

МОНГОЛ УЛСЫН ИХ СУРГУУЛЬ
ХЭРЭГЛЭЭНИЙ ШИНЖЛЭХ УХААН, ИНЖЕНЕРЧЛЭЛИЙН СУРГУУЛЬ
МЭДЭЭЛЭЛ, КОМПЬЮТЕРИЙН УХААНЫ ТЭНХИМ

Отгонбаатарын Ихбаяр

ДАДЛАГЫН АЖЛЫН ТАЙЛАН
(Internship Report)

Програм Хангамж (D061302)
Үйлдвэрлэлийн дадлагын тайлан

Улаанбаатар

2020 оны 09 сар

МОНГОЛ УЛСЫН ИХ СУРГУУЛЬ
ХЭРЭГЛЭЭНИЙ ШИНЖЛЭХ УХААН, ИНЖЕНЕРЧЛЭЛИЙН СУРГУУЛЬ
МЭДЭЭЛЭЛ, КОМПЬЮТЕРИЙН УХААНЫ ТЭНХИМ

ДАДЛАГЫН АЖЛЫН ТАЙЛАН
(Internship Report)

Програм Хангамж (D061302)
Үйлдвэрлэлийн дадлагын тайлан

Удирдагч: _____

Хамтран удирдагч: _____

Гүйцэтгэсэн: _____ О.Ихбаяр (17B1NUM2575)

Улаанбаатар

2020 оны 09 сар

Зохиогчийн баталгаа

Миний бие Отгонбаатарын Ихбаяр ”ДАДЛАГЫН АЖЛЫН ТАЙЛАН” сэдэвтэй судалгааны ажлыг гүйцэтгэсэн болохыг зарлаж дараах зүйлсийг баталж байна:

- Энэ ажлын аль нэг хэсгийг эсвэл бүхлээр нь ямар нэг их, дээд сургуулийн зэрэг горилохоор оруулж байгаагүй.
- Бусдын хийсэн ажлаас хуулбарлаагүй, ашигласан бол ишлэл, зүүлт хийсэн.
- Ажилд тусалсан бүх эх сурвалжид талархаж байна.

Гарын үсэг: _____

Огноо: _____

ГАРЧИГ

| | |
|---|----|
| УДИРТГАЛ | 1 |
| 1. СИСТЕЙН ТАНИЛЦУУЛГА | 2 |
| 1.1 Байгууллагын танилцуулга | 2 |
| 1.2 Байгууллагын үйл ажиллагаа, үйлчилгээ | 2 |
| 1.3 Компаний технологи | 2 |
| 1.4 Тухайн систем дээр өөрийн ажилласан хэсэг | 3 |
| 2. ИЖИЛ СИСТЕМИЙН СУДАЛГАА | 4 |
| 2.1 Ижил төстэй системүүдийн жишээ | 4 |
| 2.2 Системийг бүтээхэд ашигласан технологи, хэрэгсэл | 4 |
| 3. СИСТЕМИЙН ШААРДЛАГА | 6 |
| 3.1 Функционал ба функционал бус хэрэглэгчийн шаардлага | 6 |
| 3.2 Use-case диаграм | 9 |
| 4. СИСТЕМИЙН ШИНЖИЛГЭЭ ЗОХИОМЖ | 14 |
| 4.1 Өгөгдлийн сангийн диаграмм | 14 |
| 4.2 Өгөгдлийн сангийн холбоос болон хүснэгтүүдийн тайлбар | 14 |
| 4.3 Класс диаграмм | 18 |
| 4.4 Класс диаграммын холбоос болон классуудын тайлбар | 20 |
| 4.5 Үйл ажиллагааны урсгалын диаграмм | 25 |
| 5. ХЭРЭГЖҮҮЛЭЛТ | 27 |
| 5.1 Системийг бүтээхэд ашигласан технологи | 27 |
| 5.2 Use-case бүрийн хэрэглэгчийн интерфэйс | 28 |
| ДҮГНЭЛТ | 29 |
| НОМ ЗҮЙ | 31 |
| ХАВСРАЛТ | 36 |
| А. КОДЫН ХЭРЭГЖҮҮЛЭЛТ | 37 |

ЗУРГИЙН ЖАГСААЛТ

| | | |
|-----|--|----|
| 3.1 | Use Case | 10 |
| 4.1 | Өгөгдлийн сангийн диаграмм | 14 |
| 4.2 | Класс диаграмм | 19 |
| 4.3 | Үйл ажиллагааны урсгалын диаграмм | 25 |
| 5.1 | Хэрэглэгчийг системд шинээр бүртгэх хэсэг | 29 |
| 5.2 | Системд бүртгэлтэй хэрэглэгчид | 30 |
| 5.3 | Хэрэглэгчийн нууц үг солих хэсэг | 31 |
| 5.4 | Хэрэглэгч ямар үйлдэл хийж болохыг тодорхойлно | 32 |
| 5.5 | Хэрэглэгчид эрх өгнө | 33 |
| 5.6 | Сургуулийн мэдээлэл оруулах | 33 |
| 5.7 | Хичээлийн мэдээлэл оруулах | 34 |
| 5.8 | Хайлт хийх | 35 |

ХҮСНЭГТИЙН ЖАГСААЛТ

| | | |
|------|---|----|
| 4.1 | Users | 15 |
| 4.2 | Roles | 16 |
| 4.3 | Student_Info | 16 |
| 4.4 | Student_Lessons | 17 |
| 4.5 | Student_Schools | 18 |
| 4.6 | User class | 20 |
| 4.7 | Role class | 21 |
| 4.8 | Lesson class | 21 |
| 4.9 | Permission class | 22 |
| 4.10 | Student class | 23 |
| 4.11 | School class | 24 |
| 4.12 | Point class | 24 |
| 4.13 | Класс диаграммын холбоосуудын тайлбар | 26 |

Кодын жагсаалт

| | | |
|-----|--|----|
| A.1 | DEMO модулийн Interface | 37 |
| A.2 | selectDemoSchool функцийн хэрэгжүүлэлт | 38 |
| A.3 | Сургуулийн мэдээллийг хадгалах хэсэг | 38 |
| A.4 | Үйлдлийн эрх өгөх жишээ sql | 38 |
| A.5 | Дэлгэцийн кодчилол | 39 |
| A.6 | Дэлгэцийн кодчилол(module) | 40 |
| A.7 | Дэлгэцийн кодчилол(service) | 41 |

УДИРТГАЛ

Програм хангамж (D061302) хөтөлбөрийн 4-р түвшний оюутан Отгонбаатар овогтой Ихбаяр нь GrapeCity Монгол ХХК-д 21 хоногийн хугацаанд INTE401 хичээлийн индекстэй үйлдвэрлэлийн дадлагыг удирдамж, дадлагын ажлын төлөвлөгөөний дагуу GrapeCity Монгол ХХК-ны Систем интеграцийн хэлтэст дадлага хийсэн бөгөөд энэ хугацаандаа Javascript сан буюу фреймворк болох Angular.js болон Java ашиглан вэб орчинд front-end ба back-end хөгжүүлэлт хийн онолын мэдлэгээ баталгаажуулахын зэрэгцээ практик дадлага эзэмшиж өргөн хүрээний мэдлэгтэй болж чадсан бөгөөд судалгаа шинжилгээ хийн түүнийгээ практикаар хэрэгжүүлсэн.

Дадлагынхаа хугацаанд GrapeCity Монгол ХХК-ний хөгжүүлсэн NES платформ дээр суурь модулиудыг ашиглан дадлагынхаа хугацааны гуравны нэгт нь оюутны бүртгэл болон тухайн оюутантай холбоотой дүнгийн мэдээлэл оруулдаг вэб-д суурилсан систем хийсэн ба үлдсэн хугацаанд нь тухайн байгууллагын зарим системийн үйл ажиллагаатай танилцаж чанарын тест хийж үзсэн бөгөөд дотоод хөгжүүлэлтийн төлбөрийн системийн модуль дээр ажилласан. Дадлага хийх хугацаанд тулгарсан томоохон бэрхшээл болон асуудал гэвэл oracle өгөгдлийн бааз, ашиглаж буй технологи болон Angular.js-ийн талаар төдийлөн сайн мэдлэггүй байсан бөгөөд алхам тутамдаа шинэ зүйлтэй тулгарч зөв шийдлийг олох нь хүндрэлтэй байсан. Мөн өөрийн ажлын явцын талаар тайлан гаргаж газрын захиралдаа тайлагнадаг байсан.

Энэхүү тайланд дадлага хийсэн байгууллага, мөн дадлага хийх хугацаандаа юуг шинээр мэдэж авсан, түүнийгээ хэрхэн хэрэгжүүлсэн тухай өгүүлэх болно.

1. СИСТЕИЙН ТАНИЛЦУУЛГА

1.1 Байгууллагын танилцуулга

ГрэйпСити Монгол ХХК нь 2000 онд байгуулагдаж, банк санхүү ба төлбөрийн шийдлийн програм хангамж боловсруулах чиглэлээр дагнан ажилласаар ирсэн бөгөөд дотоодын энэ салбарын зах зээлд тэргүүлэгч юм.[3]

1.2 Байгууллагын үйл ажиллагаа, үйлчилгээ

Монголын 13 арилжааны банкны 12 нь ГрэйпСити Монгол ХХК-ны програм хангамжийн бүтээгдэхүүнийг ашигладагаас 10 банк энэхүү компаний боловсруулсан банкны суурь бүртгэлийн систем дээр ажиллаж байна. Активаар жигнэсэн үзүүлэлтээр харвал банкны програм хангамжийн дотоодын зах зээлийн 83 хувь нь ГрэйпСити Монгол ХХК-д ногдож байна. Бичил санхүүгийн байгууллагын нягтлан бодох бүртгэлд зориулсан ПОЛАРИС системийг 130 орчим ББСБ, ХЗХ, Бичил санхүүгийн байгууллага ашиглаж байна. ГрэйпСити Монгол ХХК нь банкны салбарт эзэлж байгаа энэхүү онцгой байр суурь дээрээ түшиглэж МОСТ платформыг нийтийн хүртээл болгоод байгаа билээ.

1.3 Компаний технологи

Банкны суурь систем хөгжүүлэхээс гадна банкны системтэй холбоотой өргөн цар хүрээтэй програм болон веб-д суурилсан програм хангамж, веб дизайн, гар утасны Android, iPhone, Windows гэх мэт платформууд дээр програм хангамжийг боловсруулдаг. GrapeCity Монгол ХХК нь Oracle болон RedHat корпорацийн албан ёсны түнш бөгөөд технологийн бүтээгдэхүүнийг дагалдах сервис үйлчилгээний хамт дамжуулан нийлүүлдэг эрх бүхий байгууллага юм. Одоогийн байдлаар NEXUS серверийг ашиглаж байгаа бөгөөд энэ нь сервер болон вэбийн кодчилолыг

хувилбар ахиулсан байдлаар нь ялган build хийх чадвартай төхөөрөмж юм. Мөн тухайн байгууллага нь SLACK chatbot дээр өөрийн API гарган серверээ өөрөө удирдах боломжийг олгосон байдаг билээ.

1.4 Тухайн систем дээр өөрийн ажилласан хэсэг

Миний хувьд ГрэйпСити Монгол ХХК-д дадлага хийх хугацаандаа тухайн байгууллагын хөгжүүлсэн NES платпормын модуль дээрх зарим нэг алдааг засаж, шаардлагатай нэмэлт сайжруулалтуудыг хийж банк дундын сүлжээний модулийн вэбийн frontend дээр Angular.JS 3 хувилбараас Angular.JS 8 хувилбарлуу шилжүүлэх ажлыг хийсэн бөгөөд backend дээр шаардлагатай нэмэлт өөрчлөлтийг хийж түүнийгээ тестэнд оруулан тестлүүлж нэмэлт сайжруулалт болон засварын ажлуудыг хийж гүйцэтгэсэн. Мөн тухайн платпормтой танилцах зорилгоор түүнд ашиглагддаг түүлүүдийг ашиглан оюутны бүртгэл хийгээд тухайн оюутны судалсан хичээлүүдийн дэлгэрэнгүй мэдээллийг харах вэб хийж үзсэн. Түүнчлэн вэб болон серверийн код бичилтэнд зориулагдсан GrapeCity компаний хөгжүүлсэн NESFullStackTool программыг дэлгэц хөрвүүлэлт болон серверийн код бичилтэндээ ашигласан нь ажлын бүтээмжийг ихээхэн нэмэгдүүлж чадсан.

2. ИЖИЛ СИСТЕМИЙН СУДАЛГАА

2.1 Ижил төстэй системүүдийн жишээ

2.1.1 *Most Money*

Most Money бол нийтийн хэрэглээний цахим төлбөр тооцооны үйлчилгээ. Гар утас, интернэт ашиглан өдөр тутмын төлбөр тооцоогоо заавал банк дээр очихгүйгээр цахим аргаар гүйцэтгэх боломжийг олгосон систем юм. [4]

2.1.2 *Гэрэгэ*

Гэрэгэ төлбөрийн систем нь QR код ашиглан төлбөр хийх технологид суурилсан хэрэглэгч хоорондын төлбөр тооцоог найдвартай, аюулгүй, түргэн шуурхай явуулах зорилготой систем юм.[5]

2.2 Системийг бүтээхэд ашигласан технологи, хэрэгсэл

2.2.1 *Angular.js*

Angular бол JavaScript фреймворк бөгөөд вэб аппликейшн бүтээхэд зориулагдсан.

Юуг мэдэж байх шаардлагатай вэ?

- HTML & CSS
- Basic JavaScript
- Programming fundamentals (Functions, conditionals, loop, etc..)

Мөн TypeScript, ECMAScript 2015 (ES6) програмчилалын хэлүүд, JavaScript code-г сервер тал дээр ажиллуулагч орчин болох Node.js технологи үүний сан зохицуулагч болох NPM багаж

ашиглалтын талаарх мэдлэгүүд хэрэгтэй. Angular CLI нь project шинээр үүсгэх, тест хийх, файл нэмэх, хөгжүүлэгч орчинд deployment хийх гэх мэт боломжуудыг агуулж эдгээрийг terminal/console орчноос командаар ажиллуулагч багаж юм.[1]

2.2.2 Node.js

Node.js гэдэг нь JavaScript дээр бичигдсэн кодыг ажиллуулж чадах орчин. PHP, Rails дээр хийж чадахгүй байсан их хэмжээний хүсэлтийг Node.js дээр асуудалгүй handle хийж чаддаг. Дан ганц вэб биш бүх төрлийн апп хийх боломжтой гэдгээрээ PHP-ээс давуу талтай байдаг бол хурд болон scale хийхэд хялбар гэдгээрээ Rails-ээс илүү гэж үздэг.[2]

2.2.3 JBOSS server

JBoss EAP нь opensource GNU LGPL /GNU lesser general public/ лиценз дээр бичигдсэн учраас эх кодыг нь ашиглах боломжтой. Гол нь гарч ирсэн бүтээгдэхүүнээ Wildfly, Jboss EAP гэж нэрлэх мөн Red hat, JBoss EAP, Wildfly-ын логог ашиглах нэрийг нь дурдахыг хориглодог учраас admin console, Welcome screen, Error screen зэрэг дээр байгаа нэрнүүд болон логог арилгаж өөрсдийн нэрээр солино. Би middleware-ийн нэрээ “Nes middleware” гэж өгсөн. Админ консоль нь JBoss-ын үндсэн кодноосоо тусдаа Hal-console нэртэй тусдаа project-ыг build хийгээд гарсан jar-аа тохирох зам дээр хуулаад module.xml дээр тухайн файлын нэрийг тавина.

3. СИСТЕМИЙН ШААРДЛАГА

3.1 Функционал ба функционал бус хэрэглэгчийн шаардлага

3.1.1 Функциональ шаардлага

1. Систем нь зөвхөн бүртгэлтэй хэрэглэгч нэвтрэх боломжоор хангах ёстой.
2. Хэрэглэгч хэрэв нууц үгээ мартсан бол систем нөөц хаяг руу холбоос явуулж баталгаажуулна.
3. Систем нь эрх бүхий хүн оюутан бүртгэх байдлаар хангах ёстой.
4. Систем нь бүртгэлд шаардлагатай хэсгийг бөглөхгүй орхигдуулсан бол сануулна.
5. Систем нь хэрэв хэрэглэгчид ойлгомжгүй зүйл байвал систем гарын авлагыг агуулсан байх ёстой.
6. Систем нь бүртгэгдсэн оюутан өөрийн анкетыг бөглөх боломжоор хангах ёстой.
7. Систем нь бүртгэлтэй оюутны боловсролын түвшинг бүртгэх ёстой.
8. Систем нь бүртгэлтэй оюутны судалсан болон судалж байгаа хичээлийн мэдээллийг бүртгэх ёстой.
9. Систем нь зөвхөн системд бүртгэлтэй оюутны мэдээллийг бүртгэх ёстой.
10. Систем нь оюутны овог, нэр, хүйс, нас, төрсөн огноог бүртгэх ёстой.
11. Систем нь оюутны бүртгэлийн мэдээллийг засах боломжтой байх ёстой.
12. Систем нь бүртгэгдсэн хичээлийн мэдээллийг засах боломжоор хангах ёстой.
13. Систем нь оюутны боловсролын мэдээллийг засах боломжоор хангах ёстой.

14. Оюутан нь зөвхөн өөрийн мэдээллийг харах, засах боломжтой байна.

15.

3.1.2 Функциональ бус шаардлага

1. Availability

- /AV01/ Оюутан Бүртгэлийн Систем нь 365 хоногийн турш 24 цаг найдвартай ажиллах ёстой.
- /AV02/ Системийн өөрчлөлт буюу шинэчлэлт хийх тохиолдолд шөнийн 12-оос үүрийн 6 цагийн хооронд хийнэ. Яаралтай үед шинэчлэлт хийгдвэл ихдээ 2 цагаас хэтрэхгүй хугацаанд өдөрт нэг л удаа хийгдэнэ.
- /AV03/ Уг системийн Амжилтгүй Болох Үеийн Хугацаа нь 168 цаг буюу 7 хоног байна.

2. Efficiency

- /EF01/ Системийг дахин ачааллах хугацаа 5 секундээс ихгүй байна.
- /EF02/ Систем нь нэгэн зэрэг 1000 хүртэлх хэрэглэгчийн хандалтын ачаалал даана.
- /EF03/ Систем нь хэрэглэгчийн хүсэлтэнд 5 секундийн дотор хариулна.
- /EF04/ Систем шинэ хэрэглэгч авч байхаар үргэлж нөөц өгөгдлийн багтаамжтай байна.
- /EF05/ Нэгэн зэрэг 4000 хүртэлх хэрэглэгч хандах боломжтой байх

3. Flexible Attribute

- /FA01/ Систем нь Англи, Монгол 2 хэл дээр хэрэглэх боломжтой.
- /FA02/ Систем нь боловсролын системтэй хоршиж ажиллана.
- /FA03/ Хэрэглэгчдэд харагдаж болзошгүй source code нь hard coded байх

4. Integrity

- /IN01/ Бусад хэрэглэгч Админы зөвшөөрөлгүйгээр Системд өөрчлөлт хийх боломжгүй.
- /IN02/ Хэрэглэгч бүр нөөц хаягаа бүртгүүлсэн байх
- /IN03/ Хэрэглэгч болон Админы мэдээлэл хадгалах болон дамжуулах арга хэлбэр нь үргэлж шифрлэлттэй байна.

5. Robustness

- /RO01/ Байгалийн эсвэл технологийн осол гэмтлээс үүдэж бий болсон системийн доголдолыг нөөц системээр дамжуулан сэргээнэ.
- /RO02/ Хэрэв гадны этгээд зөвшөөрөлгүй нэвтрэх оролдлого хийвэл тухайн хүний хандах IP хаягийг хязгаарлана.
- /RO03/ Хэрэглэгчийн нэр, нууц үг тохирох форматын дагуу биш үед хэргэлэгчид анхааруулга хүргэх.

6. Usability

- /US01/ Системтэй анх удаа ажиллаж байгаа хүнд систем ашиглах зааврыг харуулна.
- /US02/ Систем нь 16-70 хүртэлх насны хүмүүс ашиглах боломжоор хангагдсан байх ёстой.

7. Maintainability

- /MA01/ Систем нь хэрэглэгч бүртгэх үедээ түүнийг хэзээ, хэдэн цагт, хаана бүртгэснийг тэмдэглэж авна.
- /MA02/ Системтэй холбоотой аливаа гомдол маргааныг ажлын 5 өдрийн дотор холбогдох хүмүүстэй харилцаж, засч шийдвэрлэнэ.
- /MA03/ Сервер доголдсон үед түүнийг 1 – ээс 2 цагийн хооронд сэргээх ёстой.

- **/MA04/** Системийг хагас жил тутамд ядаж нэг удаа сайжруулан шинэчлэж байх ёстой.

8. Portability

- **/PO01/** Системд хадгалагдах аливаа цаг нь UTC буюу дэлхийн цагийн бүсийн координаттайгаар хадаглагдана.
- **/PO02/** Хууль эрх зүйн хүрээнд гажуудалгүй байх

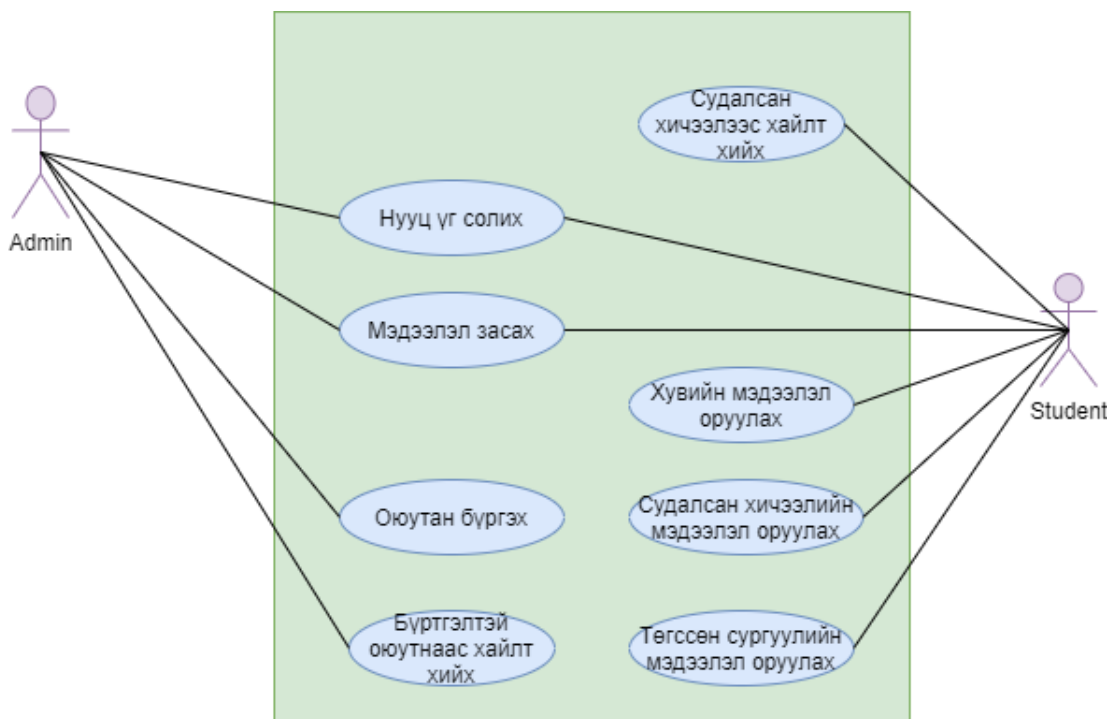
9. Reusability

- **/RE01/** Ерөнхий загвар , хэрэглэгчийн interface-г веб хуудас хийхэд ашиглах
- **/RE02/** Ерөнхий консэфтыг веб хуудас хийхэд ашиглах

3.2 Use-case диаграм

1. Оюутан бүртгэх

- **Зорилго:** Системд оюутан нэвтрэх боломжоор хангах
- **Төгсгөл нөхцөл амжилттай:** Оюутан бүртгэл амжилттай хийгдэх
- **Төгсгөл нөхцөл амжилтгүй:** Оюутан бүртгэл амжилтгүй хийгдэх
- **Тоглогч:** Админ
- **Тайлбар:**
 - Админ нь оюутан нэмэх хэсэгт хандана.
 - Админ нь оюутны овог нэр, цахим шуудан болон нууц үгийг бүртгүүлнэ.
 - Систем мэдээллийг шалгаж, бүрэн бол бүртгэнэ.
 - Цаашдаа оюутан нь цахим шуудан, нууц үгээр нэвтэрнэ.
- **Өргөтгөл:** Мэдээлэл бүрэн биш бол талбарыг бүрэн бөглөхийг шаардана.



Зураг 3.1: Use Case

2. Бүртгэлтэй оюутнаас хайлт хийх

- **Зорилго:** Оюутантай холбоотой лавлагаа авахын тулд цаг хугацааг хэмнэж үр дүнг нэмэгдүүлэх.
- **Төгсгөл нөхцөл амжилттай:** Үр дүн олдоно.
- **Төгсгөл нөхцөл амжилтгүй:** Үр дүн олдохгүй.
- **Тоглогч:** Админ
- **Тайлбар:**
 - Админ нь хайлтын хэсэгт хандана.
 - Админ нь оюутны овог нэр, цахим шуудан гэх мэт мэдээллийг оруулна.
 - Хайлт хийнэ.
- **Өргөтгөл:** Хайсан үр дүн олдохгүй бол мэдэгдэл харуулна гаргана.

3. Судалсан хичээлээс хайлт хийх

- **Зорилго:** Оюутан нь өөрийн судалсан хичээлийн талаар мэдээлэл авахын тулд цаг хугацааг хэмнэж үр дүнг нэмэгдүүлэх.
- **Төгсгөл нөхцөл амжилттай:** Үр дүн олдоно.
- **Төгсгөл нөхцөл амжилтгүй:** Үр дүн олдохгүй.
- **Тоглогч:** Оюутан
- **Тайлбар:**
 - Оюутан нь хичээл хайлтын хэсэгт хандана.
 - Оюутан нь хичээлтэй холбоотой тэмдэг мөр оруулна.
 - Хайлт хийнэ.
- **Өргөтгөл:** Хайсан үр дүн олдохгүй бол мэдэгдэл харуулна гаргана.

4. Хувийн мэдээлэл оруулах

- **Зорилго:** Системд оюутны мэдээллийг бүртгэлжүүлэх.
- **Төгсгөл нөхцөл амжилттай:** Амжилттай бүртгэгдэнэ.
- **Төгсгөл нөхцөл амжилтгүй:** Бүртгэл амжилтгүй болох.
- **Тоглогч:** Оюутан
- **Тайлбар:**
 - Оюутны хувийн мэдээллийн хэсэгт хандана.
 - Талбаруудын мэдээллийг бөглөнө.
 - Бүргэлийг хийнэ.
- **Өргөтгөл:** Талбарын мэдээлэл бүрэн биш бол бөглөхийг шаардана.

5. Оюутны мэдээлэл засах

- **Зорилго:** Алдаатай болон дутуу мэдээллийг засварлах.
- **Төгсгөл нөхцөл амжилттай:** Амжилттай өөрчлөлт бүртгэгдэнэ.
- **Төгсгөл нөхцөл амжилтгүй:** Өөрчлөлт амжилтгүй болох.
- **Тоглогч:** Оюутан, Админ
- **Тайлбар:**
 - Оюутны хувийн мэдээллийн хэсэгт хандана.
 - Өөрчлөлт оруулах талбарын мэдээллийг оруулна.
 - Өөрчлөлтийг бүртгэх.
- **Өргөтгөл:** Бүртгэл амжилтгүй хийгдэх.

6. Нууц үг солих

- **Зорилго:** Системын аюулгүй байдлын үүднээс оюутан нь нууц үгээ солино.
- **Төгсгөл нөхцөл амжилттай:** Амжилттай солигдсон бүртгэл хийгдэнэ.
- **Төгсгөл нөхцөл амжилтгүй:**
 - Өөрчлөлт амжилтгүй болох.
 - Хэтэрхий энгийн нууц үг ашиглах, шаардлага хангахгүй байх.
 - Хуучин нууц үгийн мэдээлэл таарахгүй байх.
- **Тоглогч:** Оюутан, Админ
- **Тайлбар:**
 - Нууц үг солих хэсэгт хандана.
 - Өмнөх нууц үг болон шинэ нууц үгээ оруулна.
 - Нууц үг солих.
- **Өргөтгөл:** Талбарын мэдээлэл бүрэн биш бол бөглөхийг шаардана.

7. Судалсан хичээлийн мэдээлэл оруулах

- **Зорилго:** Системд оюутны судалсан хичээлийн мэдээллийг бүртгэлжүүлэх.
- **Төгсгөл нөхцөл амжилттай:** Амжилттай бүртгэгдэнэ.
- **Төгсгөл нөхцөл амжилтгүй:** Бүртгэл амжилтгүй болох.
- **Тоглогч:** Оюутан
- **Тайлбар:**
 - Хичээл бүртгэлийн хэсэгт хандана.
 - Талбаруудын мэдээллийг бөглөнө.
 - Бүргэлийг хийнэ.
- **Өргөтгөл:** Талбарын мэдээлэл бүрэн биш бол бөглөхийг шаардана.

8. Төгссөн сургуулийн мэдээлэл оруулах

- **Зорилго:** Системд оюутны төгссөн сургуулийн мэдээллийг бүртгэлжүүлэх.
- **Төгсгөл нөхцөл амжилттай:** Амжилттай бүртгэгдэнэ.
- **Төгсгөл нөхцөл амжилтгүй:** Бүртгэл амжилтгүй болох.
- **Тоглогч:** Оюутан
- **Тайлбар:**
 - Төгссөн сургууль бүртгэлийн хэсэгт хандана.
 - Талбаруудын мэдээллийг бөглөнө.
 - Бүргэлийг хийнэ.
- **Өргөтгөл:** Талбарын мэдээлэл бүрэн биш бол бөглөхийг шаардана.

Table 4.1: Users

| Баганы нэр | PK | FK | Төрөл ба урт | Тайлбар нэр | Хоосон утга | Тайлбар |
|---------------|----|----|--------------|-------------------|-------------|---|
| user_id | ✓ | | number(8) | Student ID | NO | Хэрэглэгчийг илэрхийлэх дахин давтагдашгүй дугаар |
| first_name | | | varchar(50) | Student Firstname | NO | Хэрэглэгчийн системд ашиглагдах нэр |
| email | | | varchar(100) | Student email | NO | Хэрэглэгчийн системд нэвтрэхдээ ашиглах мэйл хаяг |
| password | | | varchar(200) | Student password | NO | Хэрэглэгчийн системд нэвтрэх нууц үг |
| role | | | varchar(50) | Student role | NO | Хэрэглэгчийн системд гүйцэтгэх үүрэг |
| created_date | | | DATE | Created date | NO | Бүртгэгдсэн огноо |
| created_by | | | number(8) | Created by | NO | Бүртгэсэн хэрэглэгчийн дугаар |
| modified_date | | | DATE | Modified date | NO | Өөрчлөгдсөн огноо |
| modified_by | | | number(8) | Modified by | NO | Өөрчлөлт хийсэн хэрэглэгчийн дугаар |

4.2.2 Roles

Энэ хүснэгтэнд системд гүйцэтгэх үүргүүдийн мэдээллийг хадгална.

Table 4.2: Roles

| Баганы нэр | PK | FK | Төрөл ба урт | Тайлбар нэр | Хоосон утга | Тайлбар |
|------------|----|----|--------------|-------------|-------------|---|
| id | ✓ | | number(8) | Role ID | NO | Хүснэгтийг илэрхийлэх дахин давтагдашгүй дугаар |
| name | | | varchar(50) | Role name | NO | системийн эрхийн нэр(админ, оюутан) |

4.2.3 Student Info

Энэ хүснэгтэнд системд бүртгэлтэй оюутны дэлгэрэнгүй мэдээллийг хадгална.

Table 4.3: Student_Info

| Баганы нэр | PK | FK | Төрөл ба урт | Тайлбар нэр | Хоосон утга | Тайлбар |
|-----------------|----|----|--------------|-------------|-------------|--|
| student_id | ✓ | | number(8) | Student ID | NO | Оюутны мэдээллийг илэрхийлэх дахин давтагдашгүй дугаар |
| user_id | | ✓ | number(8) | User ID | NO | Бүртгэлд харгалзах оюутны дугаар |
| sur_name | | | varchar(100) | Surname | YES | Ургийн овог |
| last_name | | | varchar(50) | Lastname | NO | Эцэг эхийн нэр |
| first_name | | | varchar(50) | Firstname | NO | Өөрийн нэр |
| gender | | | varchar(10) | Gender | NO | Хүйс |
| address | | | varchar(400) | address | NO | Гэрийн хаяг |
| register_number | | | varchar(10) | Register | NO | Регистерийн дугаар |
| dob | | | date | Birthday | NO | Төрсөн огноо |

4.2.4 Student Lessons

Энэ хүснэгтэнд системд бүртгэлтэй оюутны судалсан хичээлүүдийн мэдээллийг хадгална.

Table 4.4: Student_Lessons

| Баганы нэр | PK | FK | Төрөл ба урт | Тайлбар нэр | Хоосон утга | Тайлбар |
|---------------|----|----|--------------|---------------|-------------|---|
| lesson_id | ✓ | | number(8) | Lesson ID | NO | Хичээлийн мэдээллийг илэрхийлэх дахин давтагдашгүй дугаар |
| user_id | | ✓ | number(8) | User ID | NO | Бүртгэлд харгалзах оюутны дугаар |
| lesson_name | | | varchar(200) | Lesson name | NO | Хичээлийн нэр |
| seen_date | | | DATE | Seen date | NO | Судалсан огноо |
| point | | | number(4) | Point | NO | Дүн |
| created_date | | | DATE | Created date | NO | Бүртгэгдсэн огноо |
| created_by | | | number(8) | Created by | NO | Бүртгэсэн хэрэглэгчийн дугаар |
| modified_date | | | DATE | Modified date | NO | Өөрчлөгдсөн огноо |
| modified_by | | | number(8) | Modified by | NO | Өөрчлөлт хийсэн хэрэглэгчийн дугаар |

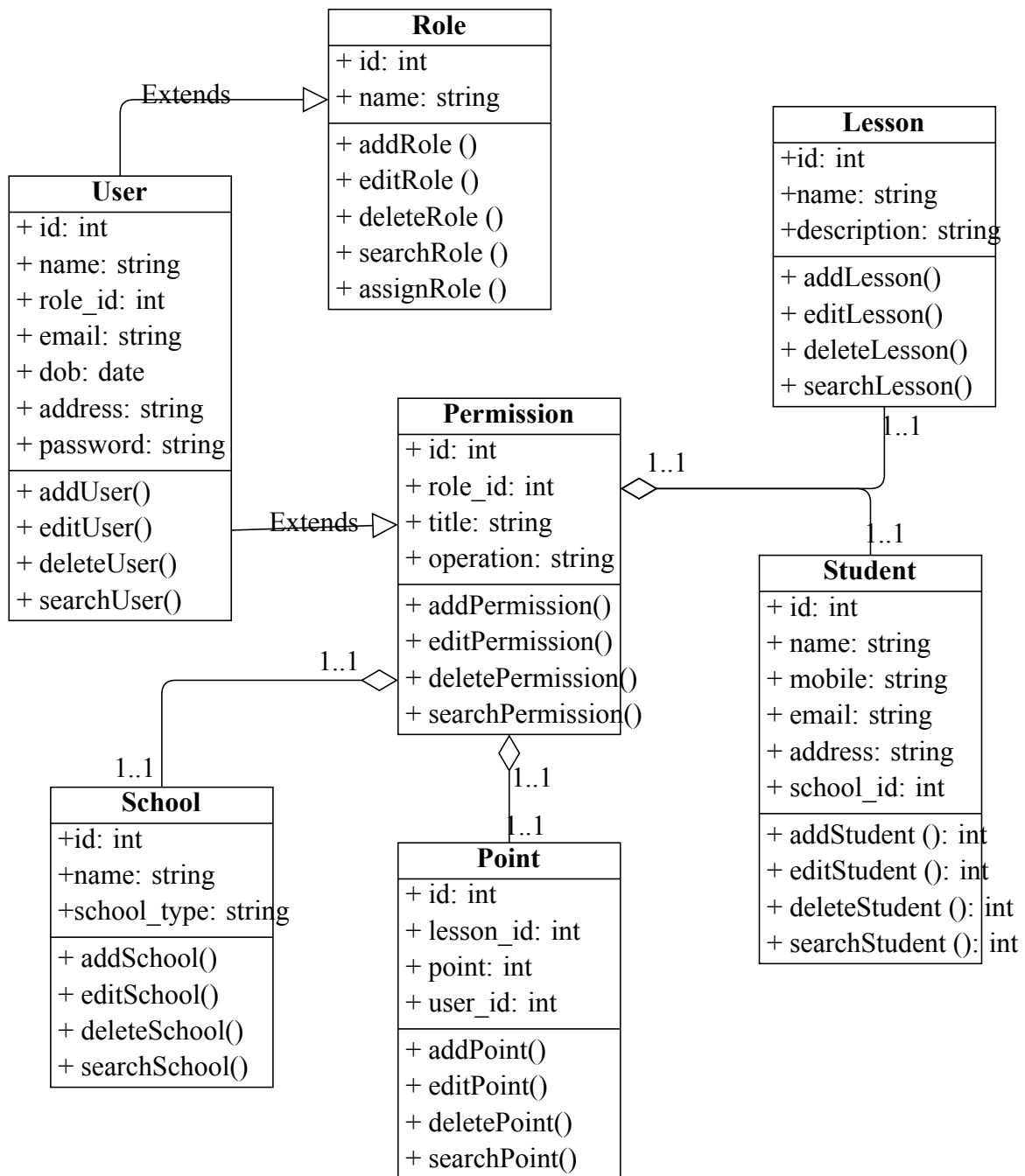
4.2.5 Student Schools

Энэ хүснэгтэнд системд бүртгэлтэй оюутны төгссөн сургуулийн мэдээллийг хадгална.

Table 4.5: Student_Schools

| Баганы нэр | PK | FK | Төрөл ба урт | Тайлбар нэр | Хоосон утга | Тайлбар |
|---------------|----|----|--------------|---------------|-------------|--|
| school_id | ✓ | | number(8) | School ID | NO | Төгссөн сургуулийн мэдээллийг илэрхийлэх дахин давтагдашгүй дугаар |
| user_id | | ✓ | number(8) | User ID | NO | Бүртгэлд харгалзах оюутны дугаар |
| start_date | | | DATE | Start date | NO | Элссэн огноо |
| end_date | | | DATE | End date | NO | Төгссөн огноо |
| grade_point | | | number(4) | Grade Point | NO | Голч дүн |
| created_date | | | DATE | Created date | NO | Бүртгэгдсэн огноо |
| created_by | | | number(8) | Created by | NO | Бүртгэсэн хэрэглэгчийн дугаар |
| modified_date | | | DATE | Modified date | NO | Өөрчлөгдсөн огноо |
| modified_by | | | number(8) | Modified by | NO | Өөрчлөлт хийсэн хэрэглэгчийн дугаар |

4.3 Класс диаграмм



Зураг 4.2: Класс диаграмм

4.4 Класс диаграммын холбоос болон классуудын тайлбар

4.4.1 User class

Энэ класс нь систем хэрэглэгчийн үндсэн мэдээллийг агуулах ба шинж бүрд өөрийн get болон set метотууд байна.

Table 4.6: User class

| Нэр | Шинж | Метод | Төрөл | Тайлбар |
|------------|------|-------|--------|---|
| id | ✓ | | int | Хэрэглэгчийн давтагдашгүй дугаар |
| name | ✓ | | string | Хэрэглэгчийн нэр |
| role_id | ✓ | | int | Хэрэглэгчийн системд гүйцэтгэх үүрэг |
| email | ✓ | | string | Хэрэглэгчийн мэйл хаяг |
| dob | ✓ | | date | Хэрэглэгчийн төрсөн огноо |
| address | ✓ | | string | Хэрэглэгчийн гэрийн хаяг |
| password | ✓ | | string | Хэрэглэгчийн системд нэвтрэх нууц үг |
| addUser | | ✓ | | Системд хэрэглэгч бүртгэнэ |
| editUser | | ✓ | | Бүртгэлтэй хэрэглэгчийн мэдээлэл засварлана |
| deleteUser | | ✓ | | Бүртгэлтэй хэрэглэгчийг устгах |
| searchUser | | ✓ | | Бүртгэлтэй хэрэглэгчээс хайх |

4.4.2 Role class

Энэ класс нь систем хэрэглэгчийн үүргийг тодорхойлж өгнө.

Table 4.7: Role class

| Нэр | Шинж | Метод | Төрөл | Тайлбар |
|------------|------|-------|--------|--|
| id | ✓ | | int | Үүргийн давтагдашгүй дугаар |
| name | ✓ | | string | Системдэх үүргийн нэр |
| addRole | | ✓ | | Системд үүрэг бүртгэнэ |
| editRole | | ✓ | | Бүртгэлтэй үүргийн мэдээлэл засварлана |
| deleteRole | | ✓ | | Бүртгэлтэй үүргээс устгах |
| searchRole | | ✓ | | Бүртгэлтэй үүргээс хайх |
| assignRole | | ✓ | | Үүргүүдийг хувиарлаж өгнө |

4.4.3 Lesson class

Энэ класс нь тухайн хэрэглэгчийн судалж буй болон судалсан хичээлүүдийг тодорхойлж өгнө.

Table 4.8: Lesson class

| Нэр | Шинж | Метод | Төрөл | Тайлбар |
|--------------|------|-------|--------|--------------------------------|
| id | ✓ | | int | Хичээлийн давтагдашгүй дугаар |
| name | ✓ | | string | Хичээлийн нэр |
| description | ✓ | | string | Хичээлийн дэлгэрэнгүй мэдээлэл |
| addLesson | | ✓ | | Хичээлийн мэдээллийг бүртгэнэ |
| editLesson | | ✓ | | Хичээлийн мэдээлэл засварлана |
| deleteLesson | | ✓ | | Бүртгэлтэй хичээлээс устгах |
| searchLesson | | ✓ | | Бүртгэлтэй хичээлээс хайх |

4.4.4 *Permission class*

Энэхүү класс нь тухайн хэрэглэгчийн ажил үүргээс хамааран тодорхой хэмжээний хязгаарлалт хийхэд ашиглана.

Table 4.9: Permission class

| Нэр | Шинж | Метод | Төрөл | Тайлбар |
|------------------|------|-------|--------|-----------------------------------|
| id | ✓ | | int | Хязгаарлалтын давтагдашгүй дугаар |
| title | ✓ | | string | Хязгаарлалтын нэр |
| role_id | ✓ | | int | Системд гүйцэтгэх үүргийн дугаар |
| operation | ✓ | | string | Хэрэглэгчийн хүсэлтийн дугаар |
| addPermission | | ✓ | | Системд хязгаарлалт нэмнэ |
| editPermission | | ✓ | | Хязгаарлалтын мэдээлэл засварлана |
| deletePermission | | ✓ | | Хязгаарлалт цуцлах |
| searchPermission | | ✓ | | Хязгаарлалтуудаас хайх |

4.4.5 *Student class*

Энэ класс нь оюутны мэдээллийг агуулах ба шинж бүрд өөрийн get болон set метотууд байна.

Table 4.10: Student class

| Нэр | Шинж | Метод | Төрөл | Тайлбар |
|---------------|------|-------|--------|--|
| id | ✓ | | int | Оюутны давтагдашгүй дугаар |
| name | ✓ | | string | Оюутны нэр |
| school_id | ✓ | | int | Оюутны суралцаж буй сургуулийг илэрхийлэх дугаар |
| email | ✓ | | string | Оюутны мэйл хаяг |
| address | ✓ | | string | Оюутны гэрийн хаяг |
| addStudent | | ✓ | | Системд оюутан бүртгэнэ |
| editStudent | | ✓ | | Бүртгэлтэй оюутны мэдээлэл засварлана |
| deleteStudent | | ✓ | | Бүртгэлтэй оюутныг устгах |
| searchStudent | | ✓ | | Бүртгэлтэй оюутнаас хайх |

4.4.6 *School class*

Энэхүү класс нь сургуулийн бүртгэлийн мэдээллийг агуулах ба шинж бүрд өөрийн get болон set метотууд байна.

Table 4.11: School class

| Нэр | Шинж | Метод | Төрөл | Тайлбар |
|--------------|------|-------|--------|---|
| id | ✓ | | int | Сургуулийн давтагдашгүй дугаар |
| name | ✓ | | string | Сургуулийн нэр |
| school_type | ✓ | | string | Сургуулийн үйл ажиллагааны чиглэл болон төрөл |
| addSchool | | ✓ | | Системд сургууль бүртгэх |
| editSchool | | ✓ | | Сургуулийн мэдээлэл засварлана |
| deleteSchool | | ✓ | | Сургуулийн мэдээлэл устгах |
| searchSchool | | ✓ | | Бүргэгдсэн сургуулиудаас хайх |

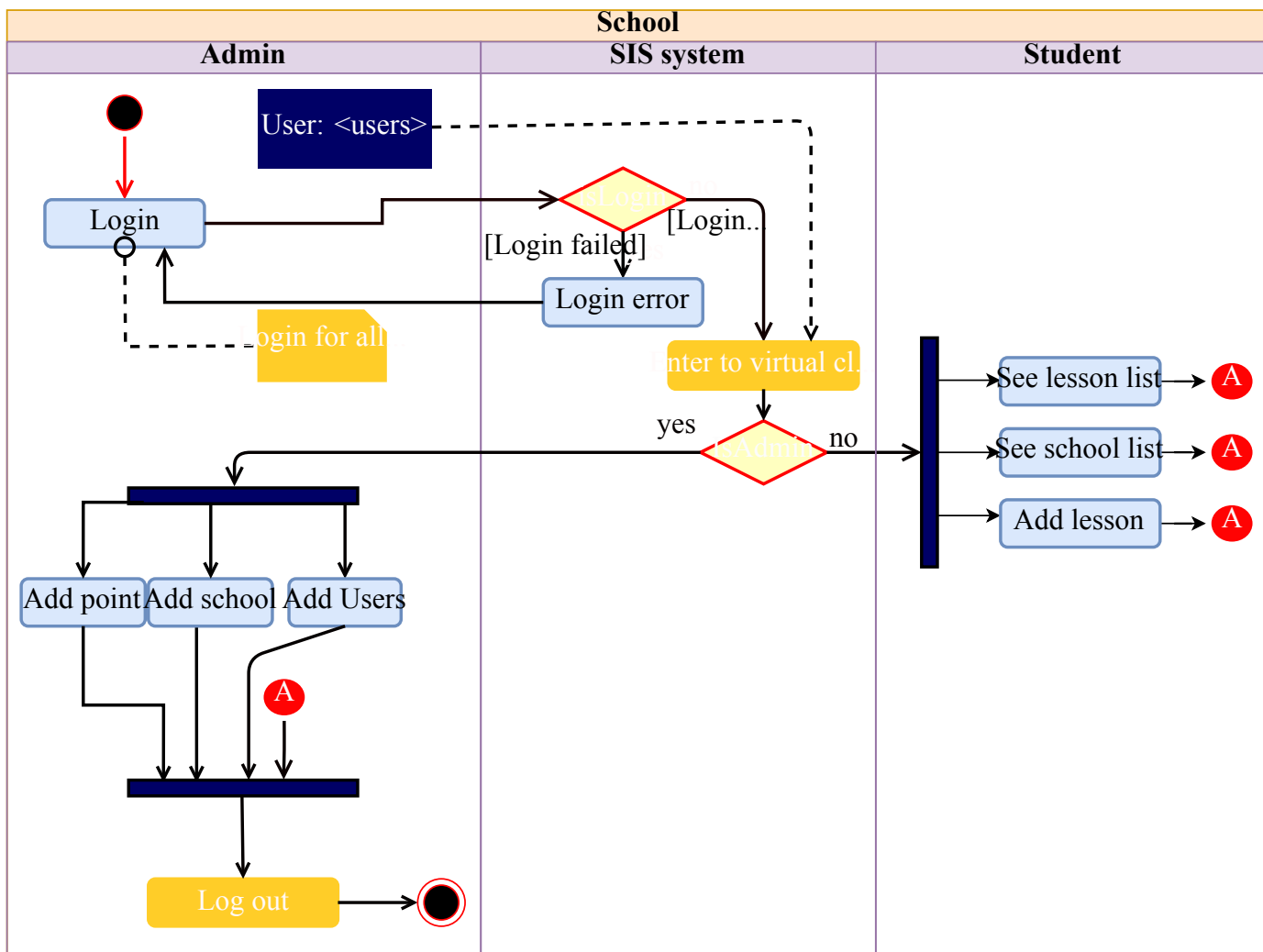
4.4.7 Point class

Энэхүү класс нь оюутны судалсан хичээлийн дүнгийн мэдээллийг агуулах ба шинж бүрд өөрийн get болон set метотууд байна.

Table 4.12: Point class

| Нэр | Шинж | Метод | Төрөл | Тайлбар |
|-------------|------|-------|-------|--|
| id | ✓ | | int | Дүнгийн давтагдашгүй дугаар |
| lesson_id | ✓ | | int | Дүнгийн мэдээлэлд харгалзах хичээлийн дугаар |
| point | ✓ | | float | Дүнгийн мэдээлэл агуулагдана. |
| user_id | ✓ | | int | Дүнгийн мэдээлэлд харгалзах оюутны дугаар |
| addPoint | | ✓ | | Системд дүнгийн мэдээлэл бүртгэх |
| editPoint | | ✓ | | Дүнгийн мэдээлэл засварлана |
| deletePoint | | ✓ | | Дүнгийн мэдээлэл устгах |

4.5 Үйл ажиллагааны урсгалын диаграмм



Зураг 4.3: Үйл ажиллагааны урсгалын диаграмм

Table 4.13: Класс диаграммын холбоосуудын тайлбар

| | |
|------------|---|
| User | User класс нь систем хэрэглэгчийн мэдээллийг агуулсан бие даасан класс юм. |
| Role | Role класс нь системийн ажил үүргийн мэдээллийг агуулсан бие даасан класс юм. Хэрэглэгч бүр системд ямар үүрэгтэй оролцохыг тодорхойлох учраас системийн үүрэггүй ажилтан байх боломжгүй. |
| Lesson | Lesson класс нь оюутны судалсан болон судалж байгаа хичээлийн мэдээллийг агуулсан бие даасан класс юм. |
| Permission | Permission класс нь систем хэрэглэгчийн үүргийн мэдээллийг бусад үйлдлүүдтэй холбож зөвшөөрөл олгох үүрэгтэй бие даасан класс юм. |
| School | School класс нь системд бүртгэлтэй сургуулийн мэдээллийг агуулсан бие даасан класс юм. |
| Point | Point класс нь системд бүртгэлтэй хэрэглэгчийн судалсан хичээлүүдийн дүнгийн мэдээллийг агуулсан бие даасан класс юм. |
| Student | Student класс нь системд бүртгэлтэй оюутны мэдээллийг агуулсан бие даасан класс юм. |

5. ХЭРЭГЖҮҮЛЭЛТ

GrapeCity Монгол ХХК дээр дадлага хийх хугацаандаа оюутан бүртгэлийн модуль болон банк дундын сүлжээний модулийн front-end талын хөгжүүлэлт дээр ажиллахын тулд TypeScript болон Angular CLI-г анхан шатнаас эхлэн суралцсан. Мөн компанийн дүрэм журам, удирдлага менежмент, үйл ажиллагаатай танилцаж, хөгжүүлэлтийн үйл явцад бодитоор оролцож эхлээд Angular фраймворкыг гүнзгийрүүлэн судалж, ашиглахын зэрэгцээ бусад технологи болон удирдлага менежменттэй холбоотой олон шинэ зүйлсийг суралцсан билээ.

5.1 Системийг бүтээхэд ашигласан технологи

5.1.1 *Angular.js*

Angular бол JavaScript фреймворк бөгөөд вэб аппликейшн бүтээхэд зориулагдсан.

Юуг мэдэж байх шаардлагатай вэ?

- HTML & CSS
- Basic JavaScript
- Programming fundamentals (Functions, conditionals, loop, etc..)

Мөн TypeScript, ECMAScript 2015 (ES6) програмчилалын хэлүүд, JavaScript code-г сервер тал дээр ажиллуулагч орчин болох Node.js технологи үүний сан зохицуулагч болох NPM багаж ашиглалтын талаарх мэдлэгүүд хэрэгтэй. Angular CLI нь project шинээр үүсгэх, тест хийх, файл нэмэх, хөгжүүлэгч орчинд deployment хийх гэх мэт боломжуудыг агуулж эдгээрийг terminal/console орчноос командаар ажиллуулагч багаж юм.

5.1.2 *Node.js*

Node.js гэдэг нь JavaScript дээр бичигдсэн кодыг ажиллуулж чадах орчин. Node.js-ын тусламжтайгаар Angular.js дээрх бичсэн кодоо build хийж терминалын орчинд шалгалт хийн

ажиллауулж үзсэн.

5.1.3 JBOSS server

JBoss EAP нь opensource GNU LGPL /GNU lesser general public/ лиценз дээр бичигдсэн учраас эх кодыг нь ашиглах боломжтой. Гол нь гарч ирсэн бүтээгдэхүүнээ Wildfly, Jboss EAP гэж нэрлэх мөн Red hat, JBoss EAP, Wildfly-ын логог ашиглах нэрийг нь дурдахыг хориглодог учраас admin console, Welcome screen, Error screen зэрэг дээр байгаа нэрнүүд болон логог арилгаж өөрсдийн нэрээр солино. Би middleware-ийн нэрээ “Nes middleware” гэж өгсөн. Админ консоль нь JBoss-ын үндсэн кодноосоо тусдаа Hal-console нэртэй тусдаа project-ыг build хийгээд гарсан jar-аа тохирох зам дээр хуулаад module.xml дээр тухайн файлын нэрийг тавина.

5.2 Use-case бүрийн хэрэглэгчийн интерфэйс

Үндсэн

Систем

Ажил үүрэг

Нууц үг

Бүтцийн тохиргоо

Уүсгэх

Х Болих

Зураг

Зураг оруулах

Овог *

Отгонбаатар

Овог 2

Otgonbaatar

Нэр *

Ихбаяр

Нэр 2

Ikhbayar

Регистрийн төрөл *

ГАДААД ИРГЭНИЙ РЕГИСТЕР

Регистр

И-Мэйл *

sdfsdf@saf.com

Гар утас

Нэвтрэх нэр *

ikhee

Нууц үг *

...

Нууц үг(давтах) *

...

Зэрэглэл *

4

Төрөл *

ЭНГИЙН

Төлөв

ШИНЭ

Зураг 5.1: Хэрэглэгчийг системд шинээр бүртгэх хэсэг

Дүгнэлт

Миний бие үйлдвэрлэлийн дадлагаа "GrapeCity Mongolia" ХХК-д гурван долоо хоногийн хугацаанд хийж гүйцэтгэн дуусгалаа. Энэ хугацаанд албан байгууллага дээр ажиллах хэв маяг сургуулиас өөр болохыг ойлгосон бөгөөд ямар нэгэн асуудал гарсан тохиолдолд түүнийг багийн хүчээр хэрхэн даван туулж болох зэрэг зүйлсийг суралцсан. Түүнчлэн багаар ажиллах,

| Админ ▾ [ADM-B-010] ХЭРЭГЛЭГЧИЙН ЖАГСААЛТ ? | | | | | | | |
|--|-----------|-----------|---------------------|-------------|----------|--------|-----|
| <div> <div>+ Үүсгэх</div> <div>XLSX</div> <div></div> <div></div> </div> | | | | | | | |
| | Нэр | Овог | Хэрэглэгчийн дугаар | Нэвтрэх нэр | Зэрэглэл | Төрөл | Төл |
| | 46 FNAME | 46 LNAME | 46 | 008 | 9 | ЭНГИЙН | ИД |
| | 59 FNAME | 59 LNAME | 59 | 009 | 9 | ЭНГИЙН | ИД |
| | 62 FNAME | 62 LNAME | 62 | 010 | 9 | ЭНГИЙН | ИД |
| | 49 FNAME | 49 LNAME | 49 | 012 | 9 | ЭНГИЙН | ИД |
| | 57 FNAME | 57 LNAME | 57 | 013 | 9 | ЭНГИЙН | ИД |
| | 61 FNAME | 61 LNAME | 61 | 014 | 9 | ЭНГИЙН | ИД |
| | 65 FNAME | 65 LNAME | 65 | 015 | 9 | ЭНГИЙН | ИД |
| | 66 FNAME | 66 LNAME | 66 | 016 | 9 | ЭНГИЙН | ИД |
| | 67 FNAME | 67 LNAME | 67 | 017 | 9 | ЭНГИЙН | ИД |
| | 68 FNAME | 68 LNAME | 68 | 018 | 9 | ЭНГИЙН | ИД |
| | 50 FNAME | 50 LNAME | 50 | 019 | 9 | ЭНГИЙН | ИД |
| | 54 FNAME | 54 LNAME | 54 | 020 | 9 | ЭНГИЙН | ИД |
| | 128 FNAME | 128 LNAME | 128 | 0202 | 1 | ЭНГИЙН | ИД |
| | 56 FNAME | 56 LNAME | 56 | 021 | 9 | ЭНГИЙН | ИД |
| | 58 FNAME | 58 LNAME | 58 | 022 | 9 | ЭНГИЙН | ИД |
| | 60 FNAME | 60 LNAME | 60 | 023 | 9 | ЭНГИЙН | ИД |
| | 63 FNAME | 63 LNAME | 63 | 024 | 9 | ЭНГИЙН | ИД |
| | 64 FNAME | 64 LNAME | 64 | 025 | 9 | ЭНГИЙН | ИД |
| | 69 FNAME | 69 LNAME | 69 | 026 | 9 | ЭНГИЙН | ИД |
| | 70 FNAME | 70 LNAME | 70 | 027 | 9 | ЭНГИЙН | ИД |
| | 88 FNAME | 88 LNAME | 88 | 029 | 9 | ЭНГИЙН | ИД |
| | 89 FNAME | 89 LNAME | 89 | 030 | 1 | ЭНГИЙН | ИД |
| | 92 FNAME | 92 LNAME | 92 | 032 | 9 | ЭНГИЙН | ИД |
| | 93 FNAME | 93 LNAME | 93 | 033 | 9 | ЭНГИЙН | ИД |
| | 95 FNAME | 95 LNAME | 95 | 035 | 9 | ЭНГИЙН | ИД |

Зураг 5.2: Системд бүртгэлтэй хэрэглэгчид

төслийг үр дүнтэй байлгах зэрэг нь судалгаа, шинжилгээ баримт бичгүүд дээр тулгуурлан явагддаг гэдгийг ухаарсан.

Эзэмшисэн мэдлэг, дадлага туршлага нь дан ганц код бичих чадвар гэхээс илүүтэйгээр хамтын ажиллагаа буюу бусадтай хэрхэн зөв харилцаж өөрийн санааг ойлгуулах, бусдаас хэрхэн суралцах зэрэг зүйлст илүү их төвлөрч ажилласан. Түүнчлэн банк санхүүгийн талаар тодорхой хэмжээний мэдлэгтэй болсон төдийгүй банк хоорондын гүйлгээ хэрхэн яаж хийгддэг

The image shows a mobile application interface with a red header containing the text 'nes'. Below the header, there are three text input fields with placeholder text: 'Хуучин нууц үг', 'Шинэ нууц үг', and 'Шинэ нууц үг (давтах)'. Below these fields is a blue button labeled 'Солих'. At the bottom of the screen, there is a blue link that says 'Өөр хэрэглэгчээр нэвтрэх'.

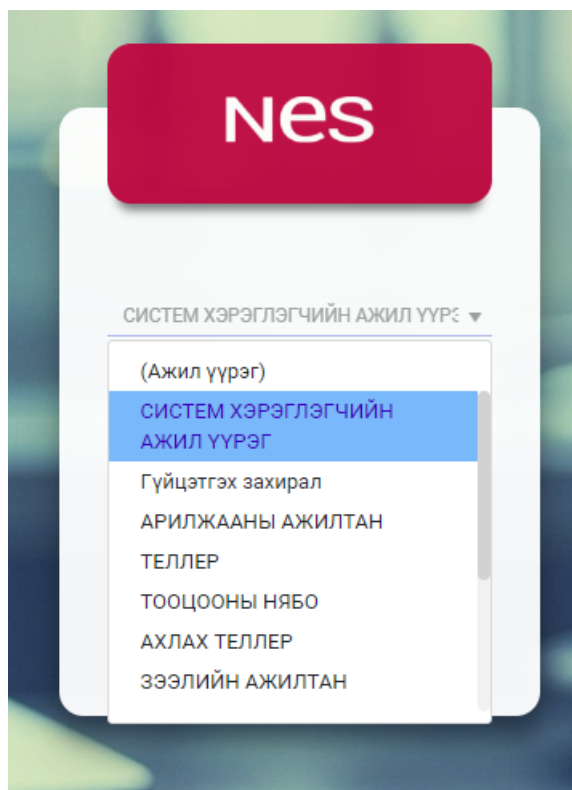
Зураг 5.3: Хэрэглэгчийн нууц үг солих хэсэг

гэх зэрэг зүйлсийг шинээр мэдэж авсан.

Бэлэн болчихсон програм сайжруулалтын үйл явц хэрхэн яаж явагддаг гэх зэрэг зүйлүүдийг мэдэж авлаа. Цаашлаад их сургуульд зохих ёсны мэдлэгийг авч түүнийгээ тодорхой хэмжээнд ашиглаж болохыг харж авсан. Дадлагын хугацаанд өмнөх семестер жава програмчлалын хэлийг судалж мэдсэн мэдлэг дээрээ тулгаарлан цааш хөгжиж болохуйц болсоны үндсэн дээр дадлагаа амжилттай хийж дуусгалаа.

| [ADM-B-130] ЭРХИЙН ХАЙЛТ | | | |
|---|----------------------|----------|-------------------|
| <input type="text"/> ☰ ⌵ ⋮ | | | |
| <input type="checkbox"/> | Систем | Дугаар | Нэр |
| <input type="checkbox"/> | 1000 - АДМИН | 10003010 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1000 - АДМИН | 10003011 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1000 - АДМИН | 10003012 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1000 - АДМИН | 10003013 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1000 - АДМИН | 10003014 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1000 - АДМИН | 10003020 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1000 - АДМИН | 10003021 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1000 - АДМИН | 10003023 | ХЭРЭГЛЭГЧИЙН Б |
| <input type="checkbox"/> | 1033 - НӨАТ-Н БАРИМТ | 10330040 | Баркодын жагсаал |
| <input type="checkbox"/> | 1033 - НӨАТ-Н БАРИМТ | 10330042 | Баркодын бүртгэл |
| <input checked="" type="checkbox"/> | 9999 - DEMO | 99990020 | Хүмүүсийн төгссөн |
| <input checked="" type="checkbox"/> | 9999 - DEMO | 99990021 | Хүмүүсийн төгссөн |
| <input checked="" type="checkbox"/> | 9999 - DEMO | 99990022 | Хүмүүсийн төгссөн |
| <input checked="" type="checkbox"/> | 9999 - DEMO | 99990023 | Хүмүүсийн төгссөн |
| <input checked="" type="checkbox"/> | 9999 - DEMO | 99990024 | Хүмүүсийн төгссөн |

Зураг 5.4: Хэрэглэгч ямар үйлдэл хийж болохыг тодорхойлно



Зураг 5.5: Хэрэглэгчид эрх өгнө

Хэрэглэгчийн төгссөн сургуулийн мэдээлэл

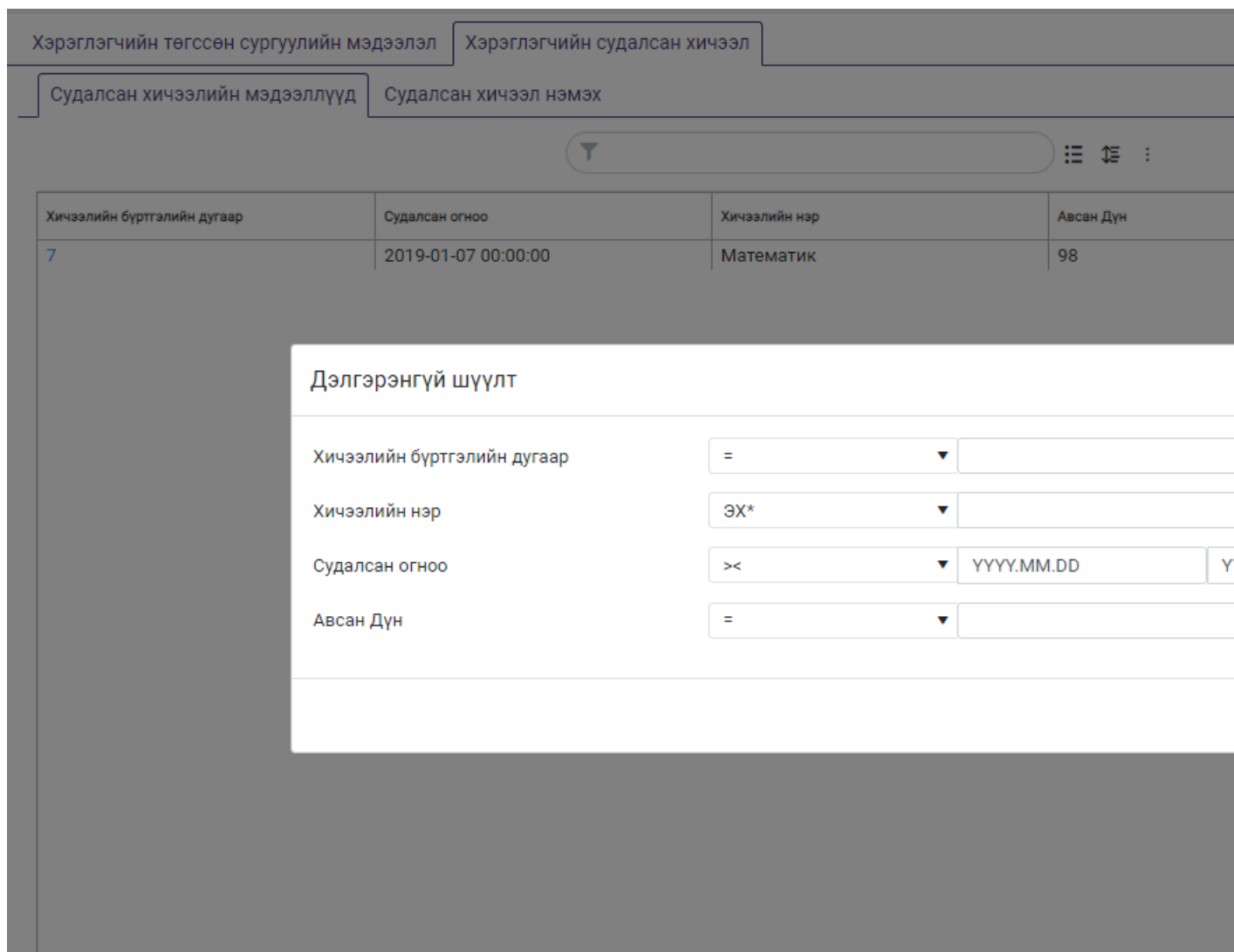
Хэрэглэгчийн судалсан хичээл

Хадгалах

Болих

| | |
|--------------------------------------|---------------|
| Сургуулийн нэр | Төгссөн огноо |
| Монгол Улсын Боловсролын Их Сургууль | 2020.09.11 |
| Голч оноо | Элссэн огноо |
| 3 | 2016.01.01 |
| Оюутны дугаар | |
| 9584565 | |

Зураг 5.6: Сургуулийн мэдээлэл оруулах



Зураг 5.8: Хайлт хийх

Bibliography

- [1] Angular.js Introduction <https://angular.io/guide/setup-local>
- [2] Node.js <https://nodejs.org/en/docs/>
- [3] GrapeCity Mongolia LLC <https://www.grapecity.mn/>
- [4] MOST MONEY <https://www.mostmoney.mn/>
- [5] Gerege <https://www.gerege.mn/>

A. КОДЫН ХЭРЭГЖҮҮЛЭЛТ

Энд CLIENT талтай харилцах функцүүдийг тодорхойлж өгнө. Өөрөөр хэрэглэгч доор байгаа функцуудтай харьцаж өгөгдөл авч, засварлаж, оруулах боломжтой.

```
1 package mn.grapecity.nes.demo.school;
2
3 import java.util.List;
4 import javax.ejb.Local;
5 import mn.grapecity.nes.NesException;
6 import mn.grapecity.nes.demo.school.entity.ClnDemoLesson;
7 import mn.grapecity.nes.demo.school.entity.ClnDemoSchool;
8 import mn.grapecity.nes.demo.school.entity.ClnVwDemoSchool;
9 import mn.grapecity.nes.entity.cln.ClnFilterfield;
10
11 @Local
12 public interface SchoolParamRemote {
13     List<ClnVwDemoSchool> selectDemoSchool(ClnFilterfield[] var1, int
14         var2, int var3) throws NesException;
15
16     Long insertDemoSchool(ClnDemoSchool var1) throws NesException;
17
18     void updateDemoSchool(ClnDemoSchool var1) throws NesException;
19
20     void deleteDemoSchool(Long var1) throws NesException;
21
22     ClnDemoSchool detailDemoSchool(Long var1) throws NesException;
23
24     List<ClnDemoSchool> selectDemoSchool1(ClnFilterfield[] var1, int
25         var2, int var3) throws NesException;
26
27     List<ClnDemoLesson> selectDemoLesson(ClnFilterfield[] var1, int
28         var2, int var3) throws NesException;
29
30     Long insertDemoLesson(ClnDemoLesson var1) throws NesException;
31
32     void updateDemoLesson(ClnDemoLesson var1) throws NesException;
33
34     void deleteDemoLesson(Long var1) throws NesException;
35
36     ClnDemoLesson detailDemoLesson(Long var1) throws NesException;
37 }
```

Код A.1: DEMO модулийн Interface

selectDemoSchool функц нь бүртгэлтэй сургуулийн жагсаалтыг буцаана. cg нь core модулын тодорхойлсон модуль бөгөөд хийсвэр өгөгдлийн сан болон төрөл бүрийн өгөгдөлтэй харьцах функцүүдийг өөр дээрээ агуулсан байдаг. Хэрвээ шүүлт хийгдэхээр байвал шүүлт хийхэд зөвшөөрөлтэй эсэхийг cg.addPossField функцаар шалгана. Filter хийхэд зөвшөөрөгдөх талбаруудыг possFields хувьсагчид хадгалсан. cg.getData() баазаас filter-т тохирох мэдээллийг авчирна. cloneVw2DemoSchool функц нь баазаас ирсэн өгөгдлийг CLIENT-ын тодорхойлсон класст хувиргаж буцаана.

```

1  /* Clntfilterfield -> field, value in filter
2  * count, startPosition -> size*/
3  public List<ClntVwDemoSchool> selectDemoSchool(Clntfilterfield[]
4      filter, int startPosition, int count) throws NesException {
5      this.cg = new Condgen();
6      String[] possFields = new String[]{"DEMOID", "USERID", "
7          SCHOOLNAME", "STARTDATE", "ENDDATE", "GRADEPOINT", "
8          SCHOOL_NAME", "SCHOOL_NAME2", "ORDER_NO"};
9      this.cg.addPossField(possFields);
10     List<VwDemoSchool> e = null;
11     Clntfilterfield ff = new Clntfilterfield();
12     ArrayList<Clntfilterfield> lstFilter = new ArrayList();
13     lstFilter.addAll(Arrays.asList(filter));
14     lstFilter.add(ff);
15     String[] orderColumns = new String[]{"SCHOOLNAME ASC"};
16     e = this.cg.getData(lstFilter, VwDemoSchool.class,
17         startPosition, count, this.em, orderColumns);
18     return this.cloneVw2DemoSchool(e);
19 }

```

Код А.2: selectDemoSchool функцийн хэрэгжүүлэлт

com.getSeq функцийн тусламжтайгаар дахин давтагдашгүй ID үүсгэнэ. em.persist функ нь өгөгдлийг баазад хадгалж өгнө.

```

1  public Long insertDemoSchool(ClntDemoSchool p) throws NesException
2  {
3      Long demoId = this.com.getSeq("DEMO_SCHOOL");
4      DemoSchool c = new DemoSchool();
5      c.setDemoid(demoId);
6      c.setUserid(p.getUserid());
7      c.setSchoolname(p.getSchoolname());
8      c.setStartdate(p.getStartdate());
9      c.setEnddate(p.getEnddate());
10     c.setGradePoint(p.getGradePoint());
11     this.em.persist(c);
12     return demoId;
13 }

```

Код А.3: Сургуулийн мэдээллийг хадгалах хэсэг

Үйлдэл болгон нь харгалзах эрхүүдтэй байдаг бөгөөд хэрэглэгч бүр тэрхүү эрхийг өөртөө авсан байх шаардлагатай байдаг.

```

1  MERGE INTO adm_operation A
2  using dual
3  ON ( A.oper_code = '99990020' )
4  WHEN NOT matched THEN
5      INSERT (oper_code,
6          NAME,
7          name2,
8          sys_no,
9          func_name,

```

```

10         lookup,
11         audit_level,
12         log_req_data,
13         log_res_data)
14     VALUES('99990020',
15            Upper(
16 '          '),
17            '',
18            9999,
19            'selectDemoPerson',
20 'java:global/nas/demo.person.EJB/DemoParam!mn.grapecity.nas.demo.
    person.PersonParamRemote'
21        ,
22        1,
23        1,
24        0)
25     WHEN matched THEN
26         UPDATE SET lookup =
27         'java:global/nas/demo.person.EJB/DemoParam!mn.grapecity.nas.demo.
    person.PersonParamRemote'
28
29     NAME = Upper(
30 '          '),
31     name2 = '',
32     func_name = 'selectDemoPerson';
33     MERGE INTO adm_oper_priv a
34     using dual
35     ON ( a.oper_code = '99990040'
36         AND a.priv_code = '9999040' )
37     WHEN NOT matched THEN
38     INSERT(oper_code, priv_code)
39     VALUES('99990040', '99990040');

```

Код А.4: Үйлдлийн эрх өгөх жишээ sql

Вэб нь ngOnInit(), ngAfterViewInit(), onRefresh(), ngOnDestroy() үндсэн функцуудыг өөртөө агуулдаг бөгөөд Service, model, html, typescript гэсэн үндсэн бүтцээс тогтоно. model дээрээ сервер талтай ямар талбаруудаар холбогдохоо тодорхойлж өгдөг бол Service нь сервертэй ямар замаар хэрхэн холбогдохыг шийдэж өгдөг. NES platform-ын хувьд нэг суурь модуль дотор бусад модуль нь агуулагддаг бөгөөд суурь модулийн функцуудийг дагалдах модулиуд нь ашиглах боломжтой байдаг бөгөөд кодын бичиглэлийг хөнгөвчилж өгсөн.

```

1     export class DemoSchool021Screen extends FormScreenBase
2         implements OnInit {
3         onRefresh({ companyCode, param }: ScreenRefreshContext<
4             DemoSchool021Screen>) {}
5         ngAfterViewInit() {
6             // Hereglegchiin delgerengui duudagdsanii daraa tolow
7             shalgalt hiih
8             this.obs = zip(
9                 this.service.syncResponse$
10            ).subscribe((res) => {
11                const detail = res[0];

```

```

9         if (detail.action === 'detail') {
10             this.service.setUserId(detail.response.userid)
11             this.userId = detail.response.demoid
12             if (detail.response.userStatus === 0) {
13                 // Hereglegchiin tolow shine baiwal readonly bolno.
14                 this.isStatusNew = true;
15             } else {
16
17             }
18         }
19     });
20
21     this.service.syncResponse$.subscribe((res) => {
22         if (!res.error && (res.action === 'create' || res.action
23             === 'update')) {
24
25         }
26     });
27 }

```

Код А.5: Дэлгэцийн кодчилал

```

1     import { ID, Field, Title, Model, FieldType } from '@nes/data';
2     import { Validators } from '@angular/forms';
3
4     @Model('DEMO-SCHOOL.DemoLesson')
5     export class DemoLessonModel {
6         /* Mandatory talbariig ID gej todorhoilj ogdog.
7         * Validators shalgalt hiij ogch boloh bogood email, болон regex
8         * shalgalt hiij болонjtoi
9         */
10        @ID()
11        @Field({
12            filterable: true,
13            sortable: true,
14            validators: [Validators.max(99999999)]
15        })
16        lessonid: number; // Zaawal talbariin torol todorhoilj ogno
17
18        @Field({
19            validators: [Validators.max(99999999)]
20        })
21        userid: number;
22
23        @Field({
24            filterable: true,
25            sortable: true,
26            validators: [Validators.maxLength(50)]
27        })
28        lessonname: string;

```

```

29  @Field({
30      filterable: true,
31      sortable: true,
32      validators: [Validators.maxLength(50)]
33  })
34  seendate: Date;
35
36  @Field({
37      filterable: true,
38      sortable: true,
39      validators: [Validators.max(100)]
40  })
41  point: number;
42  }

```

Код А.6: Дэлгэцийн кодчилол(module)

```

1  import { Injectable, Injector } from '@angular/core';
2  import { RemoteDataService } from '@nes/data';
3  import { DemoSchoolModel } from '../models/demo-school-020.model';
4  import { Observable } from 'rxjs';
5
6  @Injectable()
7  export class DemoSchool020DataService extends RemoteDataService<
8      DemoSchoolModel> {
9      constructor(injector: Injector) {
10         super(injector);
11     }
12     userId:String
13     getUserId(){
14         return this.userId
15     }
16     setUserId(id){
17         this.userId = id
18     }
19     options = {
20         modelType: DemoSchoolModel,
21         isOld: true,
22         syncOptions: {
23             read: 99990020,
24             create: 99990021,
25             update: 99990022,
26             delete: 99990023,
27             detail: 99990024,
28         },
29     };
30 }

```

Код А.7: Дэлгэцийн кодчилол(service)