

ივანე ჯავახიშვილის სახელობის თბილისის  
სახელმწიფო უნივერსიტეტი

ზუსტ და საბუნებისმეტყველო მეცნიერებათა ფაკულტეტი

საბაკალავრო პროგრამის „კომპიუტერული მეცნიერება“

ჯგუფური პროექტი

ინტრანეტ ტექნოლოგიის დანერგვა სასწავლო პროცესში

ხელმძღვანელი: ასისტენტ-პროფესორი  
პაპუნა ქარჩავა

პროექტის შემსრულებლები:

- |                         |                             |
|-------------------------|-----------------------------|
| 1. გელაშვილი გიორგი     | 11. სიმონიშვილი ნათია       |
| 2. გულისაშვილი თორნიკე  | 12. ფიცხელაური ელენე        |
| 3. გელბახიანი ომარი     | 13. ქართველიშვილი გიორგი    |
| 4. გელიტაშვილი ელენე    | 14. ქობალია ირაკლი          |
| 5. გვასალია ალექსანდრე  | 15. შალამბერიძე კონსტანტინე |
| 6. გურწიშვილი გიორგი    | 16. ჩიქოვანი ცოტნე          |
| 7. დაუთაშვილი შალვა     | 17. ჩხაიძე გიორგი           |
| 8. დეკანოიძე ერეკლე     | 18. წერუაშვილი გიორგი       |
| 9. ლალუაშვილი ვახტანგი  | 19. ჯიქურაშვილი ლევანი      |
| 10. მჭედლიშვილი ტარიელი | 20. ჯიჯიეიშვილი ანა         |

თბილისი  
2015 წელი

## ანოტაცია

პროექტის ძირითად მიზანს წარმოადგენს ინტრანეტ ტექნოლოგიის დანერგვა სასწავლო პროცესში, რაც გულისხმობს თანამედროვე ტექნოლოგიების გამოყენებას, კომფორტული სასწავლო გარემოს შექმნასა და სასწავლო პროცესის გამარტივებას.

ჩვენი ამოცანა წარმოადგენდა შეგვექმნა პროგრამული უზრუნველყოფა, რომელიც ორიენტირებული იქნებოდა სასწავლო პროცესის მართვაზე: საგნის, სტუდენტების მართვაზე, სტუდენტებსა და პედაგოგებს შორის კავშირის გამარტივებაზე და ა. შ.

აპლიკაცია შექმნილია ისეთ დონეზე, რომ მოხდეს ყველაფრის ავტომატიზაცია. იქნება ეს საგნების არჩევა თუ სალექციო მასალების განთავსება. გარდა ამისა, აპლიკაციაში გათვალისწინებულია სტუდენტების აღრიცხვის ავტონომური სისტემა, შესრულებული დავალებების მონიტორინგი და სტუდენტების საბოლოო შეფასების შესამდლებლობა.

## Annotation

The major goal of the project was to introduce intranet technology to the educational system, which means using modern technologies to create comfortable educational environment and making the educational process easy.

Our aim was to create software, which would be oriented towards managing educational process: managing subject, student, simplifying connectivity between students and teachers and etc.

The software is represented on the level, where everything is automatic. Whether it would be choosing the subjects or publishing lecture materials. Also, the software consists of automatic accounting of the students, monitoring of completed assignments and the ability to perform the final rating of each student.

## ს ა რ ჩ ე ვ ი

შესავალი .....	4
რა არის ინტრანეტი ? .....	5
რა არის ჩვენი ამოცანა?.....	6
დასახული ამოცანის გადაწყვეტის გზები .....	7
სერვერული ოპერაციული სისტემა .....	7
პოპულარობა .....	8
არსებული პროგრამული უზრუნველყოფები .....	8
ინსტალაციის სიმარტივე.....	8
უსაფრთხოება .....	8
რომელი ოპერაციული სისტემაა უკეთესი? .....	9
Full access VS No access.....	9
Security & Data Protection .....	9
სისტემური მოთხოვნები.....	10
Permissions and Encryption .....	10
ღირებულება .....	11
ვირუსები .....	11
აპლიკაციისთვის მონაცემთა ბაზის შემუშავება.....	12
აპლიკაციის შესაქმნელად პროგრამირების ენის შერჩევა და აპლიკაციის შექმნა .....	14
აპლიკაციისთვის სამუშაო ინტერფეისის შემუშავება და ზოგადი ფუნქციონალის განსაზღვრა.....	15
ადვილად ხემისაწვდომობა .....	15
გამოყენების სიმარტივე .....	15
უსაფრთხო .....	16
ავტორიზაციის სხვადასხვა დონე .....	16
გამოყენებული ლიტერატურა.....	22

## შესავალი

თანამედროვე სამყაროში აუცილებელია თანამედროვე ტექნოლოგიების გამოყენება საქმის და შესასრულებელი პროცესის გამარტივებასა და ეფექტურობისათვის. სასწავლო პროცესში თანამედროვე ტექნოლოგიების გამოყენება (როგორცაა ინტრანეტი) იძლევა სწავლების განვითარებისა და გაადვილების შესაძლებლობებს. თავად ინტრანეტი წარმოადგენს ინტერნეტ ტექნოლოგიების გამოყენებით აგებული შიდა კორპორაციული ქსელი, რომელიც შეიძლება შედგებოდეს ერთი ან რამდენიმე ქსელისაგან. მსგავსი ტექნოლოგიის გამოყენების მაგალითები სწავლების პროცესში უკვე არსებობს, მაგრამ გავრცელებული და დახვეწილი ჯერ-ჯერობით არ არის.

ჩვენი მიზანი ნათელია, თუმცა თავდაპირველად გასარკვევი იყო თუ რომელ ოპერაციულ სისტემას ავირჩევდით აპლიკაციის გასაშვებად. ოპერაციული სისტემის არჩევის დროს აუცილებელია გავითვალისწინოთ ამა თუ იმ ოპერაციული სისტემების პრიორიტეტები და მათი ნაკლები, მათი სისტემური მოთხოვნილებები, თავდაცვის მექანიზმები (რადგანაც საუბარი გვაქვს შიდა დაცულ ქსელზე, ყველაზე მნიშვნელოვან საკითხს წარმოადგენს თავდაცვა) და ა. შ.

მეორე ამოცანა მდგომარეობდა არჩეულ სერვერულ ოპერაციულ სისტემაზე რომელ ვებ სერვერს ავამუშავებდით. არჩევანი ბევრია, თუმცა ყველა ვებ სერვერი მულტიპლატფორმული არ არის, ამიტომაც ძეზნის არეალი შემცირდებოდა ოპერაციული სისტემის არჩევის შემდეგ. ვებ სერვერის არჩევის დროსაც მთელი რიგი ფაქტორების გათვალისწინება გვიწევს, როგორც გვიწევს სერვერის ოპერაციული სისტემის არჩევის დროს. ესენია: ვებ სერვერის მობილობა და სისწრაფე, ღირებულება, ეფექტურობა, მდგრადობა (ანუ იგივე თავდაცვა) და ა. შ.

ვებ სერვერის არჩევის შემდგომს ეტაპს წარმოადგენს თავად აპლიკაციის დასაწერად გამოყენებული პროგრამირების ენის არჩევა. რადგანაც ჩვენ ვაკეთებთ ვებ აპლიკაციას, მაშინ უნდა ვეძებოთ ისეთი პროგრამირების ენა, რომელიც ყველაზე პოპულარულია ვებ აპლიკაციების დეველოპმენტში და რომელსაც ყველაზე მეტი მომხმარებელი ყავს. აქაც, გასათვალისწინებელია ის მექანიზმები და ფუნქციონალი რასაც ენა შემოგვთავაზებს: არ უნდა იყოს ზედმეტად დატვირთული და არც პირიქით.

საბოლოო ამოცანა ამ ყველაფრის შემდეგ წარმოადგენს იმას თუ სად მოხდება მონაცემების შენახვა. სასურველია აირჩეს მონაცემთა ბაზის მართვის სისტემა რომელიც იქნება ვებ აპლიკაციებისთვის მორგებული, სწრაფი, შეიძლება უფასოც და რომელშიც ინტეგრირებული იქნება თავდაცვის მექანიზმები. ვებ აპლიკაციის ასაგებად გამოყენებული პროგრამირების ენა მოგვცემს საშუალებას მარტივად განვსაზღვროთ თუ რომელ მონაცემთა ბაზის მართვის სისტემასთან ვიმუშავებთ, რადგანაც უმეტესწინაა ვებ აპლიკაციებისთვის განკუთვნილი პროგრამირების ენები შეიცავენ მზა ფუნქციონალს კონკრეტულ მონაცემთა ბაზასთან სამუშაოდ.

## რა არის ინტრანეტი ?

ხშირად ისმის შეკითხვები: რა არის ინტრანეტი? რა უპირატესობების გამო მიმართავენ მას მსხვილი ორგანიზაციები? რა ნაკლოვანებები გაჩნია მას და ღირს თუ არა ამ ნაკლოვანებების გამო უარის თქმა ინტრანეტ ტექნოლოგიის გამოყენებას?

**ინტრანეტი** ეს არის ინტერნეტ ტექნოლოგიების გამოყენებით აგებული შიდა (private) კორპორაციული ქსელი, რომელიც შეიძლება შედგებოდეს ერთი ან რამდენიმე ქსელისაგან.

ფაქტიურად ინტრანეტი არის კორპორაციის შიდა web-პორტალი, რომელზეც წვდომა აქვს მხოლოდ კორპორაციის ავტორიზირებულ თანამშრომლებს. მისი უპირატესობებია:

- **მობილობა** - უნდა აღინიშნოს ის ფაქტი, რომ ინტრანეტში მუშაობა ხორციელდება სერვერზე გაშვებული აპლიკაციის მეშვეობით. ზუსტად იგი წარმოადგენს ცენტრალურ და ძირეულ კომპონენტს ინტრანეტში. მას შეიძლება ჰქონდეს ვებ-აპლიკაციის სახე, რაც მობილობას უწყობს ხელს. ეს გულისხმობს რომ ინტრანეტთან დაკავშირება შესაძლებელია ნებისმიერი სახლის მოწყობილობიდან: კომპიუტერიდან (მასზე არსებული ოპერაციული სისტემის მიუხედავად), მობილური ტელეფონიდან, ტაბლეთიდან ან სხვა სახის მოწყობილობიდან რომელშიც ჩაწერილია web-ბრაუზერი.
- **კომუნიკაციის სიმარტივე** - ინტრანეტის თანამშრომლებს მარტივად შეუძლიათ ერთმანეთში (ციფრული ინფორმაციის) მიმოცვლა, კომუნიკაცია.
- **შემცირებული დროის დანახარჯი** - იმ ფაქტორების გათვალისწინებით, რომ ყველაფერი ან უმეტესობა საქმიანობისა ხორციელდება ციფრულად, ცენტრალური აპლიკაციის მეშვეობით, ის დროთი დანახარჯი რომელიც საჭიროა რაიმე სამუშაოს შესასრულებლად შემცირებულია. აი თუნდაც წინა ფაქტორის გათვალისწინებით: თანამშრომლებს ერთმანეთთან კომუნიკაციისთვის აღარ უწევთ ფიზიკურად გადაადგილება და დროის ზედმეტად დახარჯვა. ასევე, ამოცანის გადაწყვეტა ხდება არა ერთ არამედ რამდენიმე კომპიუტერზე.
- **შემცირებული ფინანსური დანახარჯი** - იხარჯება ნაკლები რესურსები თანამშრომლების სამუშაო პირობების უზრუნველსაყოფად. არსებობს იმის შესაძლებლობაც, რომ თანამშრომელი უკავშირდებოდეს ინტრანეტს საკუთარი სახლიდან (VPN სერვისის გამოყენებით) რა დროსაც სამუშაო ადგილის გამოყოფა აუცილებელი აღარაა.
- **განაწილებული რესურსები** - ის ინფორმაცია, რომელიც ინტრანეტში ინახება ადვილად ხელმისაწვდომია ინტრანეტის ნებისმიერი მომხმარებლებს (შესაბამისი უფლებების ან პრივილეგიების გათვალისწინებით).
- **თავდაცვა** - ჩვენ ვთქვით, რომ ინტრანეტთან (ანუ ინტრანეტში არსებულ ვებ-აპლიკაციასთან) წვდომა შეუძლიათ მხოლოდ ავტორიზირებულ პირებს, ინტრანეტის მომხმარებლებს და არა სხვებს, რაც უზრუნველყოფს მონაცემთა დაცულობას.

ინტრანეტ ქსელის გამოყენებას რათქმაუნდა გააჩნია თავისი ნაკლოვანებები. ნაკლოვანებად შეიძლება ჩაითვალოს ის, რომ ინფორმაცია ხელმისაწვდომია ნებისმიერი მომხმარებლისათვის, რაც თავის მხრივ ნიშნავს, არასწორი ინფორმაციის სწრაფ გავრცელებას კორპორაციის თანამშრომლებს შორის. ასევე ნაკლოვანებად შეიძლება ჩაითვალოს რესურსის გაზიარება და სხვა. მაგრამ მსგავსი ნაკლოვანების გამო არ ღირს ინტრანეტ ქსელის გამოყენებაზე უარის თქმა. ამ ნაკლოვანებებს მნიშვნელოვნად გადაწონის ის უპირატესობები რასაც იძლევა ინტრანეტ ქსელის გამოყენება.

## რ ა რ ი ს ჩ ვ ე ნ ი ა მ ო ც ა ნ ა ?

**ჩვენი ამოცანაა ინტრანეტ ტექნოლოგიის დანერგვა სწავლის პროცესის მართვაში.** ტრადიციული სწავლების პროცესი გამართლებული და მისაღებია, მაგრამ თანამედროვე სამყარო გვთავაზობს ალტერნატიულ, უფრო გამარტივებულ და ხელმისაწვდომ მეთოდებს თანამედროვე ტექნოლოგიების გამოყენებით, როგორცაა ინტრანეტი.

ჩამოვყალიბოთ თუ რატომ არის ეს კარგი იდეა და რა უპირატესობა შეიძლება მას გააჩნდეს სასწავლო პროცესში?

ინტრანეტის გამოყენება სასწავლო პროცესში გვაძლევს შემდეგ უპირატესობებს:

- გამარტივებული სწავლების პროცესი;
- იხარჯება ნაკლები რესურსი;
- იხარჯება ნაკლები დრო;
- გვეძლევა მონიტორინგის შესაძლებლობა;
- იძლევა მეტ შესაძლებლობებს.

ის უპირატესობები, რომლებიც ინტრანეტს გააჩნია რჩება, და მას ემატება კონკრეტული შესაძლებლობები, რასაც ის გვთავაზობს სწავლების პროცესის წარმართვისათვის.

ახლა განვიხილოთ თუ რა შესაძლებლობები შეიძლება შესთავაზოს ინტრანეტმა სტუდენტს:

- პირადი სივრცე;
- საგნების არჩევის შესაძლებლობა;
- სალექციო მასალა;
- შესაბამისი დავალებები;
- ლექტორთან კომუნიკაციის შესაძლებლობა;
- ასევე, გაცდენილი ლექციის მოსმენის შესაძლებლობა;

ვისაუბროთ ლექტორის შესაძლებლობებზე:

- აქაც აღსანიშნავია - პირადი სივრცე;
- სალექციო მასალის დამატების, რედაქტირებისა და წაშლის შესაძლებლობა;

- სტუდენტების მიერ შესრულებული დავალებების დათვალიერება, შეფასება;
- სტუდენტებთან კომუნიკაციის შესაძლებლობა
- სალექციო ან არასალექციო განცხადებების გამოქვეყნების შესაძლებლობა
- შესაბამისი ლექციის ვიდეო ჩანაწერის დადების შესაძლებლობა

(ეს არის შესაძლებლობების არასრული ვარიანტი, რომელსაც შეიძლება დაემატოს ან ჩამოშორდეს ზოგიერთი თვისება. რა დაემატება და რა ჩამოშორდება ეს უკვე ადმინისტრაციულ საკითხს წარმოადგენს)

## დასახული ამოცანის გადაწყვეტის გზები

ბუნებრივია ყოველი დასმული ამოცანა საჭიროებს გადაწყვეტის მეთოდებს. ჩვენს შემთხვევაში ამოცანა დაიყო 4 ნაწილად. ესენია:

თავდაპირველად, საჭიროა ყურადღება დავუთმოთ შემდეგ საკითხებს:

- სერვერული ოპერაციული სისტემა შერჩევა;
- აპლიკაციისთვის მონაცემთა ბაზის შემუშავება;
- აპლიკაციის შესაქმნელად პროგრამირების ენის შერჩევა და აპლიკაციის შექმნა;
- აპლიკაციისთვის სამუშაო ინტერფეისის შემუშავება და ზოგადი ფუნქციონალის განსაზღვრა

### სერვერული ოპერაციული სისტემა

არსებობს შემდეგი სერვერული ოპერაციული სისტემები:

- **Windows**
- **Linux**
- **Macintosh**
- **Solaris**

დღეს-დღეობით საქართველოში ყველაზე პოპულარ ოპერაციული სისტემას წარმოადგენს **Linux** და **Windows**, არჩევანის გაკეთება მოგვიწია ამ ორ ოპერაციულ სისტემას შორის.

შევადაროთ ეს ორი ოპერაციული სისტემა შემდეგი კრიტერიუმების მიხედვით

- პოპულარობა
- არსებული პროგრამული უზრუნველყოფები
- ინსტალაციის სიმარტივე
- უსაფრთხოება
- სისტემური მოთხოვნები

## პოპულარობა

---

- დღეს-დღეობით Windows OS უფრო მეტი მომხმარებელი ყავს ვიდრე GNU/Linux. ასეთი პოპულარობა სავარაუდოდ გამოიწვია იმ ფაქტმა რომ ძირითადად ყველა პერსონალური კომპიუტერზე, რომელიც დაინსტალირებული არის Windows OS. თუმცა ეს ტენდენცია ნელ ნელა იცვლება. მაგალითად: ASUS Eee და Dell წარმოების კომპიუტერები შეიგიძლიათ შეიძინოს GNU/Linux Os-ით.
- ამ დრომდე ყველა GNU/Linux ოპერაციული სისტემა გამოდიოდა შედარებით ნაკლებად "user-friendly" ვიდრე Windows-ი. თუმცა Ubuntu-ს შემქმნელებმა დახარჯეს დიდი ძალისხმევა სისტემის გრაფიკულ მხარეზე, Windows და Apple Mac OS X-ზე დაყრდნობით მათ შექმნეს ახალი სამუშაო გარემო.
- Windows 8-ის არაწარმატებული გამოსვლის შემდგომ, იმ მომხმარებლებისთვის, რომლებისთვისაც Windows გახდა მოსაბეზრებელი Ubuntu აღმოჩნდა შესანიშნავი ალტერნატივა, რომელიც სრულიად უფასოა და "open source"-ია.

## არსებული პროგრამული უზრუნველყოფები

---

GNU/Linux ერთ-ერთ პრობლემას წარმოადგენს პროგრამების ნაკლებობა. თუმცა ძირითადი პროგრამები GNU/Linux არის უფასო და მათი რაოდენობა აჭარბებს Windows-ის პროგრამების რაოდენობას.

## ინსტალაციის სიმარტივე

---

ერთ ერთი მნიშვნელოვანი ასპექტი რის გამოც GNU/Linux OS ხშირად გაკიცხავდნენ იყო ინსტალაციის პროცესი. მომხმარებლებს როგორც წესი, ჭიდებოდათ კომპლექსური ცოდნის გამომჟღავნება ინსტალაციის პროცესში.

Windows წლების განმავლობაში სთავაზობდა მომხმარებლებს უნაკლო, მარტივ ინსტალაციისა და კონფიგურაციის პროცესს.

თუმცა დღეს-დღეობით Ubuntu-მ გაანახორციელა დიდი ნახტომი და მომხმარებლებს შესთავაზა უფრო მოხერხებული და მარტივი ინსტალაციის პროცედურა ვიდრე ამას მისი კონკურენტი აკეთებდა.

სამწუხაროთ სერვისი ღირს 10\$, მაგრამ ეს გაცილებით ნაკლებია Windows საინსტალაციო ლიცენზიებზე დახარჯულ ათასობით დოლარზე.

რაც შეეხება დრაივერებს, ძირითადად კომპიუტერული ტექნიკის მწარმოებლები უშვებენ დრაივერებს Windows OS-თვის, თუმცა ამ საკითხზე GNU/Linux-ზე სისტემის შემქმნელებმა ბევრი იმუშავეს და მოახერხეს მსოფლიოს გაოცება იმით რომ Linux-ის ინსტალაციის დროს დრაივერების ძირითადი ნაწილი ავტომატურად ინსტალირდება.

## უსაფრთხოება

---

2005 წელს აღმოაჩინეს 11,000-ზე მეტი სახეობის ვირუსი, რომლებიც აზიანებენ Windows OS-ს. Kaspersky Lab-ზე დაყრდნობით 1990 წლიდან აღმოჩენილია დაალოებით 800 ვირუსი, რომლებიც აზიანებენ GNU/Linux OS-ს. ეს სტატისტიკა განპირობებულია GNU/Linux-



ის პრივილეგირებული გამყოფი სტრუქტურული დიზაინით, რომელიც მომხმარებლის ჩარევის გარეშე არ აძლევს ვირუს მოქმედების საშუალებას. Windows კი - ვირუსებს auto-execution უნარის გამოყენების საშუალებას აძლევს მომხმარებლის ჩაურევლობის მუხედევად.

### **რომელი ოპერაციული სისტემაა უკეთესი?**

ერთი შეხედვით Windows უკეთესი OS-ია სხვებთან შედარებით. მის პოპულარობას ზდის ისეთი პროგრამული უზრუნველყოფების არსებობა, როგორცაა Chrome და სხვა. მაგრამ ბევრი მიზეზის გამო GNU/Linux უკეთესი ოპერაციული სისტემაა. უფასოა და არ მოითხოვს დიდ ტექნიკურ რესურსებს, რაც ერთ-ერთ დიდ დადებით მხარეს წარმოადგენს. შესაძლოა GNU/Linux OS გახდეს თქვენთვის Windows-ის ალტერნატივა, რომელიც აუმჯობესებს მისი კონკურენტის კონფორტსა და სიმარტივეს .

ახლა გადავიდეთ უშუალოდ სერვერებზე. განხილვისას შეიძლება გამოგვჩვენოს ზოგიერი ნიუანსი თუმცა ეს არ ცვლის მთლიან სურათს.

### **Full access VS No access**

კოდზე წვდომის შესაძლებლობა არის ერთ ერთი განმასხვავებელი ნიშანი GNU/Linux-სა და Windows-ს შორის. გსურთ Windows-ის კოდში ჩახედვა? ეს თითქმის წარმოუდგენელია.

ერთის მხრივ კოდზე სრული წვდომა შეიძლება ცუდი მიზნებით იქნას გამოყენებული, მეორეს მხრივ, სრული წვდომა კოდზე აძლევს მომხმარებელს სწრაფი განვითარებისა და ხარვეზების მოგვარების საშუალებას, რაც გზას უღობას ცუდი მიზნების მქონე დეველოპერებს. ხანდახან ყოფილა შემთხვევა როდესაც ამა თუ იმ Linux-ის აპლიკაციის კოდში დეველოპერს შეუტანია საჭირო ცვლილება და ამით ხშირი პრობლემა გადაუჭრია. Windows ამის საშუალებას არ იძლევა.

როდესაც ჯერი მიდგება სერვერის მენეჯმენტზე, Linux terminal კვლავ რჩება საუკეთესო ალტერნატივად ადმინისტრაციული (სამართვი) მიზნებისთვის.

command line შესაძლებელია Windows-შიც გამოვიყენოთ, მაგრამ არა ისე მოქნილად როგორ ამას GNU/Linux შემთხვევაში შეიძლება მივმართოთ.

### **Security & Data Protection**

GNU/Linux გააჩნია ჩაშენებული firewall-ები, რომლების ავტომატურად აქტიურდება და აკეთებს განახლებებს, ახდენს ფაილების შიფრაციას და უზრუნველყოფს მის დაცულობას. ავტორიზირებული დაშვება უზრუნველყოფს ვირუსების მიმართ სისტემის მდგრადობას.

## სისტემური მოთხოვნები

Windows Server 2012		
მოწყობილობა	მინიმუმი მოთხოვნა	სარეკომენდაციო მოთხოვნა
CPU	1.4Ghz(64 bit processor) 1.3Ghz(32bit processor)	3.1Ghz(64 bit processor) 2.6Ghz(32 bit processor)
Memory (RAM)	2GB	8GB
Hard Disks	60GB	160GB
Ubuntu Server 14.04		
მოწყობილობა	მინიმუმი მოთხოვნა	სარეკომენდაციო მოთხოვნა
CPU	1Ghz(64 bit processor) 700Mhz(32bit processor)	1.75Ghz(64 bit processor) 1.4Ghz(32 bit processor)
Memory (RAM)	512MB	1GB
Hard Disks	60GB	160GB

## Permissions and Encryption

თავდაცვისთვის პირველი ფენა არის უფლებების გაწერა. ნებართვები გამოიყენება კონტროლის დასაწესებლად სისტემისა და მომხმარებლის ფაილებზე.

**Encryption** გამოიყენება როგორც დამატებითი ფენის დაცვისთვის. **Linux** მოიცავს ჩაშენებულ ინსტრუმენტებს შიფრაციისთვის: **GPG, LUKS** და **Ecryptfs**.

Permission	Action	chmod option
read	(view)	r or 4
write	(edit)	w or 2
execute	(execute)	x or 1

There are also three types of user restrictions

User	ls output
owner	-rwx-----
group	----rwx---
other	-----rwx

## ღირებულება

იმის მიხედვით თუ Windows სერვერის რა ვერსია სჭირდება მომხმარებელს, ფასი დაახლოებით \$1000 მერყეობს, GNU/Linux შემთხვევაში კი ეს გაცილებით ნაკლებ ხარჯებთანაა დაკავშირებული, თუმცა Windows სთავაზობს მომხმარებელს შესანიშნავს სამუშაო გარემოს.

ძირითადი პუნქტები რასაც მომხმარებლები მიიჩნევენ დადებითად:

- ლინუქს სერვერი მუშაობს command line-ში რაც ნიშნავს ზემოდ სტაბილურობას
- თუ შენ აღმოაჩინე ხარვეზი ან შეუსაბამობა სისტემაში, Open Source გაძლევს საშუალებას შენით შეიტანო ცვლილებები, ლოდინის მაგივრად.
- software Windows სერვერებისთვის საკამოდ დიდ ხარჯებთანაა დაკავშირებული, განსაკუთრებით თუ ბევრი სერვერი გაქვს, ამ შემთხვევაში GNU/Linux საუკეთესო ვარიანტია
- ასევე Windows-ს ჭირდება ხშირი გადატვირთვა (reboot), გაშვებულია ანტივირუსი, რომელიც ტვირთავს სისტემას... ასევე, აღსანიშნავია დეფრეგმენტაციის გაწეილი პროცესი. GNU/Linux-ში გადატვირთვის გარეშე შესაძლებელია kernel-ის დაინსტალირება

## ვირუსები

როგორც ვიცით, ვირუსების მთავარი დანიშნულებაა კოდის ან კოდის ფრაგმენტის დაზიანება. აღსანიშნავია, რომ არ არსებობს მნიშვნელოვანი Linux ვირუსები. Linux სხვა სისტემებთან შედარებით, არანაკლებ სამიზნეს წარმოადგენს ვირუსებისთვის. ბევრი დიდი (ანუ ღირებული) ინტერნეტ საიტები იყენებს Linux ჰოსტინგს.

არ არის მართალი, რომ Linux ჩარმორჩება დროს ვირუსებთან ბრძოლაში. Linux დეველოპერებს არ დაუგნორირებიათ ვირუსები, უბრალოდ OS აგებულია ისე, რომ გამოირჩევა მაღალი გამძლეობით ვირუსების მიმართ, იმ ფაქტის თაანარსებობაში, რომ მისი კოდის გახსნილი ყველა მომხმარებლისთვის. უმეტესწილად, Linux ანტივირუსული პროგრამები ასკანერებენ Windows ვირუსებს.

ანტივირუსული პროგრამები Linux-ისთვის და ასევე ყველა HIDS (Host-based intrusion detection system) სისტემაში ორიენირებული არიან აღმოაჩინონ უკვე ცნობილი ვირუსები. სისტემა, ახალი ვირუსების წინააღმდეგ მას შემდეგაა დაცული, როდესაც მას აღმოაჩენენ, მოხდება კოდის დაუცველი ნაწილის გასწორება და სისტემაში განახორციელებს „*security updates* „.

ლინუქსის ანტივირუსის უარყოფითი მხარეები:

1. ის, პირველ რიგში, ასკანირებს Windows ვირუსებს.
2. არსებობს მაღალი მაჩვენებელი ცრუ სკანირების.
3. არ არის შექმნილი შექმნილი ცნობილი აქტიური Linux ვირუსების (so there is essentially nothing to detect).

ლინუქსის ანტივირუსის არსებობის მიზეზი:

- Linux სთავაზობს Windows მომხმარებელს ფაილურ ან მაილ სერვერებს.
- ფაილის გადაცემამდე უნდა დასკანერდეს ფაილები და მაილები.

საბოლოოდ მივედით იმ გადაწყვეტილებამდე რომ სერვერად ავირჩიეთ Linux, უფრო კონკრეტულად კი Ubuntu 14.04 ოპერაციული სისტემა.

## აპლიკაციისთვის მონაცემთა ბაზის შემუშავება

პასუხი მარტივია: MySQL

მიზეზი კი შემდეგში მდგომარეობს:

- **უფასოა** - MySQL-ი, როგორც ვიცით, არის უფასო და მასზე მოქმედებს GNU ლიცენზია. ასევე მისი source ფაილები ყველასთვის ხელმისაწვდომია ოფიციალურ ვებ-გვერდზე (ეს კარგია იმ შემთხვევაში თუ გვჭირდება ბირთვში რაიმე ცვლილების განხორციელება).
- **პოპულარობა** - MySQL-ს იყენებენ ცნობილი ვებ-გვერდები. ისეთები როგორცაა: Facebook, Google (თუმცა ძებნისთვის არა), Twitter, Flickr და YouTube.
- **სისწრაფე** - რაც შეეხება სისწრაფეს. აქ აუცილებელია ყურადღება გავამახვილოთ MySQL-ის მონაცემთა შენახვის ძრავებზე (იგივე Storage Engine-ზე). მოკლედ რომ ვთქვათ, მონაცემთა შენახვის ძრავი წარმოადგენს ძირითად პროგრამულ კომპონენტს, რომელსაც DBMS (Database Management System) მონაცემთა ბაზის მართვის სისტემა იყენებს რათა შექმნას, წაიკითხოს, განახლოს და წაშალოს (CRUD) მონაცემები ბაზიდან. (ჩამოთვლილი ძრავებიდან ზოგიერთი ტრანზაქციულია, ზოგიერთი არა. ზოგი კი სამაგალითოდ არის შექმნილი. მაგალითად, EXAMPLE და BLACKHOLE). ჩვენ გვაიტერესებს **MyISAM** და **InnoDB**.

ამ ორი ძრავიდან უძველესია MyISAM და იგი მართავს არატრანზაქციულ ცხრილებს, რაც განაპირობებს მონაცემების მაღალი სიჩქარით ჩაწერასა და ძიებას. ასევე მას გააჩნია FULLTEXT მონაცემთა ტიპი, რომელიც საშუალებას გვაძლევს დიდი მოცულობის ტექსტური ინფორმაცია ცხრილში სწრაფად მოვძებნოთ. ასევე აღსანიშნავია ისიც რომ იგი იყენებს ნაკლებ მოცულობას და ოპერატიულს განახლების ოპერაციის შესრულების დროს.

სამწუხაროდ ბლოკირება ხორციელდება ცხრილის დონეზე, რაც იმას ნიშნავს რომ მხოლოდ ერთ პროცესს შეუძლია ცხრილის განახლება დროის ერთ მომენტში. ეს ფაქტორი შემაფერხებელია ისეთი აპლიკაციებისთვის სადაც მომხმარებლებს ხშირად უწევთ ერთ ცხრილში არსებული მონაცემების შეცვლა (ან ჩამატება).

ასევე არ გააჩნია ტრანზაქციის ლოგირების სისტემა (Journaling), რაც დაკარგული ან დაზიანებული მონაცემების აღდგენას შეუძლებელს ხდის. (თავად Journaling არის ლოგირების მექანიზმი, რომელიც გულისხმობს იმას რომ ტრანზაქციის (ანუ ერთეული, ატომური ოპერაციის) შესრულებამდე ხდება ამ ტრანზაქციის შესახებ ინფორმაციის შესაბამის ლოგ ფაილში ჩაწერა.

განსხვავებით MyISAM-ისგან, InnoDB არის უფრო თანამედროვე ძრავა, რომელიც მართავს ტრანზაქციულ ცხრილებს. იგი ACID-ის (Atomicity, Consistency, Isolation, Durability) შესაბამისია, რაც გულისხმობს თითოეული ტრანზაქციის გარანტირებულად შესრულებას (საიმედოობას).

მას აქვს Foreign Key-ების მხარდაჭერა.

ბლოკირება, განსხვავებით MyISAM-ისგან, ხდება სტრიქონის დონეზე, რაც ერთდროულად ერთ ცხრილში მრავალი განახლების განხორციელების შესაძლებლობას გვამძლევს.

თუ ოპერაციის შესრულების დროს მოხდა შეცდომა, ყველა ცვლილება უკან დაბრუნდება (არატრანზაქციულში შეცდომის დროს ცვლილებების უკან დაბრუნება არ მოხდებოდა).

ერთ-ერთი უპირატესობა არის ის რომ ყოველი ტრანზაქცია ლოგირდება დამოუკიდებლად, რაც მონაცემთა აღდგენის უფრო საიმედო საშუალებას წარმოადგენს, რადგანაც მონაცემთა მდგრადობის შემოწმებას შესაძლებელს ხდის.

- არსებობს დამხმარე ხელსაწყო სახელად **MySQL Tuner**, რომელიც ერთგვარი მრჩეველის როლს ასრულებს და გვიჩვენებს თუ როგორ უნდა დავაკონფიგურიროთ MySQL ისე რომ ოპტიმალურად იმუშაოს. (რაც უფრო დიდი ხნის გაშვებული იქნება სერვერი, მით უფრო კარგ რჩევას მოგვაწოდებს ეს აპლიკაცია)
- დაცულია - MySQL გვთავაზობს შემდეგ თავდაცვის მეთოდებს, რომელთა გამოყენებაც მიზანშეწონილია:
  - მომხმარებლები და პრივილეგიები
  - მონაცემთა ბაზის ლოგები
  - კლიენტების დაბლოკვა
  - კრიპტოგრაფიული ფუნქციები
- მარტივია გამოსაყენებლად
- საუკეთესოა ვებ-გვერდებისა და ვებ-აპლიკაციების შემთხვევაში

**მომხმარებლები და პრივილეგიები** - როგორც სხვა ტიპი მონაცემთა ბაზებში, ასევე აქაც გვეძლევა უფლებების მინიჭებისა და შეზღუდვების დადების შესაძლებლობები. აუცილებელია გავითვალისწინოთ ის რომ ყოველ მომხმარებელს უნდა ჰქონდეს მინიმუმ სისტემაში შესვლის პრივილეგია მაინც.

**მონაცემთა ბაზის ლოგები** - აღსანიშნავია ორი სახის ლოგი:

- ზოგადი სახის (General Log), რომელშიც წერია ინფორმაცია ყველა ოპერაციის შესახებ.
- შეცდომების ლოგები (Error Log), რომელიც შეიცავს ინფორმაციას მხოლოდ შეცდომებზე.

**კლიენტის დაბლოკვა** - MySQL გვამძლევს შესაძლებლობას დავბლოკოთ ის მომხმარებლები რომლებიც წარუმატებლად რამდენჯერმე ცდილობენ ბაზასთან დაკავშირებას.

**კრიპტოგრაფიული ფუნქციები** - რა არის თავდაცვა კრიპტოგრაფიის გარეშე? არაფერი. MySQL-ში ჩვენ შეგვიძლია შევხვდეთ რამდენიმე ცნობილ ჰეშირებისა და დაშიფვრის ალგორითმებს:

- ჰეშირების: SHA() და ა. შ.
- დაშიფვრის: AES, DES

მიუხედავად იმისა რომ გვაქვს დაშიფვრის ალგორითმები, მათი გამოყენება ბაზის მხარეს არაა მიზანშეწონილი, რადგანაც შეცდომის შემთხვევაში დაუშიფრავი ინფორმაცია ზოგადი სახის ლოგის ფაილში გამოჩნდება.

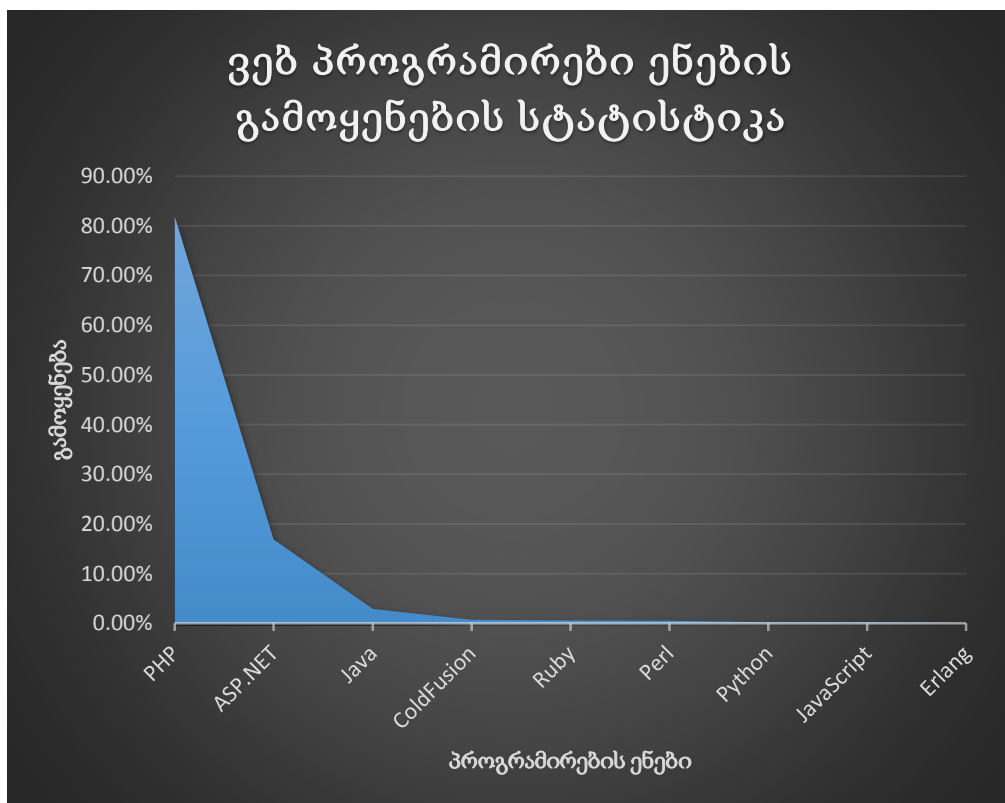
ერთ-ერთი მიზეზი ასევე არის სიმარტივე (დაყენებისა და ასევე გამოყენების).

- Ubuntu-ს ოპერაციული სისტემის შემთხვევაში დაყენება ხდება ტერმინალში ერთი ბრძანების გამოძახებით.
- გვაქვს გრაფიკული რედაქტორები ბაზის ადმინისტრირებისა და მართვის გასამარტიველად.

გრაფიკულ ინტერფეისებში შეიძლება აღვნიშნოთ შემდეგი ცნობილი და პოპულარული რედაქტორები: **Workbench, Navicat, Sequel Pro, HeidiSQL, phpMyAdmin.**

### აპლიკაციის შესაქმნელად პროგრამირების ენის შერჩევა და აპლიკაციის შექმნა

არსებობს ვებ აპლიკაციის შემუშავების მრავალი ენა. მოყვანილ სურათზე ნაჩვენებია ვებ-პროგრამირების ენების გამოყენების სტატისტიკა



სტატისტიკაზე დაყრდნობით ჩვენ არჩევანი გავაკეთეთ PHP-ზე.

- **PHP** შეიქმნა 1994 წელს და დაიწერა C ენაზე.
- თავდაპირველად არ გამოიყენებოდა როგორც **Server Side** ტექნოლოგია, შემდგომში შეიძინა მას ეს ფუნქცია
- დაემატა ბაზასთან კავშირი, შემდგომში ობიექტზე ორიენტირებული კონცეფცია.
- შემდგომში **SOAP** პროტოკოლთან ინტეგრაცია საშუალება მისცა ვერ სერვისების ჩამოყალიბებას **PHP**-ში.
- მუშაობს შემდგომ ვებ-სერვერებზე: **Apache, Microsoft IIS, Caudium, Netscape Enterprise Server.**
- ოპერაციული სისტემები: **UNIX (HP-UX,OpenBSD,Solaris,Linux),Mac OSX, Windows NT/98/2000/XP/2003**
- ბაზის მხარდაჭერა **Adabas D, dBase,Empress, FilePro (read-only), Hyperwave,IBM DB2, Informix, Ingres, InterBase, FrontBase, mSQL, Direct MS-SQL, MySQL, ODBC, Oracle (OCI7 and OCI8), Ovrimos, PostgreSQL, SQLite, Solid, Sybase, Velocis,Unix dbm**

## აპლიკაციისთვის სამუშაო ინტერფეისის შემუშავება და ზოგადი ფუნქციონალის განსაზღვრა

სამუშაო ინტერფეისი:

- ადვილად ხელმისაწვდომი
- გამოყენების სიმარტივე
- უსაფრთხო
- ავტორიზაციის სხვადასხვა დონე

## ადვილად ხელმისაწვდომობა

აპლიკაციაზე წვდომა შესაძლებელი იქნება ვებ-ბრაუზერის საშუალებით და არ იქნება საჭირო რაიმე დამატებითი მოწყობილობები.

აპლიკაცია იქნება წვდომადი მხოლოდ ინტრანეტის ქსელი მომხმარებლებისთვის. გარე წვდომა შესაძლებელი იქნება **VPN** კავშირის საშუალებით.

## გამოყენების სიმარტივე

აპლიკაცია გრაფიკული ინტერფეისის შესაქმნელად გამოვიყენეთ თანამედროვე user-friendly ტექნოლოგიები, რომელიც ორიენტირებულია მომხმარებელზე და მათ მოთხოვნებზე. ამავდროულად გავითვალისწინეთ ის ფაქტორიც, რომ იგი უნდა დაინერგოს სასწავლო პროცესში, მომხმარებლებს (სტუდენტებს და პედაგოგებს) კი მასთან შეხება უნდა ქონდეს დღეში რამოდენიმე საათის განმავლობაში.

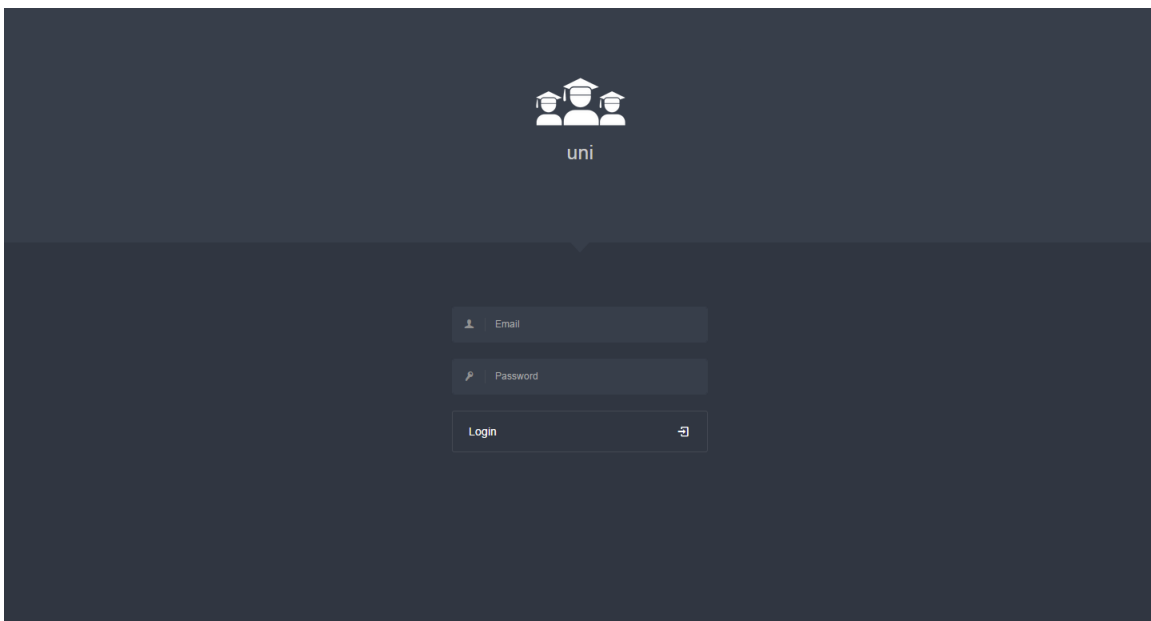
## უსაფრთხო

აპლიკაცია გაშვებული იქნება ინტრანეტ ქსელში, ამის გამო იგი ისედაც დაცული იქნება ჰაკერული თავდასხმისგან, თუმცა კოდი დაწერილია უსაფრთხოების ყველა ნორმის დაცვით და კოდში გამოყენებულია ყველა ტიპის ჰაკერული თავდასხმისგან დამცავი ფრაგმენტები

## ავტორიზაციის სხვადასხვა დონე

აპლიკაციას ეყოლება სამი კატეგორიის მომხმარებელი:

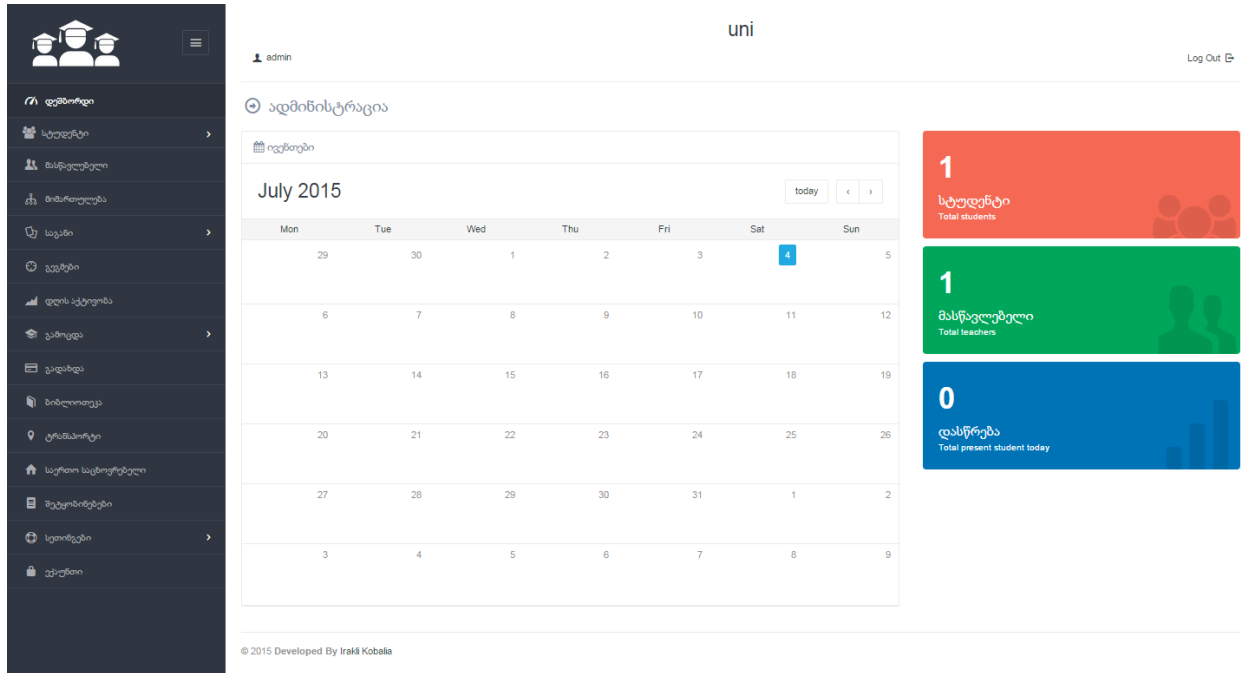
- ადმინისტრატორი
- პედაგოგი
- სტუდენტი



ავტორიზაციის დონეზე არ ხდება მომხმარებელთა განსხვავება, მომხმარებლის ტიპის განსაზღვრა ხდება მას შემდეგ რაც მომხმარებელი თავისი უნიკალური მომხმარებლის სახელითა და პაროლით გაივლის ავტორიზაციას.



## განვიხილოთ პირველი დონე: ადმინისტრატორი




ადმინისტრატორს აქვს შეუზღუდავი უფლებები. იგი განსაზღვრავს ვინ იქნება სისტემის მომხმარებელი (პედაგოგი, სტუდენტი), განსაზღვრავს მიმართულებებს, საგნებს, ლექციებსა და ჯგუფებს, ასევე აწარმოებს ქულების მონიტორინგსა და აქვს მათი რედაქტირების უფლება, ასევე ადმინს შეუძლია სტუდენტის სწავლის საფასურის მონიტორინგი. იგი ანთავსებს აპლიკაციაშია სტუდენტებისათვის და პედაგოგებისთვის საჭირო ინფორმაციას, გამოცდების განრიგს და ა.შ. შეუძლია შეტყობილებების დაგზავნა მომხმარებლებისთვის და სტუდენტთა ჩართულობის მონიტორინგი. ასევე აპლიკაცია საშუალებას გვაძლევს ვთარგმნოთ სხვადასხვა მოდულები ნებისმიერ ენაზე, რაღა აპლიკაცია მორგებული იყო უცხო ენოვანი სტუდენტებისთვის.

ადმინს აქვს საშუალება კალენდარზე დაამატოს ღონისძიები, რომელიც ყველა მომხმარებელთან ჩანს. სისტემას აქვს ელექტრონული ბიბლიოთეკა, სადაც განთავსებული იქნება სწავლებისთვის საჭირო ყველა ელექტრონული წიგნი.

განვიხილო ერთი, კონკრეტულად სტუდენტის დამატების მაგალითი:

☰ სტუდენტის დამატება

● დამატების ფორმა	
სახელი	<input type="text"/>
მამართლება	<input type="text" value="არაა"/>
წილი	<input type="text"/>
დაბადების თარიღი	<input type="text"/>
სქესი	<input type="text" value="არაა"/>
მისამართი	<input type="text"/>
ტელეფონი	<input type="text"/>
email	<input type="text"/>
პაროლი	<input type="text"/>
ფოტო	<div style="border: 1px solid gray; padding: 5px; text-align: center;"> Select image</div>
<a href="#">სტუდენტის დამატება</a>	

სტუდენტის, ან ნებისმიერი ინფორმაციის დამატება აპლიკაციაში საკმაოდ მარტივა. საჭიროა მხოლოდ სწორად შევასოს ადმინმა ველები, სტუდენტის დამატების შემთხვევაში ესენია: სახელი, გვარი, მიმართულება (რომელიც წინასწარ უკვე უნდა იყოს სისტემაში დამატებული), სემესტრი, დაბადების თარიღი, სქესი, მისამართი, ტელეფონი, ელ. ფოსტა, სისტემაში შესასვლელი პაროლი და სურათი.

## განვიხილოთ მეორე დონე: პედაგოგი

The screenshot shows a web application for a teacher. The top navigation bar includes the user name 'teacher' and a 'Log Out' button. The main content area is titled 'მასწავლებლის დემორადი' (Teacher's Dashboard) and features a calendar for July 2015. The calendar shows the following data:

Mon	Tue	Wed	Thu	Fri	Sat	Sun
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

On the right side, there are three summary cards:

- 1** სტუდენტი (Total students)
- 1** მასწავლებელი (Total teachers)
- 0** დასწრება (Total present student today)

© 2015 Developed By Irakli Kobalia

პედაგოგიც თითქმის იმავე უფლებებით სარგებლობს, რომლითაც ადმინი თუმცა შეზღუდვები მაინც გააჩნია მომხმარებლის ამ დონეს. პედაგოგს არ შეუძლია სისტემაში სხვა მომხმარებლის დამატება, ასევე მას არ აქვს უფლება კალენდარზე აღნიშნოს ღონისძიებები. თუმცა მას შეუძლია ნახოს სტუდენტთა სია, ნახოს საგნები, რომლსაც ის ასწავლის და ნახოს თავისი სამუშაო განრიგი (როდის აქვს ლექცია და სამუშაო ჯგუფები). მას ასევე შეუძლია ქონდეს მიმოწერა სტუდენტებთან და თითოეულ ჯგუფში გახსნილი ჩათითაც ისარგებლოს სტუდენტებთან დასაკავშირებლად. ასევე პედაგოგს შეუძლია სისტემაში სტუდენტთა აქტივობების ნახვა და მათი ქულების განსაზღვრა.

## განვიხილოთ მესამე დონე: სტუდენტი

The screenshot shows the 'uni' student dashboard. The user is logged in as 'student'. The main content area is titled 'სტუდენტის დემორადი' (Student Demo). It features a calendar for July 2015 and a notification feed.

**Calendar (July 2015):**

Mon	Tue	Wed	Thu	Fri	Sat	Sun
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

**Notifications:**

- 03:45 AM Today: Art Ramadani posted a status update
- 01:22 PM Yesterday: Job Meeting. You have a meeting at Laborator Office Today.

სტუდენტს სტისტემაში ცვლილებების შეტანის უფლება არ გააჩნია, მას მხოლოდ შეუძლია ნახოს კალენდარზე მიმაგრებული ივენთები, წაიკითხოს ადმინის მიერ გამოგზავნილი შეტყობინებები, ნახოს თავისი საგნები და პედაგოგები, გაეცნოს ქულებს, ნახოს ასევე თავისი მოსწრება. მოცეულ შემთხვევაში ჩანს, რომ სტუდენტს ჰყავს ერთი პედაგოგი.

The screenshot shows the 'uni' student dashboard with the user logged in as 'student'. The main content area is titled 'მასწავლებლების მართვა' (Teacher Management). It displays a table of teachers with columns for 'ლოტი' (Lottery), 'სახელი' (Name), and 'email'.

**Teacher List:**

ლოტი	სახელი	email
1	პაპუნა კარჩავა	papuna.karchava@tsu.ge

Showing 1 to 1 of 1 entries

© 2015 Developed By Irakli Kobala

მას ასევე შეუძლია ჰქონდეს მიმოწერა პედაგოგთან და თითოეულ ჯგუფში გახნილი ჩატი გამოიყენოს სასწავლო პროცესთან დაკავშირებული საკითხებზე. ხოლო უშალოდ ლექციის მიმდინარეობის პროცესში სტუდენტს სისტემაში ებლოკება ყველა საკომუნიკაციო საშუალება გარე სამყაროსთან რათა ორიენტირებული იყოს სწავლის პროცესზე და იხსნება უშვალო მიმოწერა სტუდენტსა და პედაგოგს შორის, მათ შეუძლიათ გაუზიარონ ერთმანეთს ფაილები, იქნება ეს დავალებები თუ დავალების ამოხსნა, თუმცა სტუდენტებთან ჩატი ამ დროს დახურულია. გარდა ამისა ლექციის პროცესში სისტემა ავტომატურად აღწერს სტუდენტის მოსწრებას ლექციაზე და აღრიცხავს მას თუ რამდენად ჩართულია სწავლის პროცესში.

სტუდენტს ასევე შეუძლია სისტემაში ჰქონდეს საკუთარი პირადი სივრცე, სადაც ექნება ატვირთული საჭირო ფაილები, იქნება ეს ლექციათა კურსი, ჯგუფურ მეცადინეობაზე გაკეთებული დავალება ან სხვა მასალა, რომელიც შემდგომში დასჭირება მას.

## გამოყენებული ლიტერატურა

<https://www.mysql.com>

<https://en.wikipedia.org/wiki/MySQL>

<https://dev.mysql.com/doc/refman/5.0/en/security.html>

<http://php.net>

<https://en.wikipedia.org/wiki/Intranet>

<https://help.ubuntu.com/lts/serverguide/>