკაცია და ავტორიზაცია მომხმარებლების, შესაბამისად,მათთვის დაშვებული მოქმედებების კონტროლი, ასევე ღრმა უსაფრთხეობის პრინციპების გააზრება და იმპლემენტაცია.

**დასკვნა**

ჩვენი გუნდისთვის მნიშვნელოვანია როგორც მომხმარებლის კომფორტი და დადებითი შეფასება, ასევე პროგრამისტის შრომის ოპტიმიზირება და პროგრამის მაღალი ხარისხი, როგორც გაფართოებადი პროგრამული უზრუნველყოფის. შესაბამისად, ELStudy პლათმორმა მოხერხებულია როგორც მომხმარებლისთვის, ასევე პროგრამისტისვის. დასკვნის სახით შეილება ითქვას, რომ ელექტრონული სწავლების პორტალი ELStudy  დაეხმარება სტუდენტებს მიიღონ მეტი ინფორმაცია და ამასთან ერთად მეტი სტიმული განათლების სფეროში და ამ გზით უფრო დაუახლოვდნენ დასაქმების რეალურ პერსპექტივებს, შეძლონ მიღებული ცოდნის ინვესტრება სასარგებლო საქმიანობაში, იქონიონ კონტაქტი ლეტქორებან , ხოლო ლექტორბმა შეძლონ უფრო ფაქიზად გახსნან სწავლას მოწყურებული თითოეული  ახალგაზრდის  მისწრაფებები და ხელი შეუწყონ ახალგაზრდობის, როგორც ერის მამოძრავებელი ძალის, პროფესიულ წინსვლას.

## **გამოყენებული მასალა**

* ლიტერატურა
* Pro ASP.NET MVC 5- Adam Freeman
* Design Patterns – Erich Gamma ,Richard Helm, Ralph Jonson
* Pro ASP.NET Core MVC –Adam Freeman

* ბმულები:
* <https://docs.microsoft.com/en-us/aspnet/core/>
* <https://stackoverflow.com/>
* <https://www.w3schools.com/>

## 

## დანართები

* პროექტის ბმულები:
* https://www.El-Study.ge