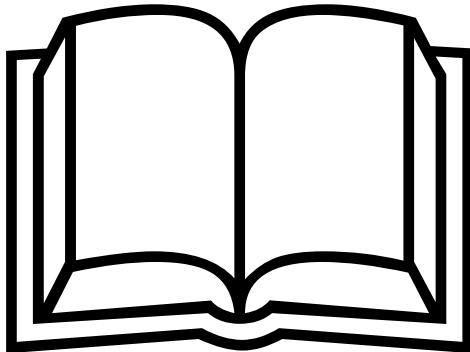




Монгол Улсын Их Сургууль

Мэдээлэл, компьютерийн ухааны танхим



Програм хангамжийн шаардлагын ШИНЖИЛГЭЭ

Лекц № 1 - 7 : Лекцын тэмдэглэлийн эмхэтгэл

Гүйцэтгэсэн: Э.Багабанди /19B1NUM0700/

Шалгасан: Б.Батням /Ахлах багш/

Улаанбаатар хот

2021 он

АГУУЛГА

❖ **Лекц №1 – Шаардлагын инженерчлэл**

1. Шаардлага гэж юу вэ?
 - Тодорхойлолт
 - Шаардлагыг олж тогтоох үед гарах асуудал
 - Шаардлага нь юуг тайлбарлан бичдэг вэ?
 - Шаардлагын төрөл

2. Дүгнэлт
3. Эх сурвалж

❖ **Лекц №2 - Шаардлагын инженерчлэлийн(РЕ) аргачлал /явагдах байдал/**

1. Шаардлагын инженерчлэлийн зорилго
2. Хөгжүүлэлт процесст хэлбэрээр нь оруулах
3. Эцсийн бүтээгдэхүүн: Систем ЮУ чадах ёстой тухай гэрээ
4. Хэрэглэгч хөгжүүлэгч хоорондын уялдаа
5. Шаардлагын инженерчлэл буюу шаардлагын шинжилгээ
 - Системтэй арга барил
 - Шаардлагын инженерчлэлийн үед хийгдэх ажил
 - Шаардлагын инженерчлэлийн сүл тал
 - Шаардлагын инженерчлэлийн эрсдэл
 - Эрсдэлийг шийдэх
6. Эх сурвалж

❖ **Лекц №3 – Шаардлагын тодорхойлолт**

1. Шаардлагын тодорхойлолтын төрөл
 - Хэрэглэгчийн шаардлага
 - Системийн шаардлага
2. Шаардлагыг ялгаатай өнцгөөс харах
3. Шаардлагын тодорхойлолт боловсруулах
4. Шаардлагыг бүтэцлэх
5. Шаардлагын тодорхойлолт дах ухагдахуун ойлголт
6. Сценарь – Шаардлага тодорхойлох туслах хэрэгсэл
7. Ажлын явц /- ын диаграмм
8. Лекцийн асуулт
9. Эх сурвалж

❖ **Лекц №4 – Шаардлагыг мөшгөх (Requirements – tracing)**

1. Шаардлагын мөшгөлт
2. Үе шат хоорондын мөшгөхүйц байдал
3. Мөшгөж болохуйц байдлын жишээ
4. Шаардлагын өөрчлөлт
 - Шаардлагын өөрчлөлтийн менежмент
 - CASE – Багажаар дэмжих
5. Лекцийн бататгах асуулт
6. Хураангуй
7. Эх сурвалж

❖ **Лекц №5 – Хэрэглэгчийн шаардлагыг тодорхойлох**

1. Шаардлага гэж юу вэ?
 - Хэрэглэгчийн шаардлагын тодорхойлолт
 - Төслийн хувьд тавигдаж байгаа шаардлагууд:
 - Функциональ шаардлага
 - Функциональ бус шаардлага
2. Төслийн шаардлагуудыг тогтоох
 - Шаарлагуудыг таних процесс
 - Төслийг хэрэгжүүлэх эсэхийг тодорхойлох
 - Хэрэглэгчээс тавигддаг онцгой нөхцлүүд
3. Ажлын баримт бичиг (АББ) буюу Statement of Work
 - Дизайн буюу ажлын нарийвчилсан тодорхойлолт төрлийн ажлын баримт бичиг
 - Хөдөлмөр зарцуулалтын түвшин АББ
 - Гүйцэтгэлд сууринсан АББ
4. Төслийн шаардлагуудыг бүрэн тодорхойлох
5. Шаардлагын матриксыг боловсруулах
6. Дүгнэлт
7. Эх сурвалж

❖ **Лекц №6 – Use case /ажлын явц/**

1. Ухагдахуун – Ажлын явц/use case
2. Ажлын явцын зорилго
3. Мэдээллийн систем дэх ажлын явц
4. Шинжэлгээний үеийн асуудал
5. Тоглогч (actor) гэж хэн бэ?

- Тоглогчийн жишээ
- 6. Ажлын явцыг тайлбарлах
- 7. Ажлын явцын задаргааны хэв (Use case template)
- 8. Ажлын явцын жишээ:
 - Торх дүүргэх
- 9. Ажлын явцын диаграм (Use case diagram)
 - Тодорхойлолт
 - Жишээ
 - Ажлын явцын диаграмын харьцаа
- 10. Бататгах асуулт
- 11. Дүгнэлт
- 12. Эх сурвалж

❖ **Лекц №7 – /Message – зурвас & Scenario – Сценаръ/**

- 1. Ухагдахуун – Message /зурвас
- 2. Зурвасын жишээ
 - №1, №2
- 3. Бататгах асуулт /4.2/
- 4. Дүгнэлт
- 5. Ухагдахуун – Scenario /сценаръ
- 6. Ажлын явц болон сценаръ
- 7. Сценарийн ЗНХ – ийн тэмдэглэл
 - Дарааллын диаграм
 - Харилцааны диаграм
- 8. Класс диаграмын жишээ
- 9. Дарааллын диаграмын жишээ
- 10. Дарааллын диаграм болон классын диаграм хоорондын нийцэмжийн (уялдах байдал, нэгдмэл чанар) дүрэм
- 11. Харилцааны болон объект диаграмын харьцуулалт
- 12. Харилцааны болон дарааллын диаграмын харьцуулалт
- 13. Бататгах асуулт /4.3/
- 14. Дүгнэлт
- 15. Эх сурвалж

Лекц №1 – Шаардлагын инженерчлэл

1. Шаардлага гэж юу вэ?

- Шаардлага – (ууд) (Requirements) нь бүтээгдэхүүний чанарын болон тоон чанарыг тогтоодог.
- Шаардлагын олж тогтоох үед гарах асуудал:

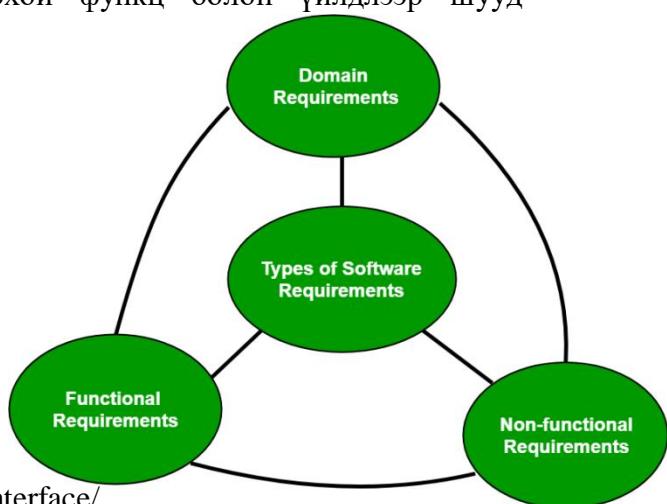
Шаардлага	Хүслийн жагсаалт
Олон утгатай	Клаус шиг унадаг дугуй
Биелүүлэх боломжгүй	Бурхантай уулзах
Хэтэрхий тодорхой	Бөөрөхий бөмбөг
Бүрэн бус	Модтой болох
Хэвийн бус	Дээр үеийн чийчаа, хурдан машин

- Шаардлага нь юуг тайлбарлан бичдэг вэ?
 - Шинж чанар эсвэл нөхцөл (ихэнхдээ хэрэглэгч тогтоож өгдөг) асуудлын шийдэхийн тулд, зорилгод хүрэхийн тулд эсвэл тухайн бүрэлдэхүүн хэсгийн биелүүлэх ёстой гэрээ, норм, нарийвчилан задалсан тодорхойлолт эсвэл өөр нэг формал тогтсон баримтын шаардлагыг хангахын тулд тухайн системд тавигдсан шинж чанар болон нөхцөлүүдийг тодорхойлон бичсэн бичиг баримтыг **шаардлагын бичиг баримт** гэнэ.
 - Ямар бүрэлдэхүүн “**Юу чадах хэрэгтэйг?**”, “**Ямар байдалтай байхыг?**” мөн “**Юугаар түүнийг хэрэгжүүлэх хэрэгтэйг?**” шаардлагын тайлбар бичиг тогтоодог. Мөн шаардлагын тайлбар бичиг нь хэрхэн хэрэгжүүлэхийг тодорхой тогтоодоггүй.
- “Сайн” шаардлагын шинж чанар:
 - Ойлгомжтой бүтээлгэдсэн
 - Бүлэг, дэд бүлэг Г.М – ээр хуваасан
 - Шаардлагыг бүтээлэсэн
 - Шаардлага нэг бүрийг таних, шалгахуйц байх
 - Мөн чанарыг олох, хэрэглэх
 - Ангилах
 - Бүрэн байдал
 - Шаардлагыг задлан шинжилж болохуйц байдал
 - Зөрчилгүй байдал

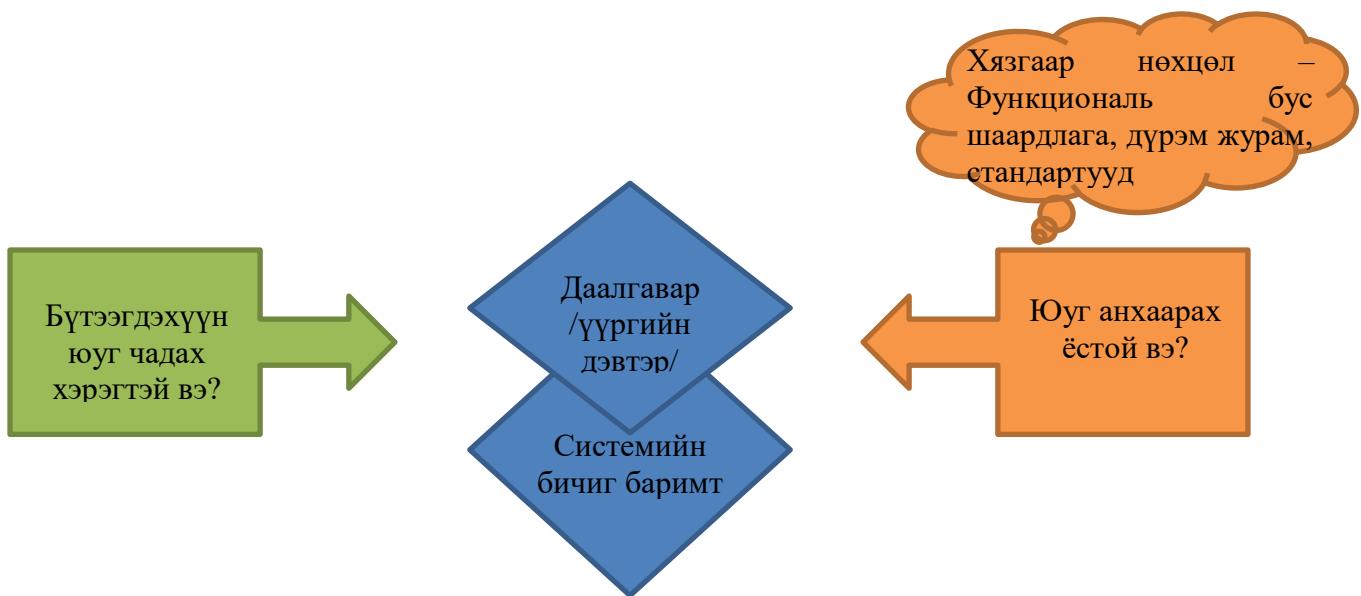
Тестийн
боломжууд

Үл
ойлголцол

- Шаардлагыг задлан шинжилж болохуйц байх бөгөөд шаардлагыг засварлан өөрчлөх болохуйц байх
- Ойлгомжтой байдал
 - Зураг үгнээс илүүг өгүүлнэ
 - Хэт их томьёоны орон байгалийн хэл ашиглах
- Шаардлагын төрөл
 - Шаардлагыг тайлбарлан бичихдээ хоёр ерөнхий байдлаа ялгадаг.
 - Функциональ шаардлага:
 - Хөгжүүлж буй програм хангамжийн систем юу хийх хэрэгтэйг тайлбарлан бичсэн шаардлага, тухайн програм хангамжийн системийн юу хийх хэрэгтэй үйл ажиллагааг шаардсан шаардлага бөгөөд тодорхой функц болон үйлдлээр шууд хэрэгжүүлж болдог.
 - Гүйцэтгэх үйл ажиллагаа
 - Алгоритм
 - Үйл явц
 - Өгөгдөл
 - Оролт
 - Гаралт
 - Өгөгдлийн загвар
 - Харилцах хэсэг /Graphic Interface/
 - Хэрэглэгчидтэй харилцах хэсэг
 - Бусад өөр систем рүү харьцах хэсэг
 - Функциональ бус шаардлага
 - Үйл ажиллагаанас бусад бүх шаардлагыг хэлэх бөгөөд хөгжүүлж буй програм хангамжийн систем нь хийж гүйцэтгэх үйл ажиллагаа дээр нэмээд тухайн систем бүтээгдэхүүн маань ямар шинж чанарыг үзүүлэх хэрэгтэйш авч үздэг, ихэвчлэн уг системийг хязгаарласан шаардлагууд байна.
 - Хүчин чадлын шаардлага
 - Тухайлбал, хэрэглэгчийн тоо, өгөгдлийн хэмжээ, хариу өгөх хугацаа, процесийн тохиох давтамж гэх мэт
 - Чанарын шаардлага



- Тухайлбал, найдвартай байдал, аюулгүй байдал, шилжүүлж суулгах боломжтой байдал гэх мэт
- Хэрэгжүүлэлтийн шаардлага
 - Хөгжүүлэлт, хөгжүүлэлтийн арга
 - Хөдөлмөр зарцуулалт
 - Чиг шугам (дүрэм, журам)
 - Хүлээлцэх
 - Засвар үйлчилгээ
 - Баримтжуулалт
- Функциональ шаардлага, хязгаар нөхцөл:



2. Дүгнэлт:

- Шаардлагын инженерчлэл гэдэг нь шаардлагыг олж илрүүлэх тогтоох бэлдэж өгөх зорилготой шаардлагын бичиг баримт бэлтгэх зорилготой үйл явц юм. Түүнээс гадна олж тогтоосон шаардлагыг ирээдүйд тухайн системийг баримтжуулахад ашигладаг байх шаардлагатай юм. Үүнээс харахад шаардлагын инженерчлэл, шаардлагын бичиг баримт нь уг систем бүтээгдхүүний суурь болж өгдөг бөгөөд сайн шаардлагын бичиг баримт бэлтгэсэн төдий чинээ сайн систем болох боломжтой юм.

3. Ашигласан эх сурвалж:

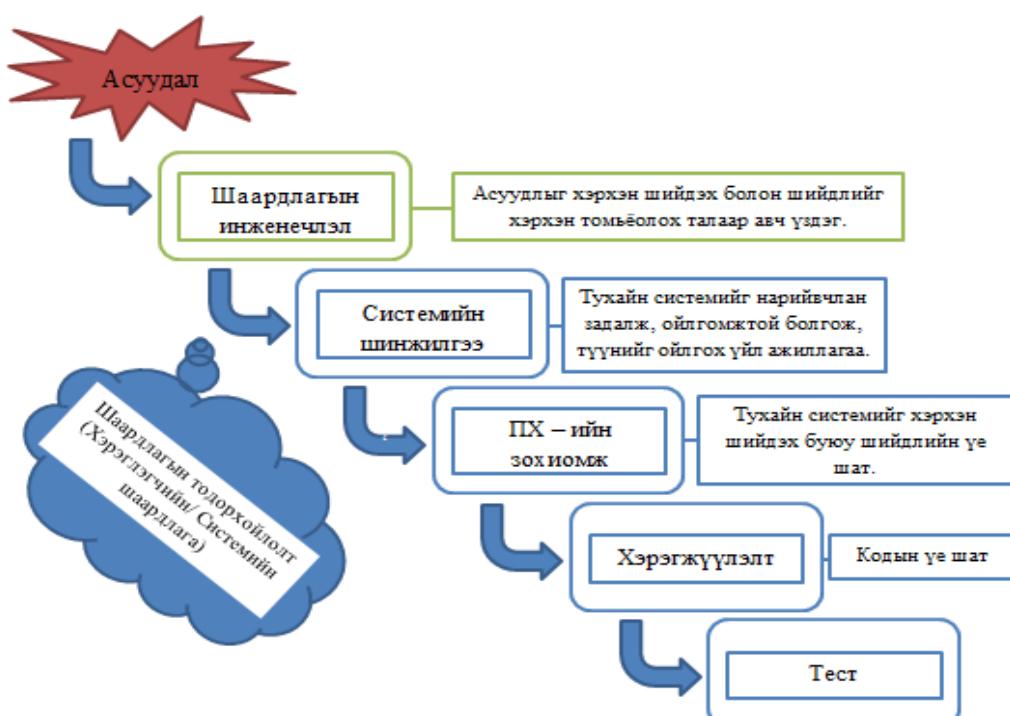
- <https://www.youtube.com/watch?v=gYDhlEuyi-w>
- <https://www.geeksforgeeks.org/software-engineering-classification-of-software-requirements/>
- <https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/>

Лекц №2 - Шаардлагын инженерчлэлийн(РЕ) аргачлал /явагдах байдал/

1. Шаардлагын инженерчлэлийн зорилго

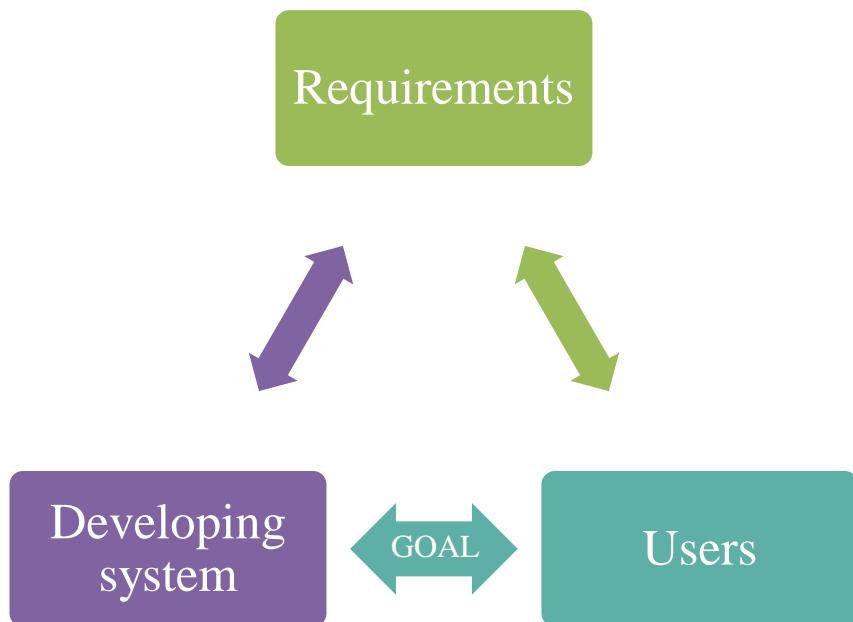
- Шаардлагыг олж тогтооход нь захиалагчид дэмжилэг үзүүлэх
- Шаардлагыг бэлдэхэд нь хөгжүүлэгчид дэмжлэг үзүүлэх
 - Хөгжүүлэгчийн тал руу тавиглаж байгаа шаардлагыг ихэндээ системийн шаардлага гэж үздэг.
 - Томьёолох /formulate/- Шаардлагуудыг тодорхой хэв загварт оруулж, ойлгомжтой болгох
 - Ангилах /classify/- Ойлгомжтой бөгөөд эмх цэгтэй болгохын тулд тодорхой ерөнхий шинж чанаруудаар болох хэрэгжүүлэх арга замаар нь ангилах. Тухайлбал функциональ болон функциональ бус шаардлага гэх мэтчилэн ангилах.
 - Эрэмбэтэй шаталсан бүтцэд оруулах /sorting/ - Шаардлага бүрт эрэмбэ тогтоож, дэс дараалалд оруулах бөгөөд ерөнхий шаардлага болон нарийвчилсан буюу дэд шаардлагуудыг тодорхойлох. Тухайлбал заавал байх ёстой шаардлага, байж болох шаардлага, зарим нэг шаардлагатай нөхцөл хэрэглэдэг гэх мэтээр эрэмбэлэх.
 - Тодруулах (Нарийвчилан задлах) /emphasize/ - Шаардлага бүрийг тодорхой болгох буюу нарийвлчлан задлан шинжлэх
- Системийг нотолгоожуулахад нь чанарын баталгаажуулалтад дэмжлэг үзүүлэх
 - Системийг хэрэгжүүлсний дараа тухайн системийг хүлээж авахын тулд уг системд тавигдсан шаардлагыг хангасан эсэхийг шалгах үйл явцыг чанарын баталгаажуулалт /quality assurance/ гэдэг.

2. Хөгжүүлэлт процесст хэлбэрээр нь оруулах



3. Эцсийн бүтээгдэхүүн: Систем ЮУ чадах ёстой тухай гэрээ

- ❖ Хэрэглэгч нь тухайн хөгжүүлэх гэж байгаа системд ямар шаардлага тавигдахыг хэлж өгнө. Уг шаардлагуудыг систем хөгжүүлэгч шаардлагын бичиг баримт болгон хувиргана. Эцэст нь хэрэглэгчтэй харилцан ярилцсаны үндсэн дээр гэрээ хийнэ.

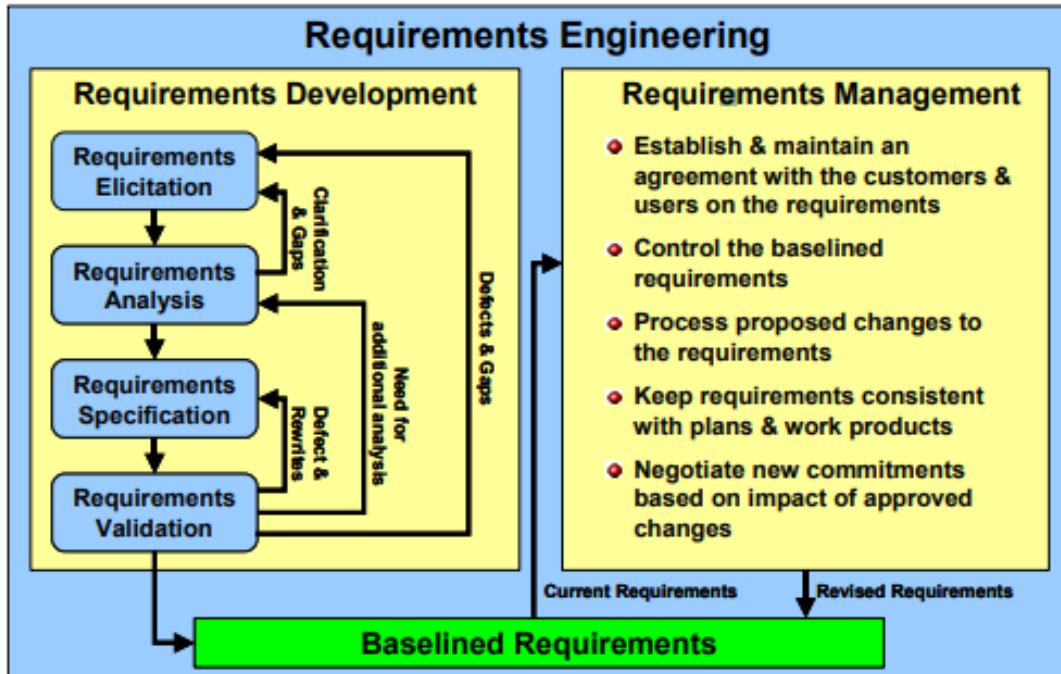


4. Хэрэглэгч хөгжүүлэгч хоорондын уялдаа

- ❖ Хэрэглэгч нь тухайн системийн шаардлагыг өөрийн үгээр тодорхойлоод, хөгжүүлэгч бага болон байгууллагад өгнө. Тухайлбал “Би тийм юмтай/ програмтай болохыг хүсч байна” гэх зэрэг хэлбэртэй байна. Үүнийг хөгжүүлэгч хүлээж аваад, тухайн систем болон програмыг хэрхэн хийх талаарх шийдэл болон ойлголт, төсөөллийг буцаагаад хэрэглэгчдэд өгдөг. Тухайлбал “Би ингэж хийхийг хүсэж байна, ингэж хийх боломжтой/ боломжгүй” гэх мэт хэлбэртэй байна. Үүнээс хэрэглэгч болон хөгжүүлэгч харилцан тохиролцсоны үндсэн дээр гэрээгээ байгуулж, хөгжүүлэлт эхэлдэг.

5. Шаардлагын инженерчлэл буюу шаардлагын шинжилгээ

- ❖ Систем, програм хангамжийг бүтээхийн тулд хэрэглэгч болон хөгжүүлэгч хоорондын шаардлага тодорхойлох үйл ажиллагааг бүтээх процесс буюу үйл явц.
- ❖ *Тодорхой системтэй арга барил*
 - Системд тавигдах шаардлагыг олж тогтоох, баримтжуулах
 - Хэрэглэгч болон төслийн баг хооронд гэрээг тохиролцох, мөн системд тавигдах шаардлагыг өөрчлөгдөхөд гэрээг янзлах
 - Системийн шалгаж болохуйц байдлыг хэрэгчлэгчийн шаардлагатай харьцуулан үүсгэхэд хэрэглэдэг арга техник.
- ❖ *Шаардлагын инженерчлэлийн үед хийгдэх ажил*



- Requirements Elicitation –
Шаардлагыг олж тогтоох
 - Шаардлага олж тогтоох
аргууд:
- Requirements Analysis –
Шаардлагад задлан шинжилгээ хийх, тухайлбал шаардлагыг томьёолох, ангилах, шатлан эрэмбэлэх
- Requirements Specification –
Тухайн шаардлага сайн шаардлагын шинж чанар, хэмжүүрийг хангаж байгаа эсэхийг шалгаж, нягтлах. Системийн хөгжүүлэгч, хэрэглэгч, өөр бусад сайн дурын хүмүүс энэхүү үйл ажиллагаанд оролцох боломжтой.
- Requirements Validation – Шаардлагыг мөшгөх, системийн шаардлагын дагуу нотлох буюу тухайн систем хийгдсэний дараа тавигдсан шаардлагынхаа дагуу хийгдсэн эсэхийг баталгаажуулах нотлох үйл ажиллагаа.

❖ *Шаардлагын инженерчлэлийн сүл тал*

- Шаардлагыг тайлбарлан бичихэд зориулсан илэрхийлэх хэрэгслээр хамааран ойлгомжтой байдал болон дахин боловсруулж болохуйц байдал нь алдагдах
- Шаардлагыг олж тогтооход гарах асуудлууд/системийн шинжээчийн туршлагаас их хамаардаг:



- Бүрэн бус, зөрчилтэй мэдээлэл
- Ялгаа болон зөрчилийг олж танихгүй байх
- Зөрчилтэй тохиолдолд ойлгомжтой шийдвэр байхгүй
- Байж болох үр дагаварыг харуулдаггүй
- Хэрэглэгчийн талын ялгаатай сонирхол
- Системийн орчинг ойлгомжтой тайлбарлаж бичээгүй
- Системийн орчин өөрөө өөрчлөгддөг
- Mash их туршлагатай системийг шинжээч эдгээрийг гүйцэтгэдэг.

❖ Эрсдэлийг шийдэх

- Эрсдэл:
 - Чухал шаардлагыг анзаарахгүй орхигдуулах – Дутуу дульмаг шинжилгээ хийх, хүнд шаардлагаас зайлсхийх үел ихэвчлэн гардаг.
 - Тавигдсан шаардлагыг хэрэглэгчид хангалтгүй дурслэх – Хэрэглэгч тавьж байгаа шаардлагаа сайн ойлгоогүй нөхцөлд үүсдэг. Хөгжүүлэгч хангалттай сайн тайлбарлаж, дүрсэлж өгөх
 - Зөвхөн функциональ шаардлагыг загварчлах – Хялбар тул функциональ шаардлагыг тодорхойлоод, бусад шаардлагыг тодорхойлохгүй орхих
 - Шаардлагыг шалгахгүй байх
 - Зохиомжийн шийдлийг таамаглаж авах
- Шийдэх арга зам:

Категорчилох

- Шаардлагыг бүлэглэх, эх идэвхжилтийн загварыг тууштай ашиглах

Зохион байгуулах

- Шаардлагыг ойлгох мөн мөшгөх багаж хэрэгсэлийг ашиглах
- Хатуу өөрчлөлтийн менежмент хийх

Чухлаар нь эрэмбэлэх
- Зайлшгүй байдал болон эрсдэлд суурилан дарааллыг олж тогтоох

- Эрэмбийн дараалал дах өсгөн нэмэх буюу инкрементээр хэрэгжүүлэх

6. Эх сурвалж

- ❖ https://www.youtube.com/watch?v=9fwJy_M5gQs
- ❖ [https://en.wikipedia.org/wiki/Quality_assurance#:~:text=Quality%20assurance%20\(QA\)%20is%20a,quality%20requirements%20will%20be%20fulfilled%22.](https://en.wikipedia.org/wiki/Quality_assurance#:~:text=Quality%20assurance%20(QA)%20is%20a,quality%20requirements%20will%20be%20fulfilled%22.)
- ❖ <https://www.geeksforgeeks.org/software-engineering-requirements-engineering-process/>
- ❖ <https://medium.com/@mrummanhasan/software-requirements-engineering-what-why-who-when-and-how-bcf5fa729e3b>
- ❖ <https://www.testbytes.net/blog/requirements-elicitation/>
- ❖ <https://anarsolutions.com/top-5-requirements-elicitation-techniques/>

Лекц № 3 – Шаардлагын тодорхойлолт

1. Шаардлагын тодорхойлолтын төрөл



Олон төсөлд:

- **Хэрэглэгчийн шаардлага** – Ихэнхдээ **ярианы хэлээр** буюу хэрэглэгчийн уг хэллэгээр бичигдсэн өгүүлбэрүүдийг агуулсан байдаг. Мөн хэрэглэгч өөрийн шаардлагыг ойлгомжтой болгохын тулд зураг, дүрс график ашигласан байж болно.
- **Системийн шаардлага** – Хэрэглэгчийн шаардлагыг авч, **шаардлагын инженерчлэл** хийсний дараах **ур дүн** гэж үздэг. Мөн **формаль** буюу математик загвартай хэрэглэгч ойлгох гэхээсээ илүү хөгжүүлэгчдэд зохимжтой байдлаар зураг, схем, диаграммуудаар баяжуулсан маш дэлгэрэнгүй байдлаар тодорхойлогдсон **тайлбар тодорхойлолт**.
- **A formal software specification** - Албан байдлаар тодорхойлогдсон үгийн сан, синтакс, семантик буюу утга зүйтэй хэлээр илэрхийлсэн мэдэгдэл юм. Албан буюу формаль семантик тодорхойлолт гэдэг нь тухайн тодорхойлолтыг байгалийн хэлээс тэс ондоо буюу математик хэв загвар дээр суурилан тодорхойлсон тодорхойлолт юм. Иймд уг тодорхойлолт нь хэрэглэгч бус хөгжүүлэгчид илүү ойлгомжтой байдлыг үүсгэхийг эрмэлздэг.
- **Хүндрэл** – Хэрэглэгч формаль тайлбарыг ойлгохгүй.

- Тусламж – Туршилтийн загвар /prototype/ гаргаж танилцуулах, хэрэглэгчийн шаардлагын график загварчлал буюу системийн төлөв байдлыг аль болох энгийн байдлаар харуулж, тайлбарлах шаардлагатай.

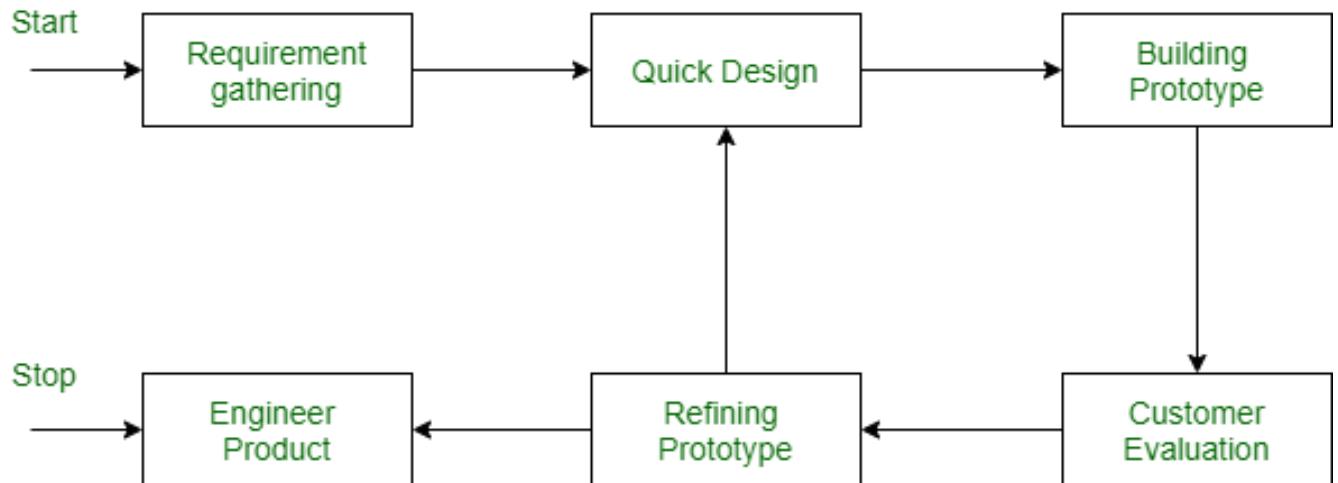


Figure - Prototype Model

Strengths and Weaknesses of Prototyping

- Strengths
 - Develops close working relationship between designer and users
 - Best method for systems with hard-to-define specifications
 - Works well with other development tools
- Weaknesses
 - Not appropriate for every type of system
 - Sometimes rushes the development process

2. Шаардлагыг ялгаатай өнцгөөс харах

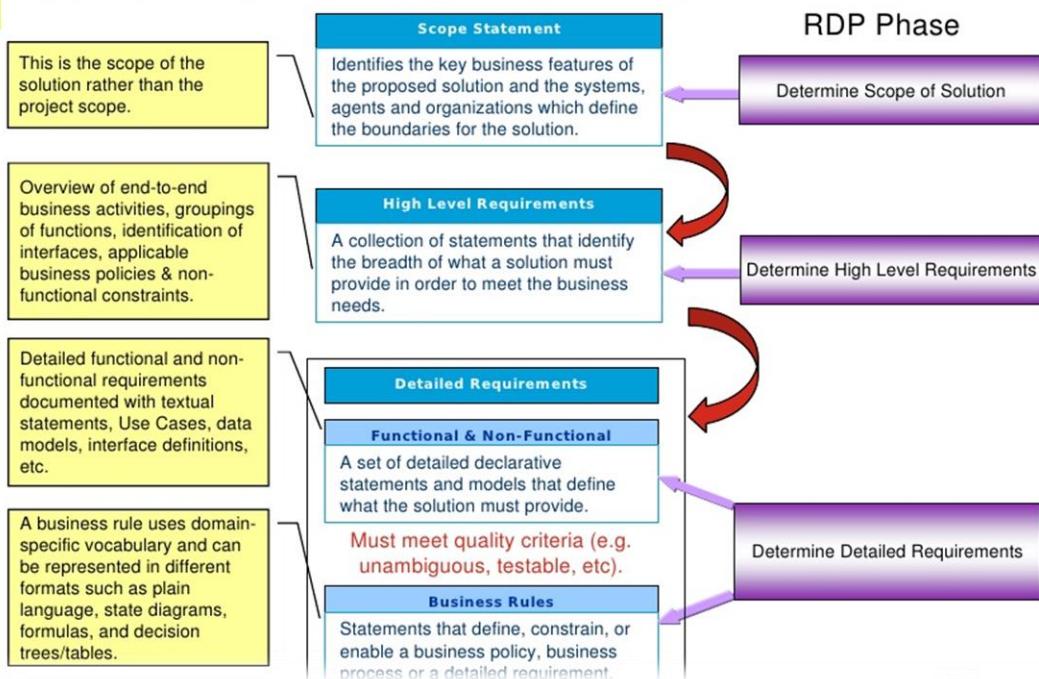
- ❖ Асуудлыг харах ялгаатай хэв маяг, арга зам байдаг бөгөөд асуудлыг маш олон өнцгөөс харж, загварчилж, тодорхойлж болдог. Системийг өндөр, нарийвчлан авч үзэхийн тулд үүнийг хийх шаардлага тулгардаг.
- ❖ Асуудал
 - Системийн шаардлагыг шинжлэх **ямар нэг зөв арга зам** байдаггүй. Тиймээс ялгаатай харах өнцгөөр шинжилгээ хийх нь чухал юм.
- ❖ **Жишээ: банкны автомат**
 - Банкны үйлчлүүлэгч
 - Засвар үйлчилгээний инженер
 - Зах зээлийн судалгааны хэлтэс
 - Банкны менежер, теллер
 - ΘС – ийн админь аюулгүй байдлын ажилтан
 - Харилцаа холбооны инженер
 - Хүний нөөцийн хэлтэс гэх зэрэг хүмүүсээс шаардлагыг нь олж, илрүүлж, боловсруулж, тодорхойлж бичи хэрэгтэй.

3. Шаардлагын тодорхойлолт боловсруулах

- ❖ Шаардлагыг олж тогтоох – шаардлага нэг бүрийг олж тогтоох
 - Захиалагч болон хэрэгчлэгчтэй уулзаж, шаардлагыг тодорхойлж авах.
 - Гүйцэтгэгч талын шаардлагыг тодорхойлж авах.
 - Орчны шаардлагыг тодорхойлж авах
- ❖ Шаардлагыг томьёолох – шаардлагыг өгүүлбэр болгон томьёолж бичих
 - Захиалагч, гүйцэтгэгчтэй нягт хамтран ажиллаж, харилцан зөвлөлдсөний үлдсэн дээр тэдгээрийн хоорондын мэтгэлцээн болон зөрчлийг шийдвэрлэх, шийдвэр гаргах
- ❖ Шаардлагыг ангилах
 - Функциональ шаардлага
 - Өгөгдлийн шаардлага
 - Чанарын шаардлага
 - Гүйцэтгэл буюу чадамжийн шаардлага
 - Хэрэгжүүлэлтийн шаардлага
 - Зохион байгуулалтын шаардлага гэх зэргээр ангилах.

- ❖ Шаардлагыг шаталсан эрэмбэд оруулах

Requirements Hierarchy - Definitions



- ❖ Шаардлагыг шалгах – тухайн шаардлага хаягдах, хэрэгжүүлэгдэхгүй хоцрох, хэрэгжүүлэх үйл явцад асуудал гарах гэх зэрэг асуудлуудаас урьдчилан сэргийлэхийн тулд шаардлага нэг бүрийг шалгаж, хянаж, дүгнэж үнэлэх шаардлагата болдог.
 - Тов тодорхой байдал
 - Бүрэн бүтэн байдал
 - Дараа нь шалгаж болохуйц байдал
 - Нэг хэв шинжтэй байх
 - Өөрчилж болохуйц байдал
 - Мөшгөх буюу гаргалгаа гаргаж болохуйц байдал
 - Хэрэгжүүлж, гүйцэтгэж болохуйц байдал

4. Шаардлагыг бүтэцлэх

- ❖ Шаардлагыг ойлгомжтой болгох, дараа нь хэрэглэхэд хялбар болгох, мөн бүрэн бүтэн байдлыг хянахын тулд шаардлагыг тодорхой бүтцэд оруулах.
- ❖ Ангилсан шаардлагуудаа нарийвчлан задлаж, дэд шаардлагуудад хуваан ангилах
- ❖ Давуу тал:
 - Илүү боловсруулж болохуйц болно.
 - Илүү ойлгомжтой болно.
 - Ялгаж болохуйц шаардлагууд үүснэ.
 - Мөшгөж болохуйц болно.
 - Гэрээний үндэс болгож болохуйц болдог.

5. Шаардлагын тодорхойлолт дахь ухагдахуун ойлголт

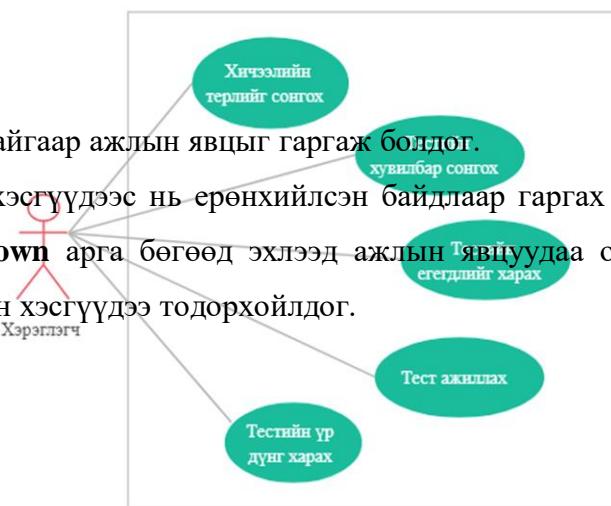
- ❖ Тодорхойлолтыг лавлагааны хэлбэрээр нь дараах байдлаар ангиж болдог.
 - Тухайн баримтыг уншиж байгаа хүн сонирхсон мэдээллээ хурдан олж, ойлгохын тулд
 - Дундын тайлбар толь бичиг буюу лавлах бичиг баримтыг боловсруулах
 - Үгийн жагсаалтын гол үгсийг оруулах
 - Хаана ч хэрэглэж хэрэглэж болохуйц буюу хоёрдмол утгагүй тодорхойлолттой байх
 - Хүчинтэй байх мужийг тогтоох буюу зөвхөн тухайн шаардлагын тайлбар болон ойлголт эсвэл зохиомжийг тодорхойлж өгөх
 - Лавлах байдлаар нь ангилсан байх. Тухайлбал :
 - Аюулгүй байдлынн чухал бүх тодорхойлолт
 - Хяналттай холбоотой бүх тодорхойлолт
 - Хэрэгжүүлэлттэй холбоотой бүх тодорхойлолт... гэх зэргээр ангилна.
- ❖ Шаардлагын тодорхойлолт дах ухагдахууны ойлголт нь :
 - Товч үгийн жагсаалт
 - Ухагдахууны тайлбар толь
 - Ашигласан ном зохиол болон эх үүсвэрүүдийн жагсаалт гэх зэрэг бичиг баримтуудыг шаардлагын бичиг баримтад хавсаргах өгөхийг хэлнэ.

6. Сценаръ – Шаардлага тодорхойлох туслах хэрэгсэл

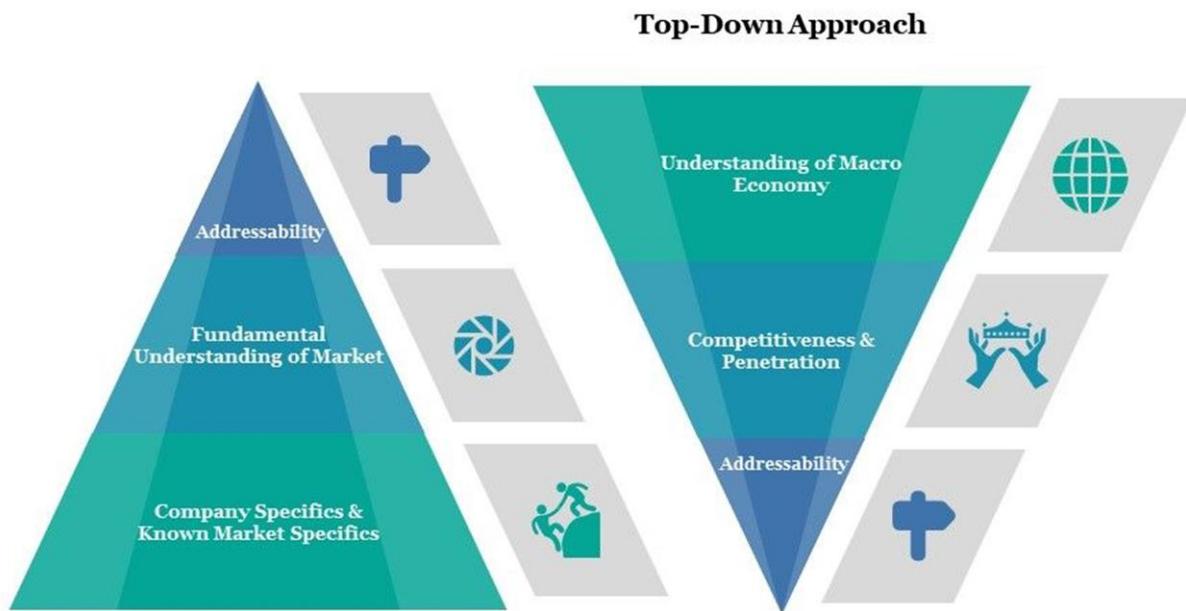
- ❖ Тухайн шаардлага тодорхойлох үйл ажиллагааг хийх арга замууд бөгөөд тухайн нөхцлөөс хамаараад тухайн үйл ажиллагаанд хийгдэх арга зам, үйлдлийн дэс дараалал.
- ❖ Сценаръ нь практикт системийг хэрхэн ашиглах тайлбар бичилтүүдийг хийж өгдөг.
- ❖ Шаардлагыг олж илрүүлэхэд тустай, учир нь хэрэглэгчийг нь хийсвэр өгөгдлөөс илүү сайн ойлгож, шаардлагыг нь тодорхойлдог. Мөн шаардлагыг детальчилан илүү нарийвчлан тайлбарлахад заримдаа их тус болдог. Тухайн үйл явцыг **ажлын явц / use – case** гэж нэрлэдэг.

7. Ажлын явц / - ын диаграмм

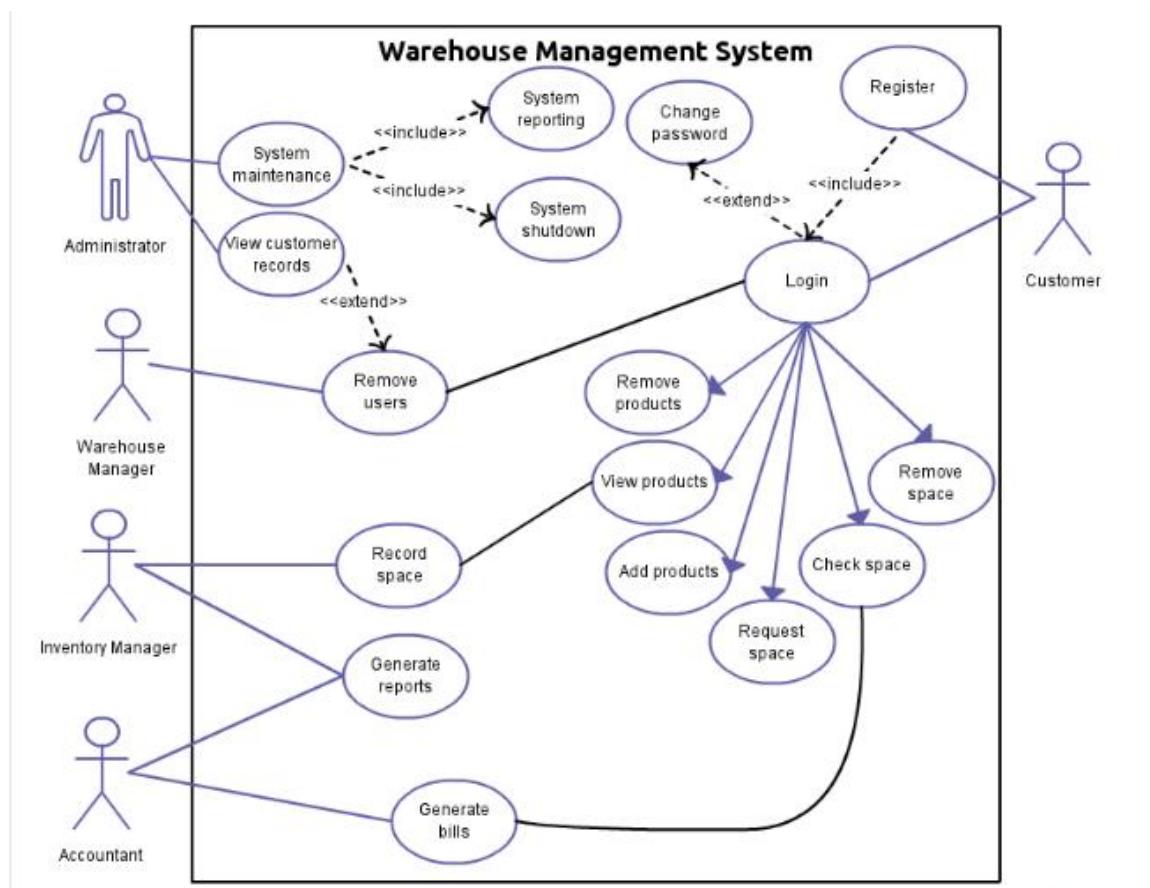
- ❖ Тодорхойлсон сценарын тусламжтайгаар ажлын явцыг гаргаж болдог.
- ❖ Үүнийг **bottom up** буюу нарийн хэсгүүдээс нь өрөнхийлсэн байдлаар гаргах арга гэж нэрлэдэг. Өөр нэг арга нь **top down** арга бөгөөд эхлээд ажлын **Тявцүудаа олчиход, егеджийн харах** дарааагаар сценариуд болон нарийн хэсгүүдээ тодорхойлдог.



Bottoms-Up Approach & Top-Down Approach



Bottoms-Up Approach



- ❖ Жийиээ: Бараа агуулахын системийн ажлын явцын диаграмм
- ❖ Тухайн үйлчлүүлэгчийг хариуцсан ажилтан бараа агуулахийн системийг ашиглаад, тухайн гэрээ, мэдээлэл боловсруулалт болон гүйцэтгэх даалгаврыг хэрэгжүүлдэг. Мөн

accountant буюу нятлан бодох бүртгэлийн хэсэг бүлэг хүмүүс, системийн хэсэг нь энэхүү даалгаврыг гүйцэтгэхэд оролцдог. Бараа агуулахын захиалгын төлбөр тооцооны асуудлыг шийдвэрлэх байдлаар оролцдог.

❖ Агуулахыг захиран зохицуулах үйл ажиллагаанд нийлүүлэгч болон агуулахыг хариуцсан ажилсан. Нийлүүлэгчийн барааг агуулахдаа хэрхэн захиран зохицуулж, зохион байгуулалттай байршуулах үйл ажиллагааг агуулах хариуцсан ажилтан зохицуулдаг.

❖ **Ажлын явцын задаргаа:**

- **Ажлын явц** – Захиалгын Даалгавар (Гэрээт) гүйцэтгэх
- **Зорилго** – Барааг үйлчлүүлэгчид нийлүүлэх
- **Категорь** – анхдагч
- **Дараах нөхцөл амжилттай** – барааг нийлүүлсэн ба хэсэгчилсэн нийлүүлэлт хийсэн.
- **Дараах нөхцөл амжилтгүй** – Үйлчлүүлэгчид боломжгүй байдлыг мэдэгдэх
- **Тоглогч** – Үйлчлүүлэгч хариуцсан ажилтан (XA), AXA, санхүү
- **Өдөөх үзэгдэл** – Үйлчлүүлэгчийн захиалга бэлэн байдаг.
- **Тайлбар:**
 1. Үйлчлүүлэгчийн мэдээллийг дуудаж авчирах
 2. Нийлүүлж болохуйц байдлыг шалгах
 3. Тооцоог хийх
 4. Даалгаврыг агуулахаар гүйцэтгүүлэх
 5. Тооцооны хуулбарыг санхүүд өгөх
- **Өргөтгөл:**
 1. Үйлчлүүлэгчийн мэдээллийг шинэчлэх
- **Хувилбар:**
 1. Шинэ хэрэглэгчийн мэдээлэл бүртгэх
 2. Дараа нь төлөх, тооцоо хийх
 3. Банкнаас шууд татах, тооцоо хийх

8. **Лекцийн асуулт**

- ❖ Шаардлагын нарийвчилсан сайн тодорхойлт нь
- Төслийн нийт өртөгийг багасгах нөлөөтэй
 - Үйлчлүүлэгчийн хүсэл, хэрэгцээг тайлбарлан бичдэг.
 - **Ялангуяа инженерт ойлгомжтой байх ёстой. Хэрэглэж буй бүх хүнд...**
 - **Хэрэгжүүлсний дараа хэрэгцээгүй болдог. Дараа нь ч гэсэн хэрэг болдог...**
 - **Функциональ шаардлагыг л харуулдаг. Бүх шаардлагыг нарийвчлан...**
 - **Голдуу захиалагчийн хүссэн шаардлагыг агуулдаг. Бүх хэрэглэгчдийн...**

9. Эх сурвалж

- ❖ https://www.youtube.com/watch?v=_2YXa6R81hM
- ❖ <https://www.unf.edu/~ncoulter/cen6070/handouts/specifications.pdf>
- ❖ https://en.wikipedia.org/wiki/Formal_specification
- ❖ <http://www.ofnisystems.com/services/validation/user-requirement-specifications/#:~:text=The%20User%20Requirements%20Specification%20describes,with%20input%20from%20Quality%20Assurance.>
- ❖ <https://www.inflectra.com/ideas/topic/requirements-definition.aspx>
- ❖ <https://www.geeksforgeeks.org/advantages-and-disadvantages-of-prototype-model/>
- ❖ <https://slideplayer.com/slide/5185259/>
- ❖ <https://creately.com/diagram-community/popular/t/use-case>
- ❖ <https://creately.com/diagram/example/h89w91171/Warehouse%20Management>
- ❖ <https://creately.com/diagram/example/imqjnpx12/%D0%90%D0%B6%D0%BB%D1%8B%D0%BD%D1%8F%D0%B2%D1%86%D1%8B%D0%BD%D20%D0%B4%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC>

Лекц № 1.4 – Шаардлагыг мөшгөх (Requirements – tracing)

1. Шаардлагын мөшгөлт

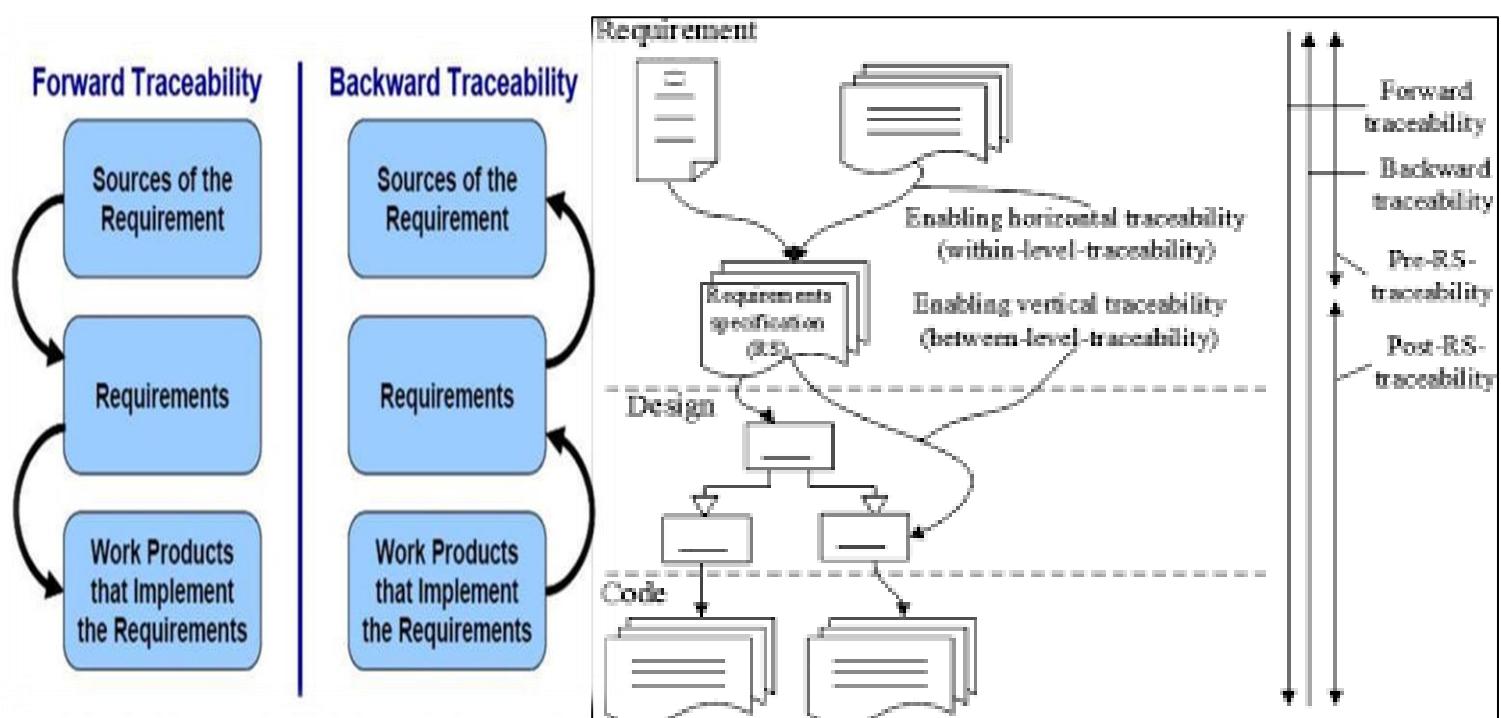
❖ **Mөшгөлт /Tracing/** - Мөрдөж, мөрдлөг хийх, мөрөөр мөшгөж, өмнөх хэсгүүд рүүгээ очих гэсэн утгатай.

❖ **Traceability**

- The ability to discover information about where and how a product was made.
Бүтээгдэхүүн хэрхэн, хаана, яаж бүтээгдсэн талаарх мэдээллийг олж илрүүлэх чадвар. [/https://dictionary.cambridge.org/dictionary/english/traceability/](https://dictionary.cambridge.org/dictionary/english/traceability/)
- Мөрдөж, баталж, мөшгөж болохуйц байдал

❖ **Шаардлагын мөшгөлтийн төрлүүд:**

- Урагш мөшгөх /forward traceability/ - Шалгалтын тохиолдлуудад тавигдах шаардлагыг зураглал хийхэд ашигладаг. Энэ нь бүх шаардлагыг дээрээс доошоо шалгагдаж байгааг тогтоогоод зогсохгүй төслийн траекторыг найдвартай гэдгийг бататгахад туслах боломжтой.
- Ухарч мөшгөх /backward traceability/ - Шалгалтын тохиолдлуудыг шаардлагад нийцүүлэн буулгаж, буцааж мөрдөх боломжтой матрицыг хийж болно.
- Хоёр чигт мөшгөх /bidirectional traceability/ - Урагш болон ухрах мөшгөлтийг хослуулж ашигладаг.

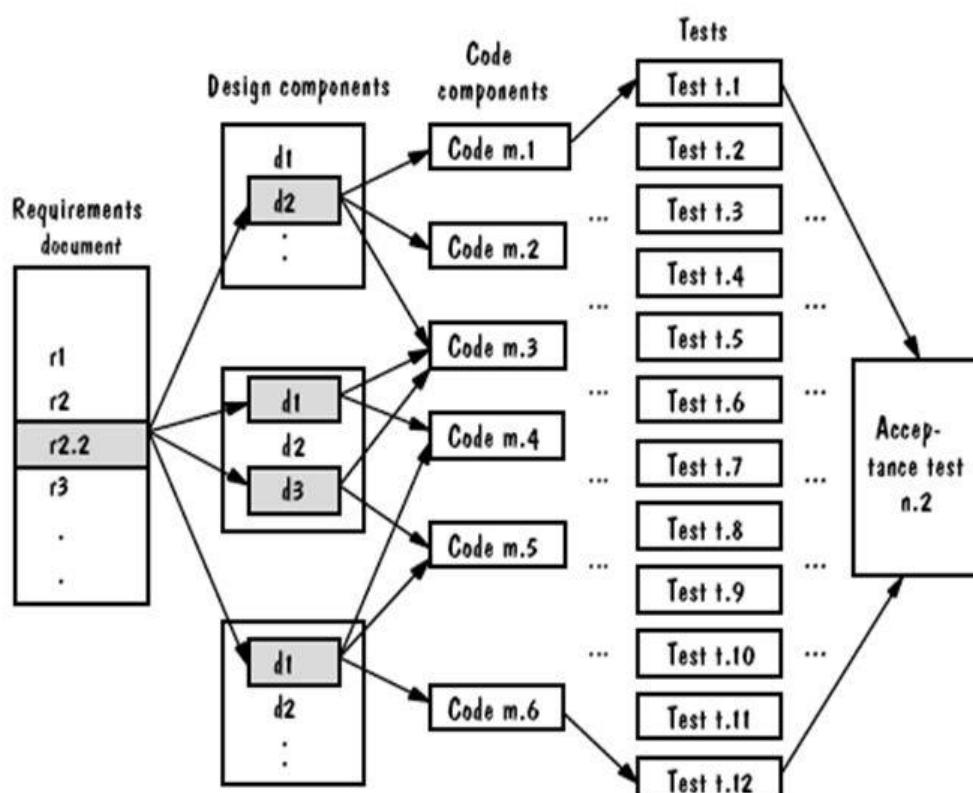


❖ **Шаардлагын мөшгөлтийн давуу талууд**

- Тухайн шаардлагад ямар нэг өөрчлөлт орохын бол буцааж янзлах

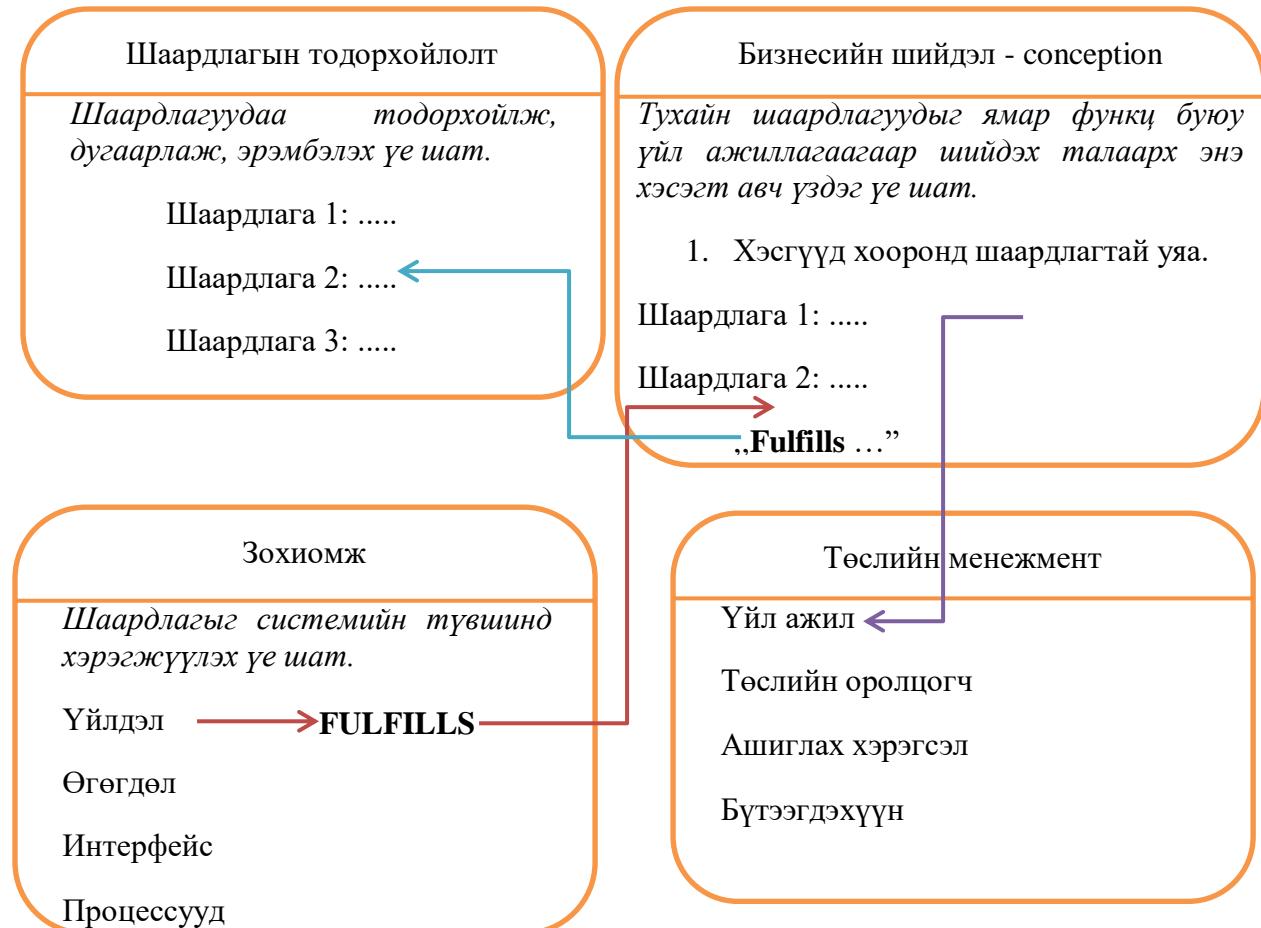
- Өөрчлөлтийг хэн гаргасань хаан хийсэн болохыг олж тогтоох гэх мэт олон талуудтай.
- ❖ *Шаардлагын мөшгөж болохуйц байдал = Мөрдөж + Баталж болохуйц байдал*
 - Шаардлагын мөрдөж болохуйц байдал :
 - Гарал үүслийн – Шаардлагын аль ч шатанд тухайн шаардлагыг санал болгосон хүмүүстэй нь уг шаардлагыг харьцуулж мөшгөх үйл ажиллагаа
 - Шаардлагын – Дэд шаардлагууд буюу бүхэл болон үндсэн шаардлагуудтай хамааралтай, уялдаа холбоотой жижиг шаардлагуудын хамаарал болон уялдаа холбоотой байдлаар нь тухайн үндсэн бүхэл шаардлагуудыг мөшгөх үйл ажиллагаа
 - Шаардлагын баталж болохуйц байдал (шаардлагын бичиг баримт гарсны дараа нь) :
 - Зохоймж буюу хэрэгжүүлэлтийн үед, Тестын үед, Кодын үед

Horizontal traceability in software products

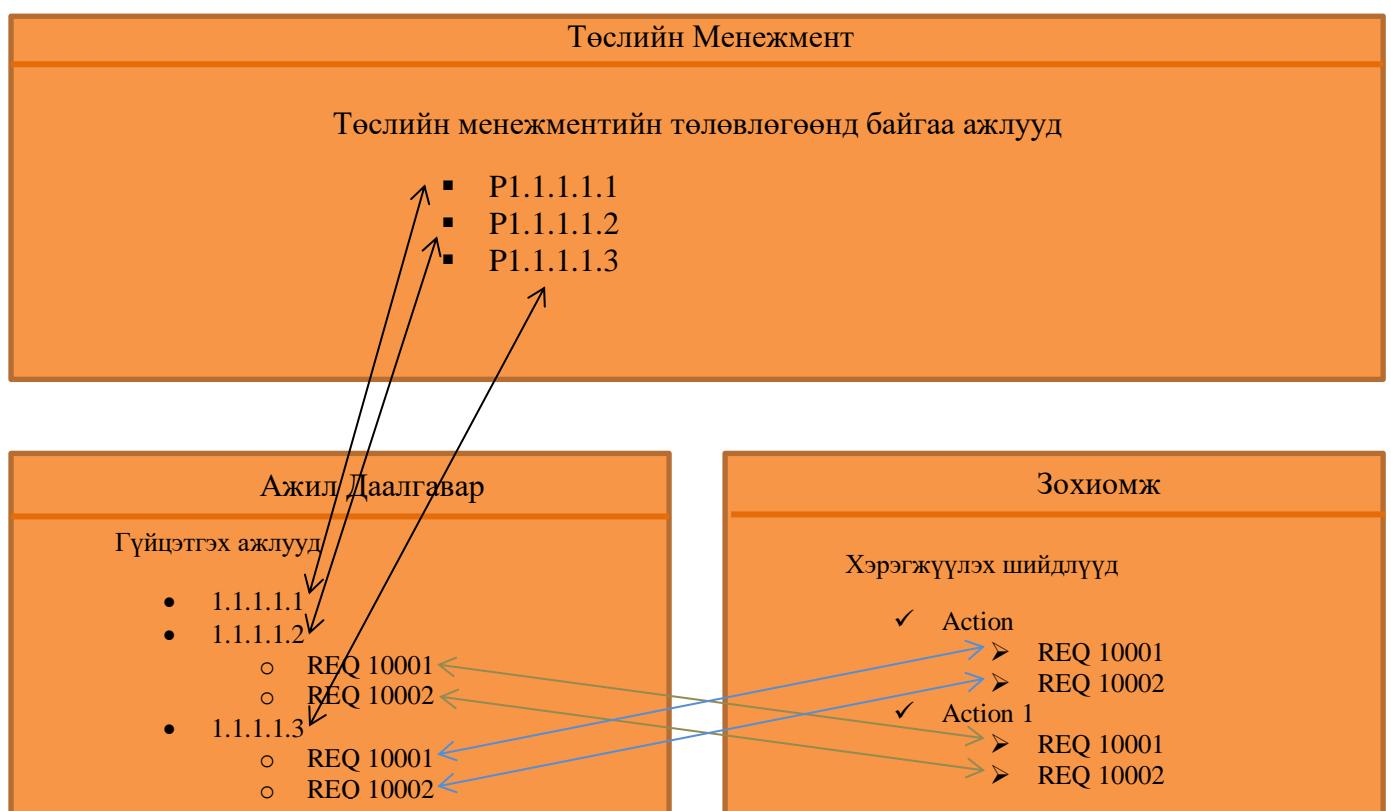


- Шаардлагыг буцааж, мөшгөж хянах болон харах шаардлага гардаг.

2. Уе шат хоорондын мөшгөхүйц байдал



3. Мөшгөж болохуйц байдлын жишээ



4. Шаардлагын өөрчлөлт

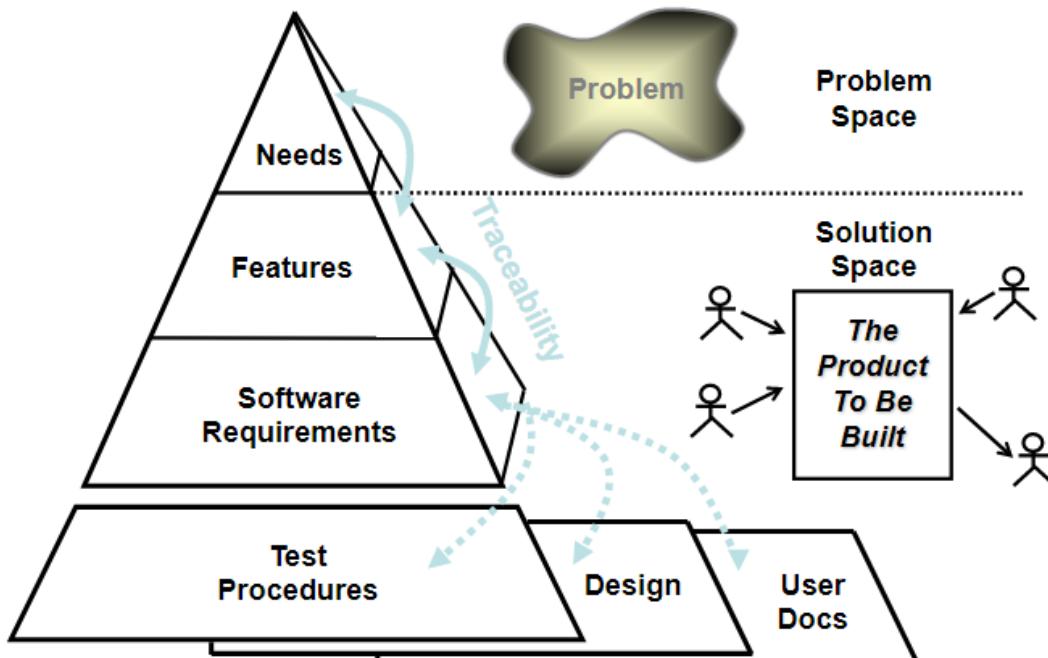
- Хэрэглэгчийн хүсэл бодол өөрчлөгдсөнөөс эсвэл системийн нөхцөл байдал өөрчлөгдсөнөөс гэх зэрэг тодорхой шалтгаануудыг улмаас шаардлагын өөрчлөлт бий болдог. Шаардлага өөрчлөгдөх нь хэвийн зүйл бөгөөд амьдралын нэг цикл мэт зүйл юм.
- Үүсэх гол шалтгаанууд:
 - Хөгжүүлэх процессын явцад шаардлагын зэрэглэл өөрчлөгдөх
 - Эцсийн хэрэглэгчийн шаардлагатай зөрчилдөх шаардлагыг ЭЗ – ийн, санхүүгийн суурь нөхцөлөөс үүдэн үйлчлүүлэгч гүйцэлдүүлэхээр болох
 - Системийн ашиглалтын болон техникийн орчин хөгжүүлэх явцад өөрчлөгдөх

Шаардлагын өөрчлөлтийн менежмент

- ❖ Шаардлагын захиран зохицуулах үйл ажиллагааг шаардлагын инженерчлэл болон систем хөгжүүлэлтийн үед гүйцэтгэдэг.
- ❖ Шаардлага өөрчлөгдөх шалтгаанууд:
 - Шаардлагууд бүрэн бус, зөрчилтэй түүнээс зайлсхийх боломжгүй байдаг.
 - Процессын үед шинэ шаардлага гарч ирдэг, бизнесийн зорилго өөрчлөгддөх эсвэл системийн ойлголт сайжирч байдаг.
 - ¹Өөр өөр хүмүүс ялгаатай өнгөөс харж, ялгаатай шаардлагуудыг тавьдаг. Тэдгээр нь ихэнхдээ зөрчилтэй байдаг.

Managing Changing Requirements - Overview

What

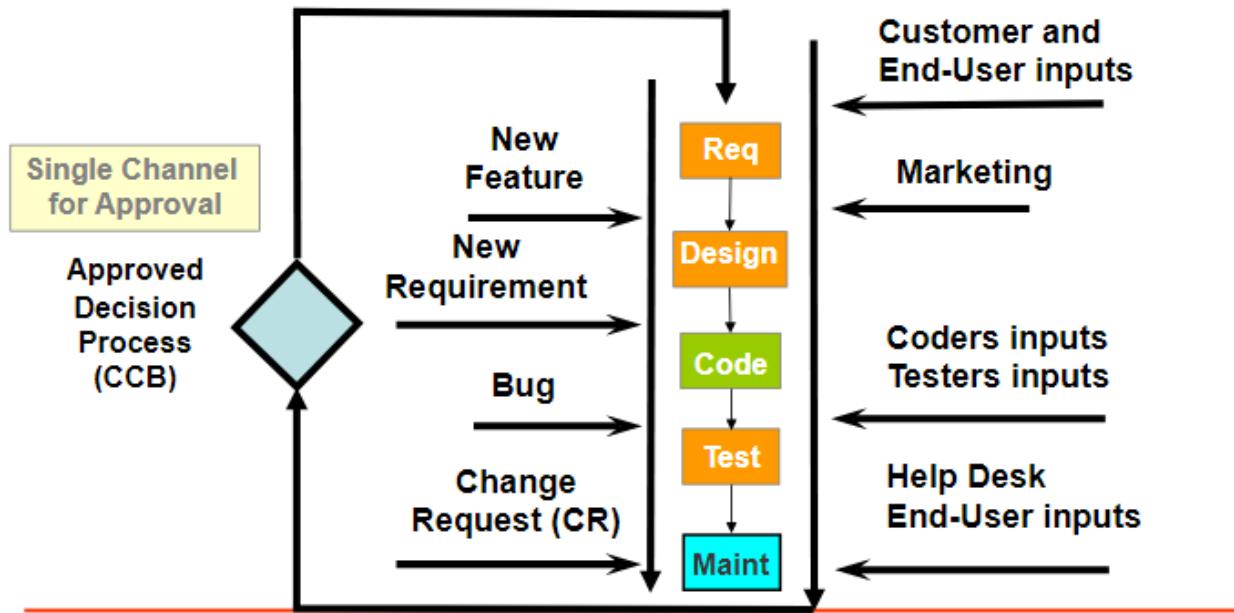


¹ <https://slideplayer.com/slide/5153693/>

A Key to Managing Change

All requests go through a single channel

Change requests come from many sources throughout each iteration of the product lifecycle



Amirkabir University of Technology, Computer Engineering Faculty, Intelligent Systems Laboratory

12

2

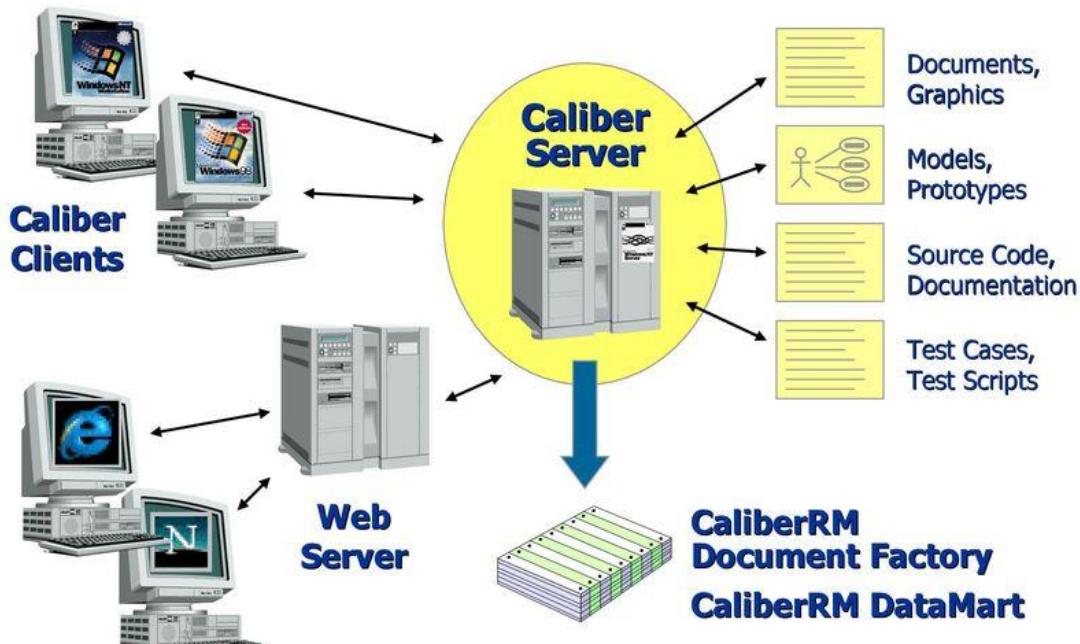
CASE – Багажаар дэмжих

Багаж	Тайлбар
Шаардлагыг архивлах багажууд	Тухайн шаардлагын мэдээллүүд өгөгдлийн санд эмх цэгтэй хадгалах боломжийг нээж өгдөг.
Өөрчлөлтийн менежмент хийдэг програмууд	Процесс болон ажлын урсгалын үе шатуудыг тодорхойлоод, тухайн үе шатуудын хоорондох мэдээллийн урсгалыг хэсэгчлэн автоматжуулах боломжийг олгодог.
Traceability management	Шаардлага хоорондын заагчийг автоматаар асуух боломжийг олгодог.

- Жишээ:

² <https://slideplayer.com/slide/5153693/>

Arquitectura de Borland Caliber RM



Borland®

17 | Copyright © 2005 Borland Software Corporation. All rights reserved. | 30 June 2005

5. Лекцийн батамгах асуултын дүгнэлт

- Requirements – Tracing нь шаардлага болон түүний системийн өөр хэсэг рүү хандсан уялдааг авч үздэг. Мөн хаана ямар шаардлага хэрэгжүүлсэнийг тогтохын тулд ашигладаг.
- Онцгой тохиолдол бус процессийн аль ч үед шинэ шаардлагыг авч үздэг.
- Шаардлага хоорондох заагч буюу холбоосыг хадгалах зайлшгүй шаардлагатай. Учир нь ингэснээр шаардлагыг мөшгөх боломжтой болдог.

6. Хураангуй

- ❖ Шаардлагын инженерчлэл (RE) нь шаардлагыг олж тогтоох, шинжлэх, тодорхойлох, tracing болон өөрчлөлтийн захиран зохицуулах үйл ажиллагааг агуулдаг.
- ❖ Шаардлагын шинжилгээ нь давталттай. Үйлүүд:
 - Шаардлагыг Илрүүлэх, Шинжлэх, Шалгах, Мөрдөх мөн Баталгаажуулах.
- ❖ Систем нь ялгаатай шаардлага бүхий олон нүүр царайтай, мөн ялгаатай харах байдал нь нэг ижил хөгжүүлж буй системд чиглэдэг.
- ❖ Их тохиолдох алдаа

- Үйлчлүүлэгч юу хүссэнийг нь хэрэгтэй байгааг нь бус, өөрийнхөө юу хүссэнийг үйлчлүүлэгчид өгөх нь том алдаа болдог. Тиймээс шаардлагын инженерчлэлийг маш сайн хийх чухал.

7. Эх сурвалж

- <https://dictionary.cambridge.org/dictionary/english/traceability>
- <https://www.youtube.com/watch?v=ZoIpmxjmdgY>
- https://en.wikipedia.org/wiki/Requirements_traceability
- <https://slideplayer.com/slide/5153693/>
- <https://slideplayer.com/slide/5042304/>
- [Dick,-Jeremy;-Hull,-Elizabeth;-Jackson,-Ken]-Requirements Engineering
- <https://www.simplilearn.com/project-management-and-the-requirements-traceability-matrix-article>
- <https://simplicable.com/new/requirements-traceability>
- <https://docplayer.es/9117040-Borland-y-caliberm-framework-tecnologico-y-de-servicios-para-mejorar-la-gestion-de-requisitos.html>

Лекц № 5 – Хэрэглэгчийн шаардлагыг тодорхойлох

1. Хэрэглэгчийн шаардлага гэж юу вэ?

❖ Ямар нэг зүйлийг хийх, ямар нэг ажлыг хөнгөвчлөх, байгууллагын хүчин чадлыг нэмэгдүүлэхэд чиглэсэн байдаг.

❖ Шаардлагыг тодорхойлолт:

- Бүтээгдэхүүнийг хөгжүүлэхэд ашигладаг бөгөөд захиалагчийн тодорхойлж чадахаас илүү өргөн хүрээтэй, нарийн тодорхойлогдсон зүйл байдаг.
- Бүтээгдэхүүний гүйцэтгэдэг байх ёстой зүйл эсвэл түүнд байх ёстой чанар.

❖ Төслийн хувьд тавигдаж байгаа шаардлагууд:

▪ Функциональ

- Тодорхой үүргийг гүйцэтгэх, тодорхой ажлыг хялбарчлах боломжийг хэрэглэгчид өгдөг эцсийн бүтээгдэхүүнд байх ёстой бүрэлдэхүүн хэсэг.
- Бүтээгдэхүүнийг бий болгох үндэс суурь бөгөөд эцсийн бүтээгдэхүүнд ийм ийм үйлдэл гүйцэтгэдэг функциүүд байх, тэр ямар нэг ажлыг гүйцэтгэж чаддаг байх гэх зэрэг нөхцөл шаардлагуудыг тавьж өгдөг.
- Хэр олон функциональ шаардлага гарч ирэх нь бүтээгдэхүүний хэмжээ, түүний цогц байдлаас хамааралтай.

Функциональ шаардлагын жишээ:

- Эцсийн бүтээгдэхүүн өдөр бүрийн гүйлгээг хаахад тухайн өдрийн нийт бүртгэгдсэн орлого, зарлагыг автоматаар нэгтгэж сарын эцсийн тайланд тусгадаг функциүүдтэй байх ёстой гэх мэт.
 - Эндээс бүтээгдэхүүн хэрэглэгчид ямар нэг үйлдлийг гүйцэтгэхэд туслах, хялбар атлаа хөдөлмөр их зарцуулдаг ажиллагаануудыг өөрөө автоматаар байх шаардлагатай.
 - Дээрх жишээнээс харахад хэрэглэгчийн тодорхойлсон энгийн шаардлагаас хэдэн зуун функциональ шаардлага урган гарч болдог.

▪ Функциональ бус

- Бүтээгдэхүүнд байх ёстой чанар эсвэл шинж чанар.
- Бүтээгдэхүүний амжилтанд чухал нөлөөтэй байдаг ч ихэнх тохиолдолд эцсийн бүтээгдэхүүнийг, захиалагч байгууллагад тохиромжтой, үр өгөөжтэй байх талаас нь тодорхойлсон шаардлага юм.
- Захиалагчийн хувьд бол функциональ шаардлагуудаас илүү чухлаар тавигддаг.

Функциональ бүс шаардлагын жишээ:

- Бүртгэлийн системийн баруун дээд буланд захиалагч байгууллага болох “Өгөөж” компанийн лого байрлах ёстой гэх мэт.
 - Энэ нь бүтээгдэхүүний функциональ хэсэгт нөлөөлөхгүй бөгөөд захиалагч компанийн хувьд тус бүтээгдэхүүн дээр өөрийн лого – г байрлуулах нь маркетингийн чухал ач холбогдолтой.

2. Төслийн шаардлагуудыг тогтоох

- ❖ Ажлын тодорхойлолт болон төслийн бусад холбогдох баримт бичгүүдийг анхааралтай судалснаар төслийн шаардлагыг хялбархан тодорхойлж болдог.
- ❖ *Шаардлагуудыг таних процесс:*
 - Хэрэглэгчийн шаардлагыг бүрэн зөв тодорхойлох нь хүндрэлтэй, их хугацаа шаардсан ажил учраас хөнгөвчлөхийн тулд дараах алхмуудыг дэс тодорхой дарааллын дагуу хийх хэрэгтэй болдог.
 - Эдгээр дөрвөн алхам алхмууд нь дотроо дэд алхмуудтай болно.
 - 1. Тухайн төсөл бидний хийх хэрэгтэй төсөл мөн эсэхийг тодорхойлох
 - 2. Захиалагчийн шаардаж байгаа онцгой нөхцлүүдийг судалж үзэх
 - 3. Төсөлтэй холбогдох баримт бичгүүдэд тусгагдсан бүхий л шаардлагуудыг анхааралтай судлах
 - 4. Шаардлага бүрийг төслийн төлөвлөгөөний аль хэсэгт гарч ирэх вэ гэдгийг тодорхойлох

❖ Төслийг хэрэгжүүлэх эсэхийг тодорхойлох

- Төслийг хэрэгжүүлэх эсэхээ шийдэхдээ дараах асуултуудад хариулт авах байдлаар тодорхойлж болдог
 - 1. Төсөл бидний эрхэлдэг үндсэн бизнестэй харшлах уу?
 - 2. Төслийг хэрэгжүүлэх нь компанийн зорилго зорилтуудад харшлах уу?
 - 3. Энэ төслийг хэрэгжүүлэх ямар туршлага бидэнд байгаа вэ?
 - 4. Бидний техникийн хүчин чадал ямар вэ?
 - 5. Бид захиалагчийн талаар юу мэдэх вэ?
 - 6. Захиалагч бидний талаар юу мэдэх вэ?
 - 7. . Захиалагч бидний талаар юу мэдэх вэ?
 - 8. Төслийн багийн гишүүдийг гаднын байгууллагуудаас сонгон авснаар төслийг амжилттай хэрэгжүүлэхэд тус болж чадах уу?

9. Захиалагчийн хувьд бид тухайн бүтээгдэхүүнийг хөгжүүлж чадах гол байгууллага байж чадах уу?
10. Бидний өрсөлдөгч хэн бэ?
11. Бидний өрсөлдөгч биднээс хүчтэй юу эсвэл сул уу?
12. Бидэнд бүтээгдэхүүнийг бий болгож чадах хангалттай нөөц, хүчин чадал бий юу?
13. Бидэнд гэрээг байгуулж бүтээгдэхүүнийг үйлдвэрлэж эхлэх ямар боломж байна вэ?
14. Бүтээгдэхүүнийг амжилттай хийж дуусгах ямар боломж байна вэ?
15. Төслийг хэрэгжүүлж эхлэхэд биднээс ямар хэмжээний зардал гарах вэ?

Асуулт

1	2	4	5	6	7	8	9	10	11	Эцэст нь
Тухайн төслийг хэрэгжүүлэх нь компанийн эрхэлдэг үндсэн бизнесийн үйл ажиллагаанд харшлах уу бидний хэтийн зорилгод нийцэж буй эсэхийг тодорхойлно	Байгууллагын дотоод хүчин чадал нь тухайн төслийг хэрэгжүүлэхэд хүрэлцэх эсэхийг тодорхойлно.	Захиалагч болон үйлдвэрлэгч нэг нэгнээ судалж мэдэх нь төслийн амжилттай хэрэгжилтэд сайнаар нөлөөлдөг.	Төслийн багийг ямар байдлаар бүрдүүлэх вэ гэдэг шийдвэрийг гаргахад чухал нөлөөлтэй.	Захиалагчийн хүссэн бүтээгдэхүүнийг үйлдвэрлэж өгөх боломжтой ямар ямар компаниуд байна, тэдгээр нь захиалагчтай ямар харилцаа холбоотой гэдгийг тогтооно.	Zахиалагчийн хүссэн бүтээгдэхүүнийг үйлдвэрлэж өгөх боломжтой ямар ямар харилцаа холбоотой гэдгийг тогтооно.	Асуултууд бүгд эсвэл дийлэнх нь эерэг байвал төслийг хэрэгжүүлэхэд ямар нэг саад бэрхшээл байхгүй.				

❖ Хэрэглэгчээс тавигддаг онцгой нөхцлүүд.

- Төслийн хувьд төслийн болон бүтээгдэхүүний шаардлагуудаас гадна төсөлд хэрэглэгчээс тавигддаг онцгой нөхцлүүд байдаг бөгөөд эдгээрийг захиалагчид бүтээгдэхүүний ерөнхий шаардлагаас илүү чухлаар авч үздэг.

Үндсэн нөхцөл шаардлагууд

Цаг хугацааны нөхцлүүд	Тайлан, болон түүнд агуулагдах	Төсөв хөрөнгийн талаар	Холбогдох стандартууд	Хөгжүүлэлтийн болон чанарын	Ерөнхий гэрээнээс бусад дагалдах
------------------------	--------------------------------	------------------------	-----------------------	-----------------------------	----------------------------------

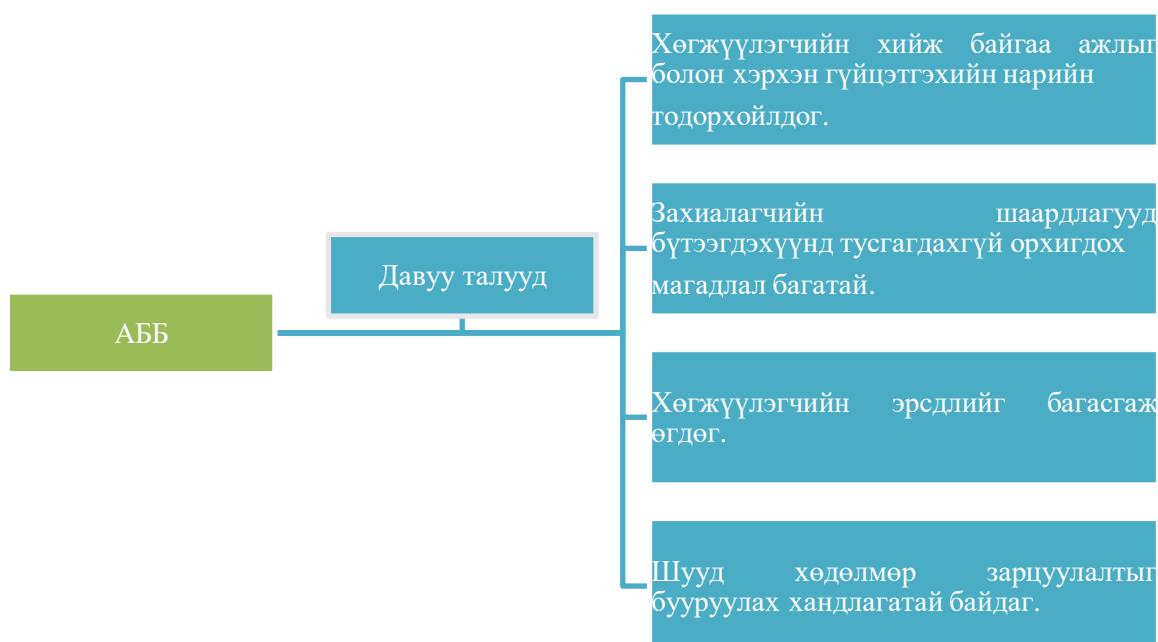
	мэдээллүүдийн талаар тайлагнах	тайлагнах	стандарт	гэрээ хэлэлцээрүүдэд оролцохгүй байх
Тодорхой хугацааны дотор захиалсан бүтээгдэхүүнийхээ үр шимийг хүртэж эхлэх нь захиалагчийн хувьд бизнесийн амжилт, Үр ашигтай чухал зүйл учраас энэ нөхцөл шаардлагыг тавьдаг.	Захиалагчид төслийн гүйцэтгэлийг амжилттай хэрэгжиж байгаа эсэх талаар хангалттай мэдээлэлтэй байхын тулд уг нөхцөл шаардлагыг тавьдаг.	Захиалагч бүтээгдэхүүнийг үйлдвэрлэх төсвийн тооцог сайн хийгээгүйгээс төсөв хөрөнгийн асуудал хязгаарлагч хүчин зүйлүүдийн нэг болж байдаг.	Бүтээгдэхүүний хөгжүүлэлтэд тухайн салбарт мөрдөгдөг эсвэл зайлшгүй шаардлагатай бусад стандартуудыг хангаж өгөх шаардлагатай тулгардаг.	Хөгжүүлэгчид тухайн бүтээгдэхүүнтэй холбоотой үндсэн гэрээ хэлэлцээрт оролцох.

3. Ажлын баримт бичиг (АББ) буюу Statement of Work(SOW)

- ❖ Хийгдэх ёстой ажлуудыг бүрэн дүүрэн тодорхойлсон баримт бичиг байх бөгөөд ямар ажлын ямар баг хийхийг зааж өгсөн хэсэгтэй байдаг. Тухайн баримт бичиг нь захиалагчид төслийн гүйцэтгэлийг үнэлэх шалгуур болж өгдөг.
- ❖ Ажлын баримт бичгийг юун дээр суурилж бүрдүүлснээс шалтгаалж, ерөнхийд нь гурван үндсэн төрлийн ажлын баримт бичигт хувааж үзэж болдог. Үүнд:

▪ Дизайн буюу нарийвчилсан тодорхойлолт төрлийн АББ

- Үйлдвэрлэгчид тухайн ажлыг хэрхэн хийхийг тодорхойлсон байдаг бөгөөд нь ажлын гүйцэтгэлийг хэмжих хэмжүүрүүд, материалууд,



чанарын хяналтын шаардлагууд болон хөгжүүлэгчид шаардлагатай бусад тодорхойлолтуудыг агуулсан байдаг.

▪ **Хөдөлмөр зарцуулалтын түвшиний АББ**

- Тухайн бүтээгдэхүүнийг үйлдвэрлэхэд ямар хөдөлмөр зарцуулалт орохыг тодорхойлдог. Эцсийн бүтээгдэхүүнийг хэмждэг хэмжүүр нь хүн цаг, хүн сар, гэх мэтийн хэмжүүрүүд байна

▪ **Гүйцэтгэл сууринсан АББ**

- Тухайн ажлыг хэрхэн хийхэд гэдэгт бус тухайн ажлын зорилгод чиглэдэг.

• Үндсэн Зн давуу тал:	
	<ol style="list-style-type: none">1. Хөгжүүлэгч захиалагчийн шаардлагыг хангахын тулд дурын шийдлийг боловсруулж болдог.2. Захиалагч нь хөгжүүлэгчийн ажлын гүйцэтгэлийг хянахын оронд зөвхөн тухайн ажлын зорилго амжилттай биелж байгаа эсэхийг хянахад л хангалттай байдаг.3. Зардлын хэмнэлттэй.

▪ **Мөн дээрх төрлийн АББ – уудыг хослуулан хэрэглэж болдог.**

- ❖ АББ-г мэргэжлийн болон мэргэжлийн бус ямар ч хүн уншаад ойлгохуйц энгийн бичсэн байх хэрэгтэй.
- ❖ **Алдаа мадаггүй, бүрэн гүйцэд тодорхойлсон АББ бичихэд дараах хүчин зүйлс нөлөөлдөг.**

- Төслийн багийг болоод төсөлтэй ямар нэг байдлаар холбогдох бүх хүмүүст хэрэглэгчийг шаардлагыг сайтар ойлгуулах
- Төслийн багийг өртөг болон цаг хугацааны тооцооллыг байж болох хамгийн бодитойгоор хийх бүх бололцоог хангах
- Төслийн даалгавруудыг гүйцэтгэх дарааллыг өөрчлөх бололцоо аль болох бага гаргах. Энэ нь төслийн өртгийг нэмэгдүүлэх хандлагатай байдаг.
- Ажлын гүйцэтгэлийн шалгуурыг сайтар тогтоож өгөх
- Захиалагч болон төслийн багийг аль алийг нь төслийн гүйцэтгэлд хяналт тавих боломж олгох

- ❖ АББ-г бичдэг олон төрөл байдаг боловч үр дүнтэй ашиглагддаг өрөнхий нэг л аргачлал байдаг.
- ❖ АББ-ийн өрөнхий хэлбэр

▪ 2. Үндэслэл

▪ 3. Хэрэглэгдэх баримт бичгүүд

- Тодорхойлолт
- Стандартууд
- Аж үйлдвэрийн болон байгууллагын баримт бичгүүд
- Бусад холбогдох баримт бичгүүд

▪ 4. Шаардлагууд

- Ерөнхий
- Төслийн нарийвчилсан шаардлагууд
 1. системийн инженерчлэл
 2. системийн шинжилгээ болон хяналт
 3. Baseline буюу чиг шугамыг тодорхойлох
 4. Програм хангамжийн зохиомж
 5. Техник хангамжийн зохиомж
 6. Сургалтын төлөвлөгөө
 7. Системийг хүлээлгэн өгөх буюу суурилуулах
 8. Үйл явцын концепт
 9. Захиалагч буюу хэрэглэгчийг засвар үйлчилгээгээр хангах
 10. Зохиомжийг эргэж харах
 - i. Системийн шаардлагуудыг эргэн харах
 - ii. Системийн зохиомжийг эргэн харах
- Програмын менежмент
 1. Програмын менежментийн систем
 2. Эрсдийг удирдах менежмент
 3. Амьдралын циклийн өргтийн тооцоолол болн хяналт

▪ 5. Хүлээн зөвшөөрөх шаардлагууд

- Ерөнхий чиг шугам
- Хөгжүүлэгчийн хүлээн зөвшөөрөх хэмжүүрүүд
 1. Бүтээгдэхүүний milestones
 2. Тест болон шалгалт хийх шаардлагууд
- Хөгжүүлэгчийн үүрэг хариуцлага

▪ 6. Илтгэлийн шаардлагууд

- Хяналтын уулзалт
- Статус репорт

4. Төслийн шаардлагуудыг бүрэн тодорхойлох

- ❖ Хэрэв байх ёстой шаардлагуудын 99 хувь нь эцсийн бүтээгдэхүүнд бүрэн тусгагдсан байвал үүнийг сайн үзүүлэлт гэж үзэж болдог.
- ❖ Төслийн цогц байдлын, хөдөлмөр зарцуулалтын, төслийн техникийн, инженерийн тодорхойлолтууд гэх мэтийн хэтэрхий олон баримт бичгүүд нь хариуцаж буй хүнийг төрөгдүүлдэг. Үүнээс гарах 2 арга зам байдаг. Үүнд:
 - (I) *Төслийн шаардлагыг тодорхойлох хамгийн чухал баримт бичиг бол SOW (Statement of Work) гэдгийг санах хэрэгтэй.*
 - Бусад баримт бичгүүд дэх төслийн хэмжээ гэх мэт зүйлсийн үзүүлэлт нь SOW-оос зөрвөл голлон дагаж мөрдөх баримт бичиг нь SOW байна.
 - (II) *SOW бол шаардлагыг тодорхойлсон цорын ганц баримт бичиг биши гэдгийг санах хэрэгтэй.*
 - Ямар нэг дүгнэлт хийх, шийдвэр гаргахад SOW-д орсон материал хангалттай биш бол өөр бусад шаардлагыг дэлгэрэнгүй тодорхойлсон материалуудыг сөхөж үзэх нь зүйтэй.

5. Шаардлагын матриксыг боловсруулах

- ❖ Төслийн шаардлагуудыг АББ-ийн аль хэсэгт байгааг мэдэхэд тусалдаг чухал хэрэгсэл.
- ❖ Матриксын жишээ:

Шаардлагууд	АББ	Тодорхойлолт	Зураглал	АББ
Төслийн шаардлагатай дэд бүтцийг бий болгох	Para 1, цар хүрээний тодорхойлолт	Танилцуулга		2.3.2 dist, мэдээлэл технологийн дэд бүтэц

6. Дүгнэлт

- ❖ Хэрэглэгчийн шаардлагыг бичихдээ хэрэглэгчийн шаардлага хаанаас үүссэн талаар мөрдөх боломжтой байх нь зүйтэй бөгөөд тухайн шаардлага нь ганц үйлчлүүлэгч эсвэл бүтээгдэхүүний менежерээс ирсэн эсэхээс үл хамааран яг хаанаас ирснийг нь

нарийн ойлгох нь чухал байдаг. Иймд тухайн шаардлагуудын ул мөрийг олж тогтоох, боловсруулахад дээрх ухагдахуун болон арга техникиүүдыг ашигладаг. Өгөгдсөн материалыг өөрт ойлгомжтой байдлаар боловсруулж, шаталсан бүтцээр загварчилж, ухагдахуун ойлголтуудыг тодорхойлолтыг өөрийн үгтэй хослуулж найруулсан болно. Мөн зарим ойлгоогүй ухагдахууны талаар дэлгэрүүлж судалсан бөгөөд нэмэлтээр тухайн мэдээллийн эх сурвалжуудыг хавсаргав.

7. Эх сурвалж

- ❖ Лекц Нэмэлт – Шаардлага Ажлын даалгавар.pdf
- ❖ <https://www.guru99.com/requirements-tracing-matrix.html>
- ❖ <https://www.sciencedirect.com/topics/engineering/user-requirement>
- ❖ <https://www.projectmanager.com/blog/statement-work-definition-examples>
- ❖ [Dick,-Jeremy;-Hull,-Elizabeth;-Jackson,-Ken]-Requirements Engineering
- ❖ Software-Engineering-9th-Edition-by-Ian-Sommerville

Лекц № 6 – Use case /ажлын явц

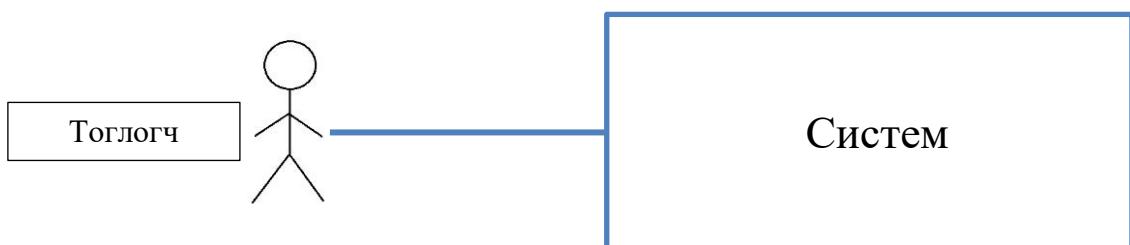
1. Ухагдахуун – Ажлын явц/use case

❖ Тодорхойлолт:

- Ямар нэгэн зорилгод хүрэхийн тулд юм уу эсвэл хүссэн үр дүнг бий болгохын тулд тоглогчоор хийгддэг, хоорондоо уялдаатай олон тооны үүрэгт ажлаас тогтдог.



- **Тоглогч** - Системийн хэрэглэгчийн тоглож буй үүрэг болон хүн бөгөөд системд үргэлж нөлөөлж байдаг. Мөн зохион байгуулалтын нэгж эсвэл өөр систем



байж болдог бөгөөд системийн гадагш байна.

- ❖ Анх Ivar Jacobson объект хандлагат ажлын явц буюу use case – ийг оруулж ирсэн. Мөн use case – ийг ажил хэргийн явц, бизнесийн процесс, ашиглах тохиолдол гэх зэргээр орчуулдаг.

2. Ажлын явцын зорилго

- ❖ Хэрэгжүүлэх гэж буй програм хангамжийн ашиглалтын үеийн үр дүн хандлагат ажлын явцын задаргааг хийж, нарийвчилан тодорхойлох.
- ❖ **Хүссэн үр дүнд** хүрэхийн тулд ямар даалгаврыг програм хангамжийн шинэ системээр хийж дуусгахыг олж тогтоох – **Функциональ шаардлагыг** олж тогтооход хэрэглэж болно.

3. Мэдээллийн систем дэх ажлын явц – Байгууллагын үйл явц биши

- ❖ Хоорондоо уялдаатай **цогц ажиллагааны** (**transaction** – цогц үйлдэл) дараалал бөгөөд энэ нь тоглогч системтэй харилцсаны үндсэн дээр хэмжиж болохоор **утга бүхий үр дүнг** бий болгохын тулд гүйцэтгэгддэг.

- ❖ **Байгууллагын үйл явц** – Тухайн систем буюу програм хангамжаар хийгдээгүй бодит амьдрал дээр тухайн байгууллагын зохион байгуулалтаар болон хэрэглэгчээр хийгдэж байгаа үйл явц буюу ажил хэргийн дараалал.

❖ Байгууллагын болон системийн үйл явцын ялгаа:

Системийн ажлын явц	Байгууллагын үйл явц
Бодит бус хийсвэр	Бодит амьдрал дээр үйл ажил
Програм хангамжаар хийдэг	Тоглогч болон хэрэглэгчээр хийгддэг

- ❖ **Цогц ажиллагаа (Transaction /all or nothing)** – Боловсруулалтын алхамын иж бүрдэл, тэдгээр нь бүгд гүйцэтгэгдэнэ эсвэл нэг нь ч гүйцэтгэгдэхгүй



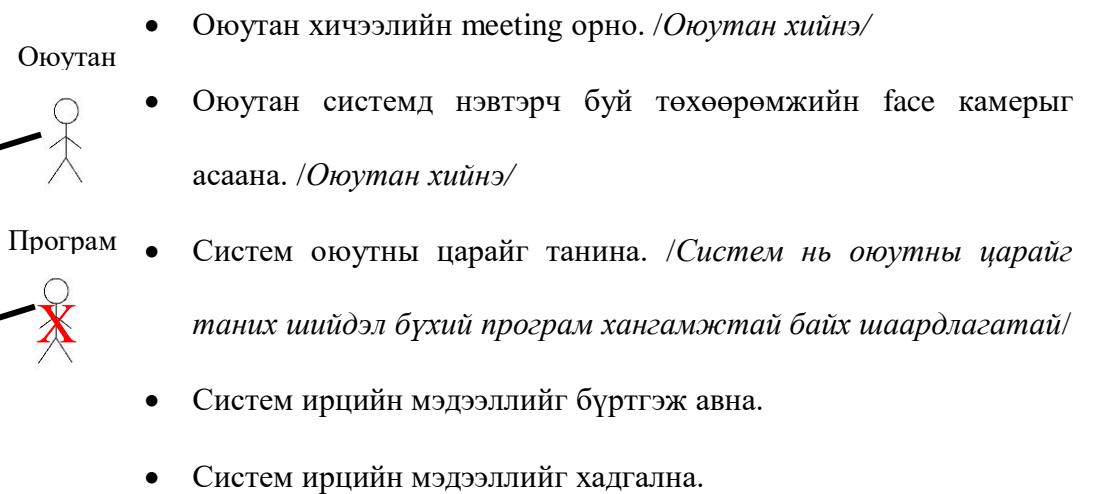
- ❖ Бүх ажлын явц нь нийтдээ систем ашиглалтын бүх боломжийг баримтжуулдаг. (use case model)
- ❖ **Use case model** – Асуудлыг шийдвэрлэхийн тулд янз бүрийн төрлийн хэрэглэгчид системтэй хэрхэн харьцаж байгааг харуулдаг загвар.

4. Шинжилгээний үеийн асуудал

- ❖ Бүх даалгаврыг програм хангамжаар гүйцэтгэх үү эсвэл хэрэглэгч шийдэл гаргах эсвэл тодорхой үйлдлийг хэрэгжүүлэхэд шаардлагатай зохион байгуулалтын алхамыг бас агуулсан байна уу гэдгийг урьдчилан хэлэхэд хэцүү байдаг.
- ❖ Жишээ: МИУС, Сиси мэдээллийн систем, Ажлын явцын диаграм загварчлах
 - Хичээлийн ирцийг бүртгэх – Ирцийн бүртгэл /Байгууллагын ажлын явц/

- Тоглогч
 - Багш
 - Оюутан
- Хийгдэх үйлдлүүд –Үйл/ ажлын явц
 - Багш оюутныг нэрээр нь дуудна.
 - Оюутан байвал байна гэж мэдэгдэнэ.
 - Багш оюутны мэдэгдлийг хүлээж авна.
 - Багш оюутанг ирсэн эсэхийг **тэмдэглэж авна.** /*систем хийнэ/*
 - Багш ирцийн **баримтыг хадгалж авна.** /*систем хийнэ/*
- Ажлын явц – Хицээлийн ирцийг бүртгэх – Ирцийн бүртгэл дараах байдлаар **системийн ажлын явц** болгож болно.

- Тоглогч
 - Оюутан
 - Шийдэл бүхий дэд систем буюу програм
- Хийгдэх үйлдлүүд –Үйл/ ажлын явц

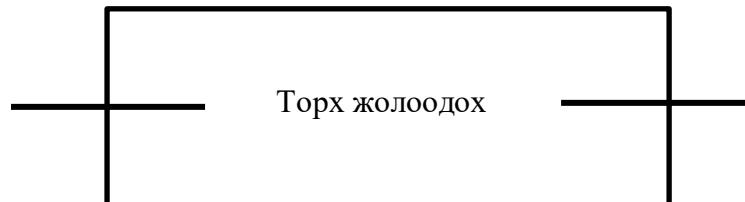


5. Тоглогч (actor)

❖ Тоглогчийн жишээ:

- Торх жолоодох /Үйл ажиллагаа/
 - Тоглогч
 - Инженер

- Исэнгэ нэрэх үйлдвэр



Исэнгэ нэрэх
үйлдвэр

6. Ажлын явцыг тайлбарлах

- ❖ Ажлын яцыг хагас формал (semiformal) эсвэл формал биш (ярианы хэлний) хэлбэрээр тайлбарлан бичдэг.

- ❖ Тус тусдаа ажиллагааны дараалал хэлбэрээр тайлбарлан бичдэг.
- ❖ Ойлгомжтой байдлыг илүү сайн болгохын тулд дугаарладаг.

❖ Ялгах

- **Стандарт тохиолдол** буюу хамгийн их тохиолддог үйл ажиллагаа
- **Өргөтгөл**

▪ Стандарт тохиолдлыг өргөтгөн авч үздэг бөгөөд эдгээр тохиолдол стандарт тохиолдлуудаас бага тохиолддог.

▪ **Өөр хувилбарын үйл явц**

▪ Стандарт тохиолдлын өөр ямар нэг байж болох хувилбрыг үздэг бөгөөд энэ нь ховор тохиолдох тохиолдлууд байдал.

- ❖ Задаргааны их идэвхжилт буюу хэв загварыг хэрэглэх
- ❖ Үргэлж хэрэглэгчийн хэсгээс хамааралгүйгээр тайлбарлан бичиж байх шаардлагатай. Учир нь хэрэглэгчийн хэсэг байнгын өөрчлөгдөж байдал.

7. Ажлын явцын задаргааны хэв (Use case template)

- ❖ **Маш олон ажлын явцын задаргааны хэвүүд байдаг.**

- **Ажлын явц – Задаргааны хэв** /Лекц дээр гарсан задаргааны хэв/

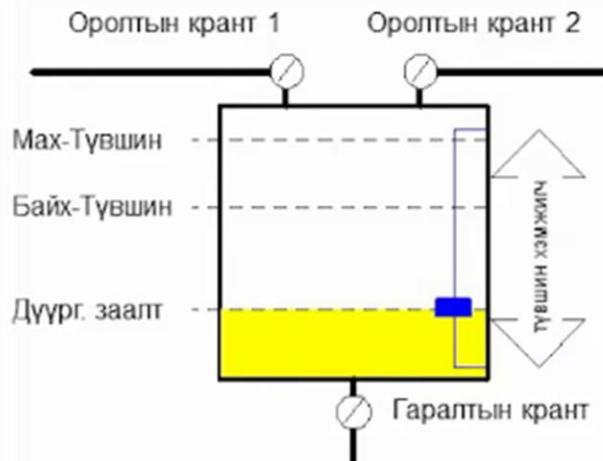
Ажлын явцын нэр:	<Ажлын явцын нэр>
Зорилго	<Ажлын явцыг амжилттай гүйцэтгэх үеийн ерөнхий зорилго>
Ангилал	<Анхдагч – шаардлагатай, байнга хэрэг болдог Хоёрдогч – шаардлагатай, хааяа л хэрэг болдог

	Сонголт – тустай, зайлшгүй шаардлагагүй.>
Угтвар нөхцөл	<Ажлын явц эхлэхийн өмнөх хүлээж байсан төлөв>
Дараах нөхцөл амжилт	<Амжилттай болсон мэдэх нөхцөл>
Дараах нөхцөл бүтэлгүйтэл	<Бүтэлгүйтлийг мэдэх нөхцөлүүд>
Тоглогч	<Ажлын явцыг гүйцэтгэх тоглогч>
Өдөөгч үзэгдэл	<Ажлын явцын эхлэлийг өдөөх үзэгдэл>
Тайлбарлалт	<Энд стандарт тохиолдлыг тайлбарлан бичдэг> <ol style="list-style-type: none"> 1. <Эхний ажиллагаа> 2. <Хоёр дахь ажиллагаа>
Өргөтгөл	1а. <Эхний ажиллагааны цар хүрээний өргөтгөл>
Хувилбар	1б. <Эхний ажиллагааны хувилбарын гүйцэтгэл>

8. Ажлын явцын жишиээ:

❖ Торх дүүргэх үйл ажиллагаа

- Эхний ээлжинд тухайн үйл ажиллагаа ямар байдлаар, хэрхэн яаж явагдах сценарь буюу **үйл ажиллагааны дарааллуудыг маш сайн тайлбарласан байх шаардлагтай.**
- Хэрэглэгчийн шаардлагын баримт бичигт ярианы хэлээр тайлбарласан:
 - Хэрэв торх хоосон бол, инженер эсвэл исэнгэ нэрэх үйлдвэр шингэний хүссэн түвшинг өгч болно.
 - Инженер эсвэл исэнгэ нэрэх үйлдвэр торхыг дүүргэх үйл ажиллагааг эхлүүлнэ.
 - Торхыг дүүргэхдээ оролтын хоёр крантыг нээх бөгөөд урьдчилан өгөгдсөн дүүргэлтийн түвшин буюу байх ёстой түвшин хүртэл шингэн торход урсан орно.



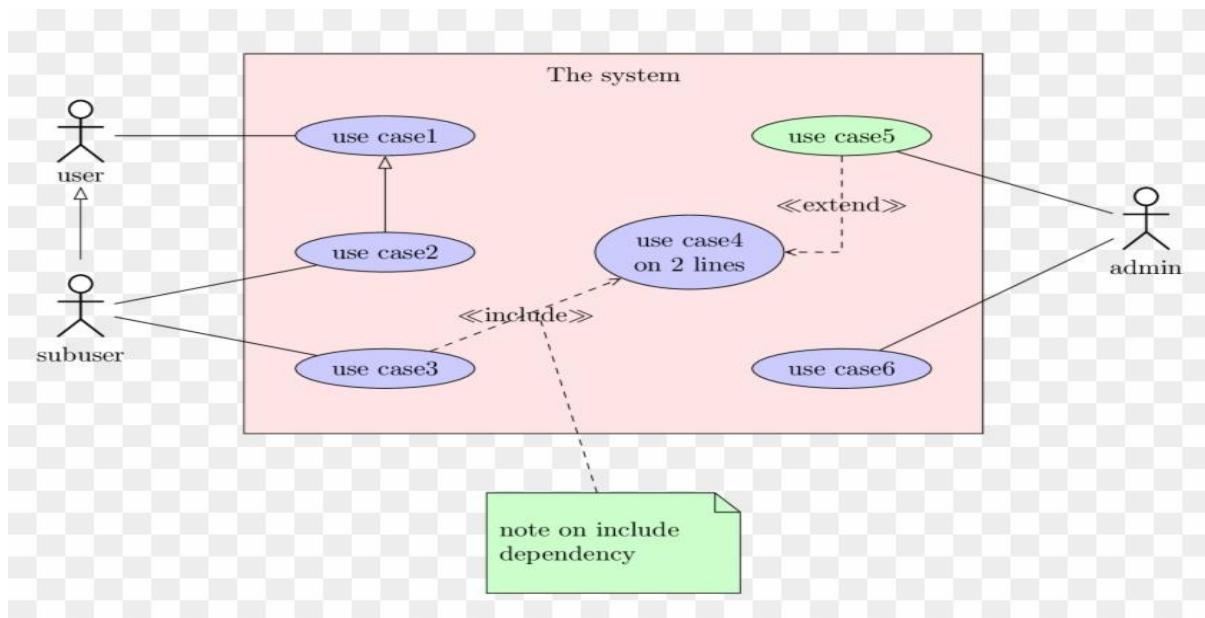
- Инженер дүүргэлтийн хамгийн их түвшинг урьдчилан өгч болно.
- Олж илрүүлсэн тоглогчид:
 - Инженер
 - Исэнгэ нэрэх үйлдвэр

Ажлын явцын нэр:	Торх дүүргэх
Зорилго	Торхыг шингэний өгөгдсөн түвшин хүртэл дүүргэх
Ангилал	Анхдагч – шаардлагатай, байнга хэрэг болдог
Угтвар нөхцөл	Торх хоосон байх
Дараах нөхцөл амжилт	Шингэний өгөгдсөн түвшин хүртэл дүүрсэн
Дараах нөхцөл бүтэлгүйтэл	Торх дүүргэлт хийгдээгүй, алдааг мэдээллэх
Тоглогч	Инженер, Исэнгэ нэрэх үйлдвэр
Өдөөгч үзэгдэл	Байх ёстой – түвшинг оруулах
Тайлбарлалт	<ol style="list-style-type: none"> 1. Инженер байх ёстой – түвшинг өгнө. 2. Инженер торх дүүргэлтийг эхлүүлнэ. 3. Оролтын крантыг нээнэ. 4. Байх ёстой – түвшинд хүрвэл, оролтын крантыг хаана.
Өргөтгөл	1а. Торх хоосон биш – алдааг мэдээллэх 1б. Байх ёстой түвшин хамгийн их түвшингээс их болох – алдааг мэдээллэх 4а. Байх ёстой түвшинд хүрэхгүй байх – алдааг мэдээллэх
Хувилбар	1а. Исэнгэ нэрэх үйлдвэр байх ёстой – түвшинг өгнө. 2а. Исэнгэ нэрэх үйлдвэр торх дүүргэлтийг эхлүүлнэ.

9. Ажлын явцын диаграм (Use case diagram)

- ❖ Тоглогчид болон ажлын явцууд хоорондын харьцааг харуулдаг.

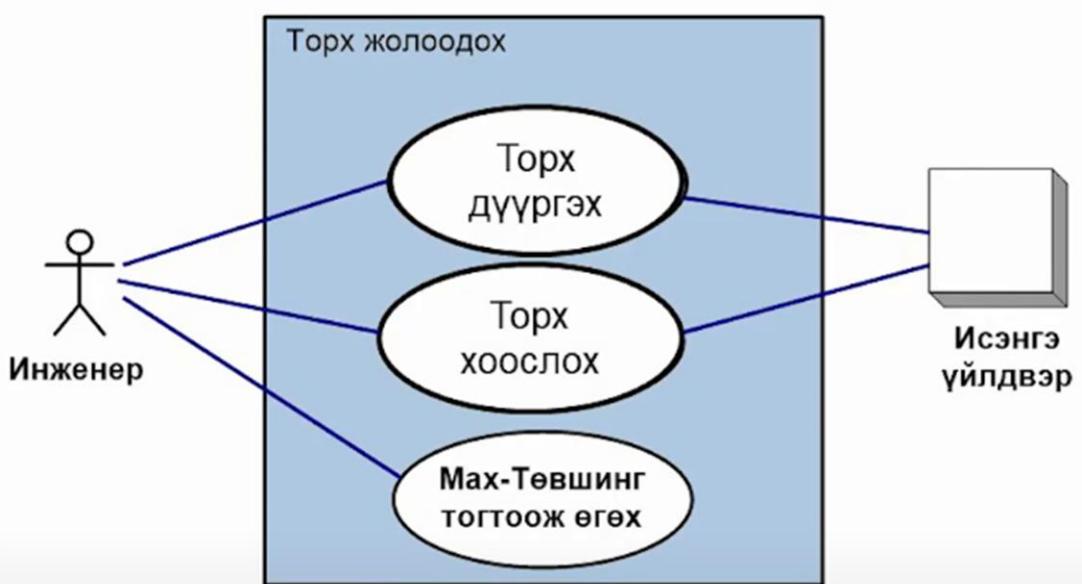
- ❖ Ажлын явцын диаграммын өрөнхий загварчлал:



- ❖ Жишээ: Торх жолоодох ажлын явцын диаграмм

- Тайлбар:

- Тоглогчид
 1. Инженер /хүн/
 - i. Торх дүүргэх үйл ажиллагааг гүйцэтгэх
 - ii. Торх хоослох үйл ажиллагааг гүйцэтгэх

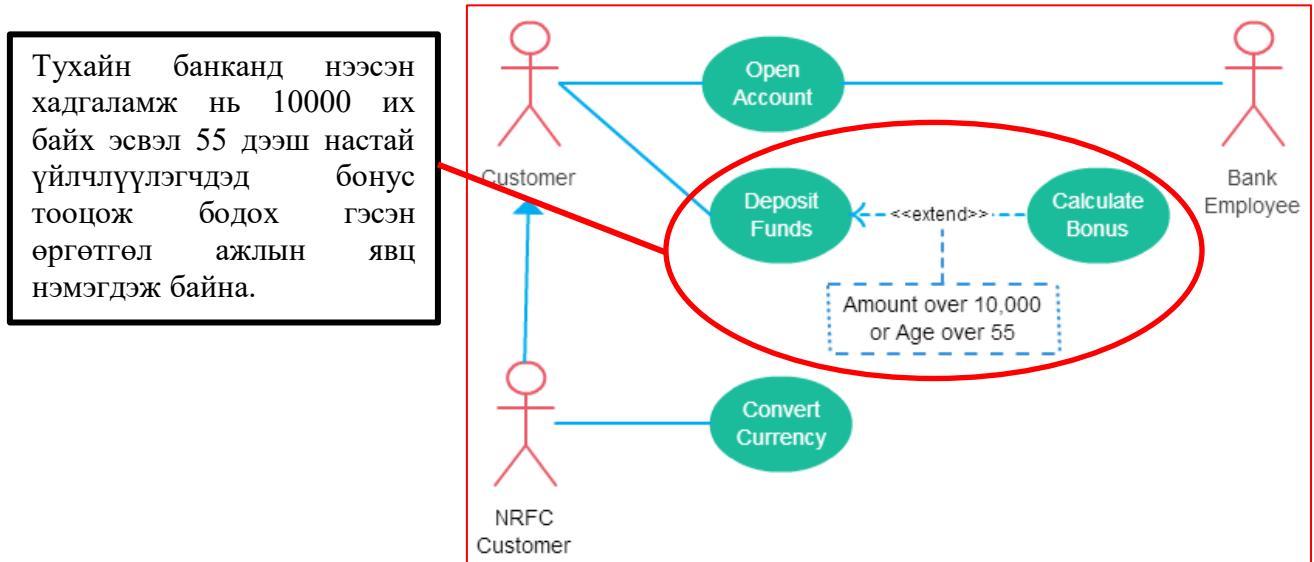


- iii. Хамгийн их буюу тах төвшинг тогтоож өгөх үйл ажиллагааг гүйцэтгэх
 - 2. Исэнгэ нэрэх үйлдвэр /систем, байгууллага/
 - i. Торх дүүргэх үйл ажиллагааг гүйцэтгэх
 - ii. Торх хоослох үйл ажиллагааг гүйцэтгэх
- Ажлын явцын диаграм дахь харьцаа

- **Харьцаа – Extend**

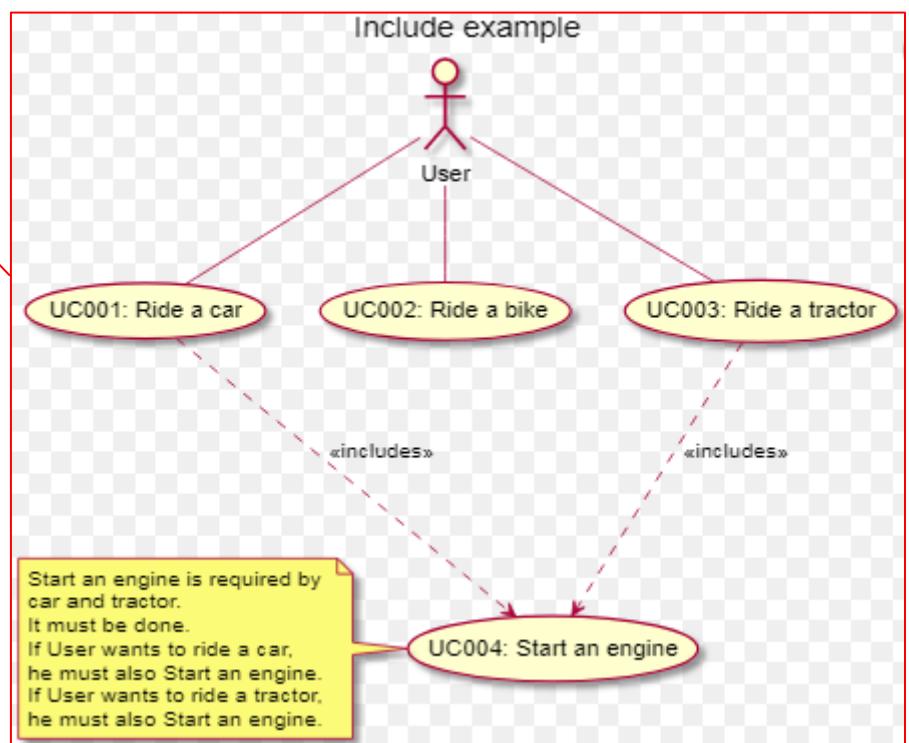
1. Байгаа ажлын явцыг өргөтгөсөн байна.

2. Жишээ:



- **Харьцаа – Include**

Уг жишээн дээр хэрэглэгч машиг унах, дугуй унах, трактор унах гэсэн 3 үйл явц байгаа бөгөөд маши унах болон трактор унах гэсэн 2 үйл явц нь хөдөлгүүр асаах гэсэн үйл явцад хамаарч байгааг харуулж байна.



- Хоёр ажлын явц өрөнхий нэг төлөв байдалтай байна. Нэг ажлын явцыг задлах эсвэл олон ажлын явцыг өрөнхийлөн авч үзэхэд ашигладаг
- Лекц № 7 : Динамик загварчлал – I /4.1. Use case – ажлын явц **Ач холбогдол:**

1. Ажлын явцын давхардсан задаргаанаас зайлсхийхэд тусалдаг.

10. Бататгах асуулт

- ❖ Дараах хэрэглэгчийн шаардлагын баримтын хэсгээс тоглогч болон ажлын явцыш олж тогтоогоод, ажлын явцын диаграмыг зур.
 - “... Улсын номын санд зориулж програм хангамжийн систем хөгжүүлэх хэрэгтэй болов. Ном олгогч түүний тусламжтайгаар ном зээлэх, буцаан авах болон захиалах боломжтой болно. Мөн түүнчлэн уншигчийг бүртгэх хэрэгтэй. Номын сангийн захиргаа шинэ ном бүртгэх болон цагаа баригүй уншигчийг сануулах хэрэгтэй.”
- **Системийн нэр:** Номын сан
- **Тоглогчид**
 - Хүн төлөөлөл: Ном олгогч
 - Ажлын явцууд:
 - Ном зээлэх
 - Ном буцаах
 - Ном захиалах
 - Уншигч бүртгэх
 - Систем төлөөлөл: Номын сангийн захиргаа
 - Ажлын явцууд:
 - Ном бүртгэх
 - Уншигчид сануулах

▪ Ажлын явцын диаграм:



11. Дүгнэлт/Бататгах асуулт болон тэмдэглэл/

- ❖ Ажлын явцын диаграмыг гаргахдаа юун түрүүнд тухайн системийн уг ажлын явц дах тоглогчдыг олж тогтоох бөгөөд тухайн тоглогчийн хийх үйл ажиллагааг дэлгэрэнгүй тайлбарлаж бүтэн өгүүлбрүүд болгох хэрэгтэй. Тоглогчийн үйл ажиллагааны мэдээллийг тухайн системийг захиалсан байгууллагын мэдээллээс болон системийн хэрэглэгчтэй харилцах замаар олж тогтоож болно. Ингэснээр системийн ажлын явцыг диаграмыг гаргахад илүү ойлгомжтой бөгөөд хялбар болдог.
- ❖ Системийн ажлын явцыг гаргахын тулд мөн байгууллагын ажлын явцыг системийн ажлын явцаас ялгахын тулд ажлын явцын диаграмыг гаргаж ашигладаг. Үүнээс гадна хэрэглэгчийн хүүрнэлээс гарах маш олон функциональ шаардлагуудыг тодорхойлоход тусалдаг.

12. Эх сурвалж

- ❖ <https://twitter.com/ivarjacobson/status/951525132119486464>
- ❖ <https://www.youtube.com/watch?app=desktop&v=qyeT9I35-o0>
- ❖ https://en.wikipedia.org/wiki/Use_case
- ❖ <https://templatelab.com/use-case-templates/>
- ❖ https://www.pngfind.com/mpng/hoomhoJ_full-size-of-wiring-diagram-note-use-case/
- ❖ [https://en.wikipedia.org/wiki/Actor_\(UML\)](https://en.wikipedia.org/wiki/Actor_(UML))
- ❖ https://www.google.com/url?sa=i&url=https%3A%2F%2Fmartinschnurer.github.io%2Fthe-difference-between-includes-and-extends-in-use-case-diagram%2F&psig=AOvVaw2vpNhF8IHN97zfHebH_o0-&ust=1616394872024000&source=images&cd=vfe&ved=0CAMQjB1qFwoTCKCIrMbiwO8CFQAAAAAdAAAAABAA

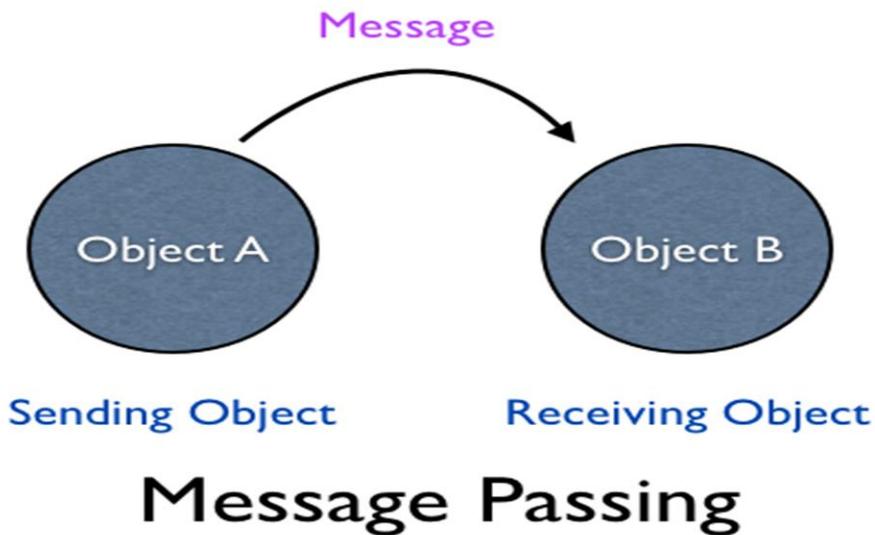
- ❖ https://www.google.com/url?sa=i&url=https%3A%2F%2Fcreately.com%2Fblog%2Fdiagrams%2Fuse-case-diagram-relationships%2F&psig=AOvVaw2-jn8ZrMIGjPB77vk_T2fa&ust=1616395531008000&source=images&cd=vfe&ved=0CA0QjhxqFwoTCNj9_eDkwO8CFQAAAAAdAAAAABAP
- ❖ http://www.utm.mx/~caff/doc/OpenUPWeb/openup/guidances/concepts/use_case_model_CD178AF9.html#:~:text=A%20use%20case%20model%20is,system%20in%20satisfying%20these%20goals.
- ❖ [Dick,-Jeremy;-Hull,-Elizabeth;-Jackson,-Ken]-Requirements Engineering
- ❖ Software-Engineering-9th-Edition-by-Ian-Sommerville

Лекц № 7 – Message /зурвас & Scenario /Сценаръ

1. Ухагдахуун – Зурвас /message (жижиг хэмжээтэй мэдээлэл)

❖ Тодорхойлолт:

- Илгээгчийн (*client*) хүссэн үйлчилгээг **хүлээн авагч** руу (*server, supplier*) явуулж буй шаардлага.



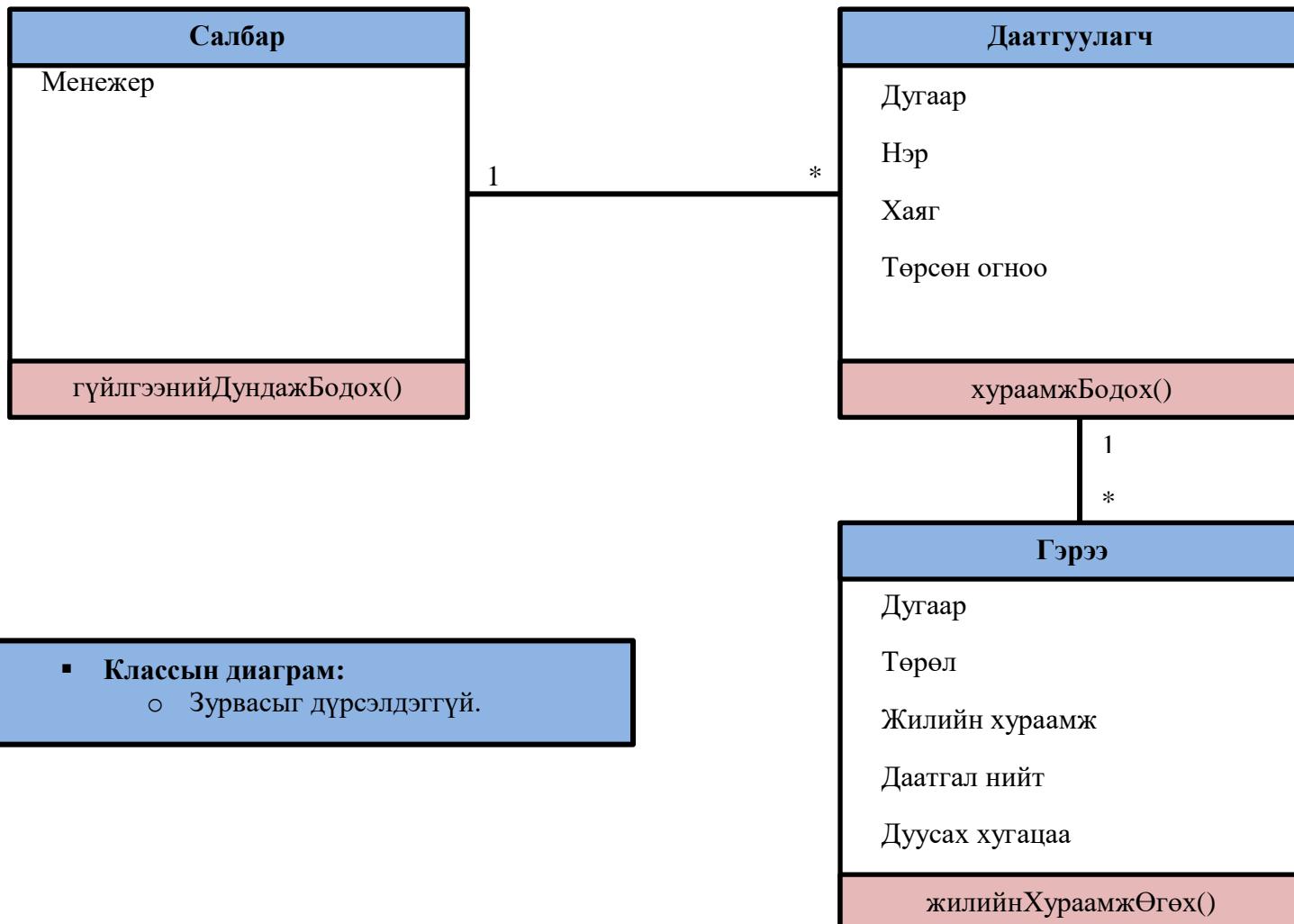
❖ Зурвасын ажиллах байдал:

- Хүлээн авагч тэр зурвасыг тайлж, уншаад дараагаар нь ижил нэртэй үйлдлийг хэрэгжүүлдэх бөгөөд зурвас илгээгч нь тэрхүү үйлдлийг яаж хэрэгжүүлэхийг мэдэхгүй бөгөөл мэдэх шаардлагагүй юм.
- Классын харьцах хэсэг /class interface гэдэг нь классын объектууд түүнд хариу үйлдэл үзүүлж чадах зурвасын иж бүрдэл.
 - Объект хандлагат програмчлал:**
 - Класс гэдэг нь хоорондоо логикийн хувьд уялдаа холбоотой хувьсагчид болон тэдгээр дээр үйлдэл хийх функциудыг агуулсан хэрэглэгчийн тодорхойлсон төрөл юм.
 - Класс төрөлтэйгээр зарлагдсан хувьсагчыг тухайн классын объект гэж нэрлэдэг.
 - Зурвас нь объектууд хоорондоо мэдээлэл солилцож, нэг нэгэндээ шаардлага тавьж, үйл ажиллагаа явуулах гол хэрэгсэл нь болж өгдөг бөгөөд мөн зурвасыг ZNH /UML(Загварчлалын Нэгдсэн Хэл) – ийн янз бүрийн диаграмын төрөлд хэрэглэдэг.
 - Объектууд нь өөр хоорондоо зөвхөн зурвас буюу жижиг бага хэмжээтэй мэдээгээр харьцдаг.

2. Зурвасын жишээ

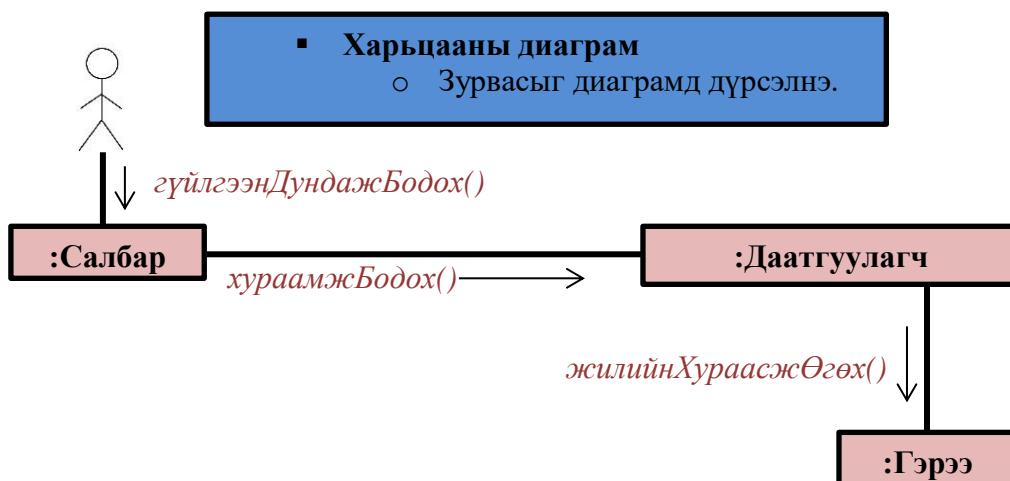
❖ Даатгалын салбар бүрийн дундаж гүйлгээг бодох:

- Даатгуулагч бүрээр хураамжийн нийлбэрийг бодохын тулд тэрхүү даатгуулагчийн гэрээ бүрийн жилийн хураамж мэдэгдсэн байх ёстой.



- Классын диаграмм:
 - Зурвасыг дүрсэлдэггүй.

❖ Хэрэв салбар нь *гүйлгээнийДундажБодох()* гэсэн зурvas авбал, дараа нь өөрийн даатгуулагч бүрд *хураамжБодох()* гэсэн зурвасыг явуулна, даатгуулагч бүр бас дахиад *жилийнХураасжӨгөх()* гэсэн зурвасыг өөрийн гэрээний бүх объект руу илгээнэ.



3. Бататгах асуулт /4.2/

❖ Аль нь зурвасын хэрэглээний тухайн зөв өгүүлбэр вэ?

- Дурын объектууд зурвасаар дамжуулан харьцдаг. – **зөв.** Учир нь *объектууд хоорондоо заавал зурвасаар харьцдаг.*
- Зурvas солилцоход объектын холбоо заавал байна, харин илгээгч болон хүлээн авагчийн хооронд холбоос байхгүй байна. – **зөв.** Учир нь *объект нь хоорондоо хамааралтай байх шаардлагатай бөгөөд хоорондоо уялдаа холбоотой байж л зурvas солилцено, харин илгээг нь хүлээн авагчтай холбоостой буюу холбоо хамааралтай байх албагүй юм.*
- **Холбоос /association** – Хоорондоо шууд холбоо хамаарал бүхий классуудыг холбоосоор холбодог.
- Зурvas нь объект диаграмын үндэс юм. – **буруу.** *Объект болон класс диаграм дээр зурвасыг дүрсэлдэггүй бөгөөд харин communication буюу харьцааны диаграм дээр дүрсэлж өгдөг.*
- Зурvas нь ижил нэртэй үйлдлийг өдөөдөг. – **зөв.** *Хүлээн авагч нь зурвасыг хүлээж авсны дараагаар ижил нэртэй үйлдлийг хэрэгжүүлдэг.*
- Классын объект нь зөвхөн классын харьцах хэсэгт тодорхойлогдсон зурваст л хариу үйлдэл үзүүлдэг. – **зөв.** *Объектэд тодорхой үйлдлүүдийг програмчилж өгсөн гэж үзвэл тэдгээр тодорхойлогдсон үйлдлүүдээр л тухайн объект хариу огөх юм.*

4. *Дүгнэлт*

- ❖ Аливаа классын объектууд нь зурвасаар дамжуулан харьцах бөгөөд зурvas солилцоход объектын холбоо заавал байна, харин илгээгч болон хүлээн авагчийн хооронд холбоос байхгүй. Мөн зурvas нь ижил нэртэй үйлдлийг өдөөдөг бөгөөд объект болон класс диаграм дээр түүнийг дүрсэлдэггүй, харин communication буюу харьцааны диаграм дээр дүрсэлж өгдөг. Классын объект нь зөвхөн классын харьцах хэсэгт буюу өөрийн цар хүрээнд тодорхойлогдсон зурваст л хариу үйлдэл үзүүлдэг байна.

5. Ухагдахуун – Scenario /сценаръ

❖ Тодорхойлолт:

- Тодорхой нөхцөлд гүйцэтгэгдэх боловсруулалтын алхамын дэс дараалал.

Мөн узэгдэл тохиоход эхлэх бөгөөд зорилгодоо хүрэх хүртэл эсвэл бүтэхгүй болж орхих хүртэл үргэлжилнэ.

❖ Тодруулбал:

▪ Боловсруулалтын алхам

- Тоглогчийн гол зорилгыг биелүүлэх ёстой бөгөөд мөн тохирсон үр дүнг гаргана.

▪ Тодорхой нөхцөл

- Сценаръ бүр онцгой ажлын явцад хүргэх нэг эсвэл олон тооны нөхцөлөөр тодорхойлогддог.

6. Ажлын явц болон Сценарь

- ❖ Ажлын явц нь олон тооны **сценариар** баримтждаг бөгөөд **ямар нэгэн зорилгод хүрэхийн тулд** юм уу эсвэл хүссэн үр дүнг бий болгохын тулд **тоглогчоор** хийгддэг, хоорондоо уялдаатай олон тооны **үүрэгт ажлаас** тогтдог.

Use case and Scenario

- **Use Case:** collection of related success and failure scenarios describing the actors attempts to support a specific goal

❖ Сценарийн ангиалал:

- **Scenario:** specific sequence of actions and interactions between actor(s) and system (= one story; success or failure)

Ажлын явцын амжилттай боловсруулалтыг тайлбарлан бичдэг.

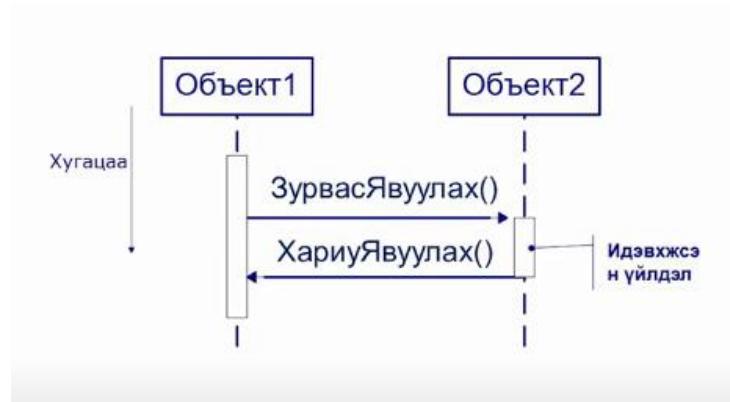
Бүтэлгүйтэлд хүргэдэг.

Сценарь

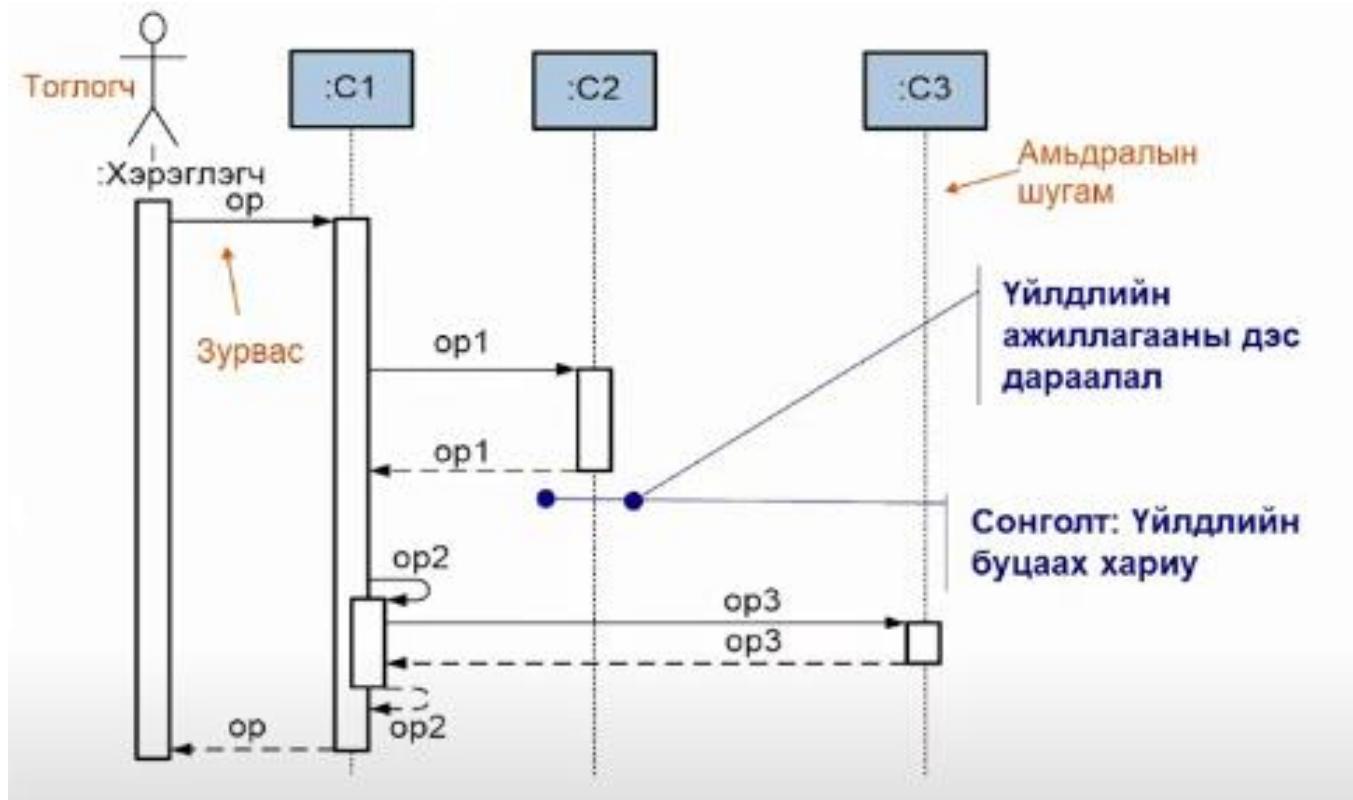
7. Сценарийн ЗНХ – ийн тэмдэглэл

- ❖ Сценарий нь харилцан үйл ажиллагааны / үйлдлийн диаграммаар(interaction diagrams) дүрслэгддэг.
- ❖ ЗНХ нь энэ төрлийн хоёр чухал диаграмм
 - Дарааллын диаграмм

- Ерөнхий загвар:



- Дарааллын диаграммын ЗНХ/UML – ийн тэмдэглэл



- Харилцааны диаграмм

- Объект болон тэдгээрийг хамтын ажиллагааг тайлбарладаг бөгөөд холбоогоор нь дамжуулан зурvas илгээдэг.

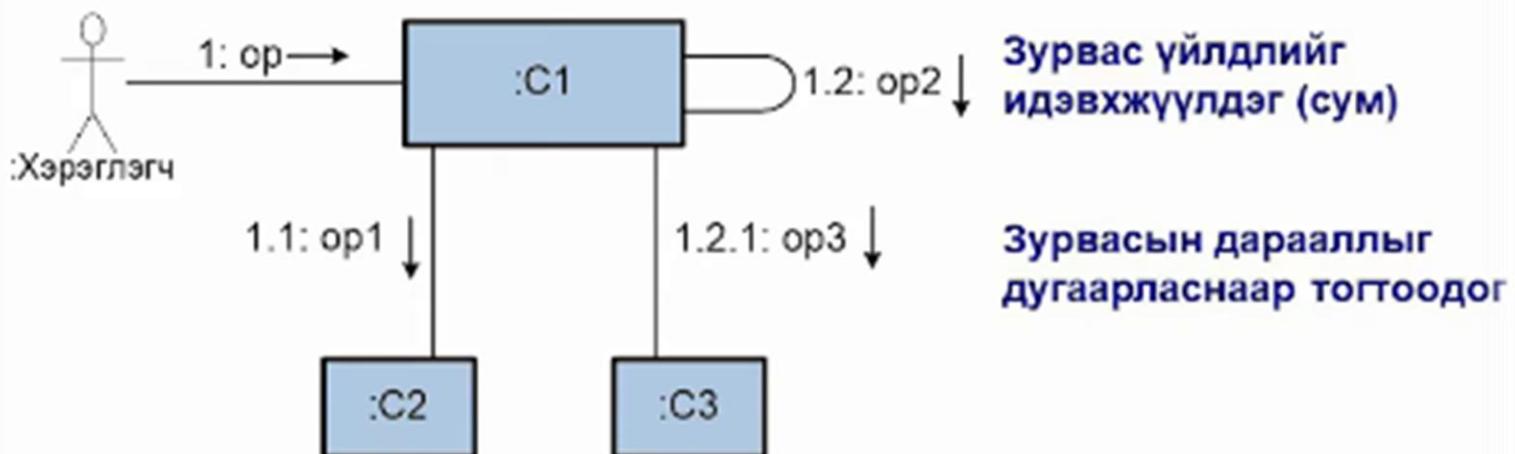
- **Объектын холбоо**

- Association буюу холбоосоор холбогдож болно.
- Объектын түр холбоогоор холбогдож болно. Байнгын холбоотой байхгүй гэсэн санааг илэрхийлнэ.
 - Харилцаа орох хугацааны туршид л зөвхөн байна.
 - Хандуулж буй хүлээн авагч объект холбоос /association байхгүй ч гэсэн илгээгчээр олж тогтоогдож чадаж байгаа бол байна.
 - Бусад холбоосоос ялгахын тулд <<temp /temporary>> stereeo төрөл буюу type – аар тэмдэглэдэг.

- Объектын өөрийн холбоо /link self – өөр өөртөө хандах

- Объект бүр дурын буюу ямар ч үед өөр өөртөө зурvas илгээж болдог.

- **Харилцааны диаграмм өрөнхий загварчлал:**

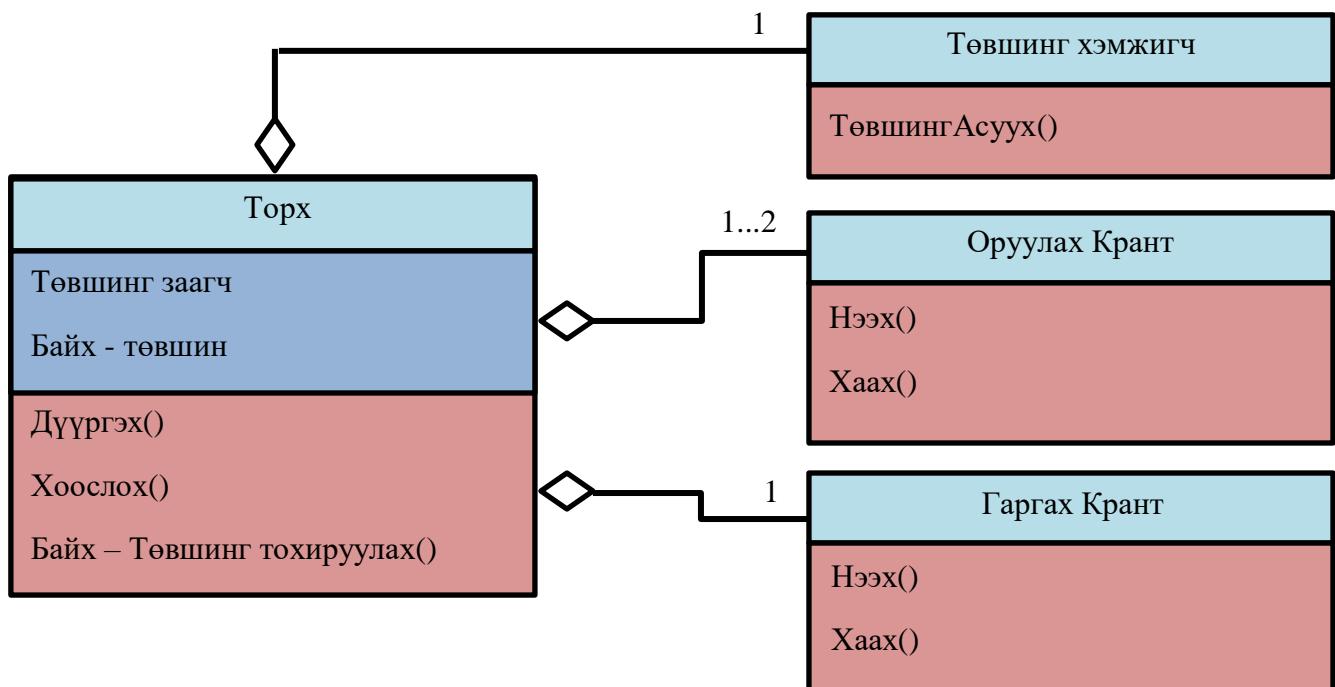


8. Класс диаграммын жишээ

- ❖ Торх дүүргбэх ажлын явцын сценаръ

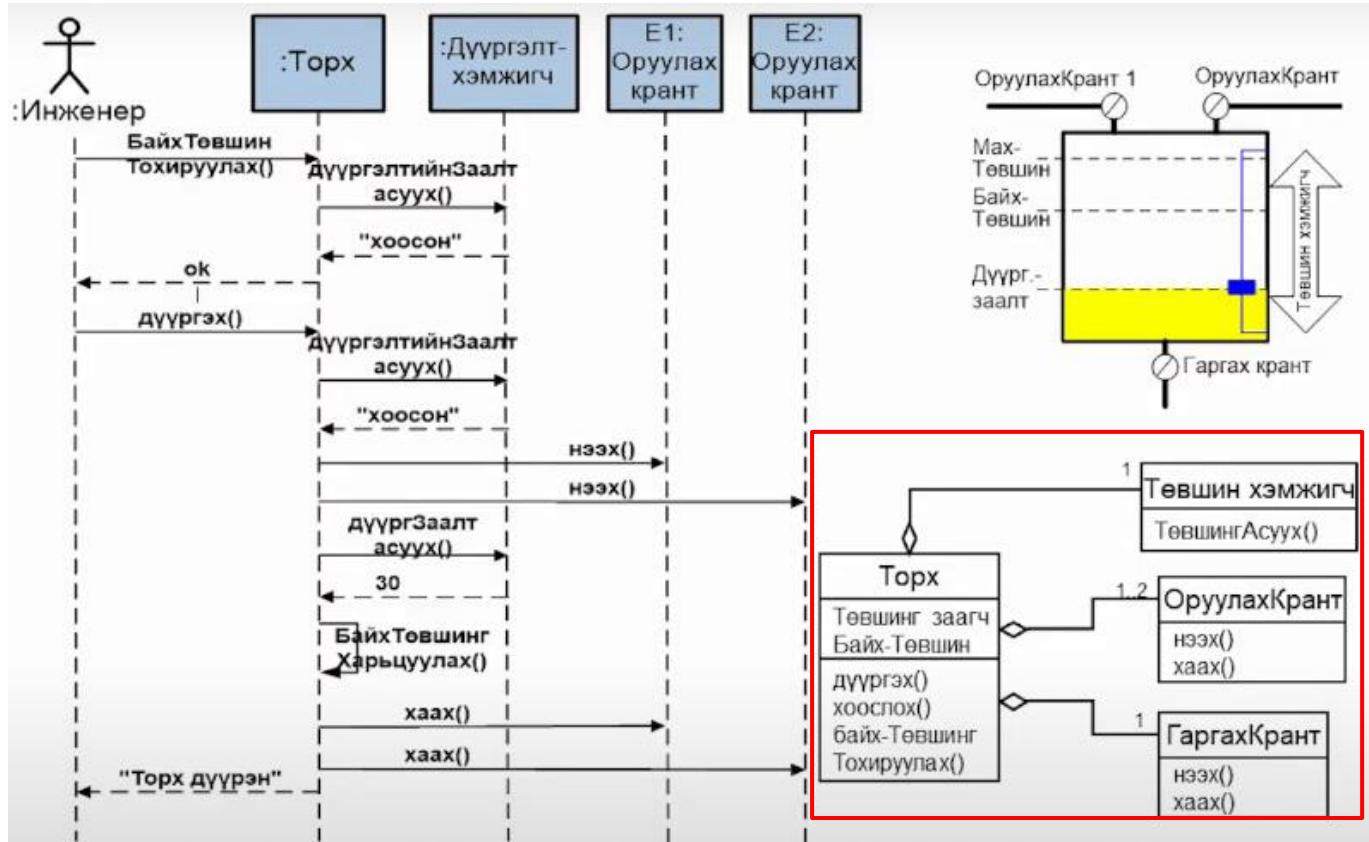
▪ Эхлэл байдал – Классын диаграмм:

- Статик төвлүйг дүрсэлдэг диаграмм бөгөөд ямар нэгэн үйл ажиллагааг харуулдаггүй зөвхөн ямар бүтэцтэйг буюу динамик бишийг харуулдаг.



9. Дарааллын диаграммын жишээ

❖ Инженер торх дүүргэх



- **Тайлбар:** Тоглогч – инженер байна.

- *Инженер* торх руу байх – ёстой төвшин тохируулах гэсэн шаардлагыг илгээнэ.
- *Торх дүүргэлтийг хэмжигч объектоос* дүүргэлтийн хэмжээг асууна.
- *Дүүргэлтийг хэмжигч объект* торх руу дүүргэлтийн хэмжээг илгээнэ.
- *Дүүргэлт* – хэмжигчээс хоосон гэсэн зурvas ирвэл *торх* нь инженер лүү дүүргэх үйл ажиллагаагаа эхэлж болно гэсэн хариуг илгээнэ.
- *Инженер* торх руу дүүргэх гэсэн шаардлагыг илгээнэ.
- *Торх дүүргэлт* – хэмжигчээс дүүргэлтийн хэмжээг асууна.
- *Дүүргэлт* – хэмжигч торх руу хэмжээг явуулна.
- *Дүүргэлт* – хэмжигчээс хоосон гэсэн хариу ирвэл *торх* крант 1 руу нээх гэсэн зурvasыг илгээнэ. Мөн крант 2 руу нээх гэсэн зурvasыг илгээнэ.
- *Торх дүүргэлт* – хэмжигчээс дүүргэлтийн төвшинг асууна.
- *Дүүргэлт* – хэмжигч төвшинг торх руу илгээнэ.
- *Торх* тухайн төвшинг байх – төвшингтэй харьцуулна. Хэрэв тэнцүү байвал крант 1, 2 руу хаах гэсэн зурvasыг илгээнэ.
- *Торх* инженер лүү торх дүүрсэн гэх зурvasыг илгээнэ.

❖ **Торх хоосон биш - Амжилтгүй сценарийн жишээ**

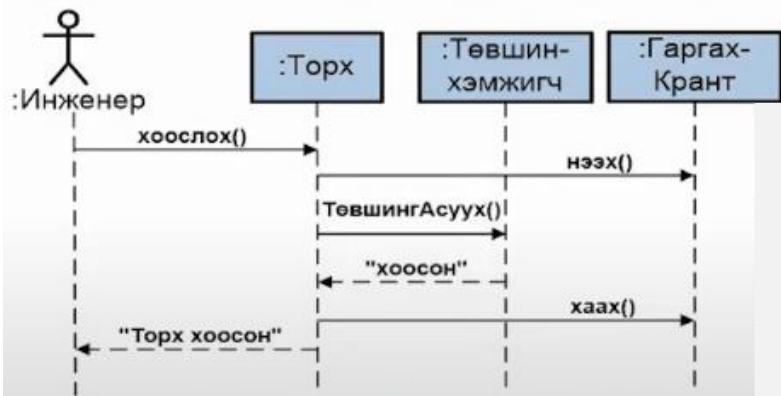


- **Тайлбар:** Тоглогч – инженер

- *Инженер* торх руу байх – төвшинг тохируулах гэсэн зурvasыг илгээнэ.
- *Torx Төвшин* – хэмжигчээс дүүргэлтийн төвшинг асууна.

- Төвшиин – хэмжигч торх дүүргэлтийн төвшинг илгээнэ.
- Хэрэв дүүргэлтийн төвшин 0 – ээс ялгаатай буюу хоосон биш байвал торх инженер лүү алдаа гэх зурvasыг илгээнэ.

❖ Торх хоослох



- Тайлбар: Тоглогч – Инженер
 - Инженер торх руу хоослох гэсэн шаардлагыг илгээнэ.
 - Торх гаргах – крант руу нээх гэсэн зурvasыг илгээнэ.
 - Торх төвшин – хэмжигчээс төвшинг асууна.
 - Төвшиин – хэмжигч төвшинг торх руу илгээнэ.
 - Хэрэв хоосон гэсэн зурvas ирвэл торх инженер лүү торх хоосон гэх зурvasыг илгээнэ.

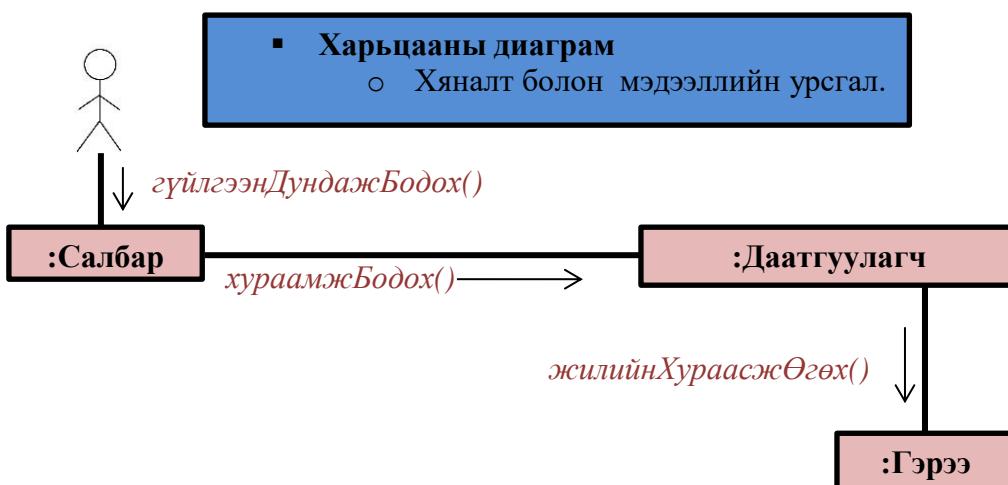
10. Дарааллын диаграм болон классын диаграм хоорондын нийцэмжийн (уялдах байдал, нэгдмэл чанар) дүрэм

- ❖ Ямар нэгэн классад харьялагдах объект руу явуулж буй зурvas нь классын диаграмд орших тэрхүү классын уйлдлийн жагсаалтад агуулагсан байх ёстой.
- ❖ Классын диаграмд зориуд загварлах шаардлагагүй ч гэсэн объект хоорондын харилцааг бүрэн тайлбарлан бичихийн тулд захиран зохицуулах /administration үйлдлийг дарааллын диаграмд нэмж оруулдаг.

11. Харилцааны болон объект диаграммын харьцуулалт

- ❖ Харилцааны диаграм:

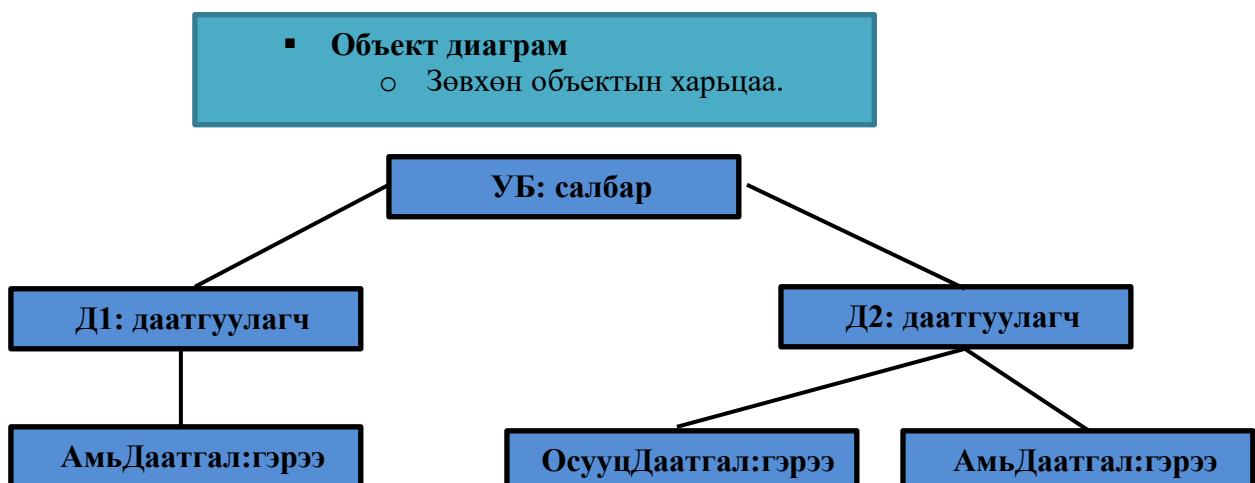
- Тухайн объект хэрхэн үйл ажиллагааны хяналтаа шилжүүлж байгааг болон зурвасаар дамжуулагдсан мэдээллийн урсгалыг харуулдаг. Өөрөөр хэлбэл



объектууд яаж мессеж буюу зурвасуудыг өөр хоорондоо солилцож байгааг харьцааны диаграммаас харж болдог.

❖ Объект диаграмм:

- Объект диаграмм тухайн үйл ажиллагааг нарийн тодорхой харуулдаггүй, харин зөвхөн объектууд хоорондоо яаж холбогдсон буюу яаж харьцаанд орж байгааг харуулдаг.



12. Харилцааны болон дарааллын диаграммын харьцуулалт

❖ Дарааллын диаграмм:

- Динамик төлөв байдлыг хугацааны талаас нь илүү тодруулсан бөгөөд ямар үйлдлүүдийг ямар дэс дарааллаар хэдий хэр хугацаанд хэрхэн хийж байгааг шууд мэдэх боломжтой байдаг. Мөн дарааллын диаграммыг цогц сценарийг тайлбарлан бичихэд ашигладаг.

❖ Харилцааны диаграм:

- Объект хоорондын холбоог илүү тодруулсан бөгөөд объектууд хоорондоо яж, ямар уялдаа холбоогоор ажиллаж байгааг шууд мэдэх боломжтой байдаг. Мөн харилцааны диаграмыг цогц үйлдлийн үр нөлөөг буюу нэг объект нь нөгөө объектдоо хэрхэн нөлөөлж байгааг тайлбарлан бичихэд ашигладаг.

❖ Харьцуулалтын дүгнэлт:

- Дарааллын диаграмыг цогц сценарийг тайлбарлан бичихэд, харин харилцааны диаграмыг цогц үйлдлийн үр нөлөөг тайлбарлан бичихэд ашигладаг учраас аль өнцгөөс харж байгаагаас хамаарч диаграмын сонголтыг хийнэ.

13. *Бататгах асуулт /4.3/*

❖ Дарааллын диаграмд зурвасын дэс дарааллыг яж тогтоосон бэ?

- **Дээрээсээ доошоо босоо чиглэлтэй шугамаар** тогтоосон.

❖ Харилцааны диаграмд зурвасын дэс дарааллыг яж тогтоосон бэ?

- **Шаталсан дугаарлалтаар** тогтоосон.

❖ Дарааллын диаграмд зурvas болон түүгээр өдөөгдсөн үйлдэл дэхь хяналтын урсгалыг хэрхэн дүрсэлдэг вэ?

- Илгээгч объектоос зурvas явуулснаар **хүлээн авагч объектол** шилждэг. Хүлээн авагч объект **уйлдлийг эхлүүлдэг**, түүний амьдрах хугацааг **нарийхан тэгш өнцөгтөөр** объектын амьдралын шугамд дүрсэлдэг. Уйлдлийн төгсгөлд сумтай тасархай шулууныг **илгээгч объект руу буцаан** заадаг. Ингэснээр хяналтын урсгал илгээгч объектын үйлдэлд буцаж ирдэг.

14. *Дүгнэлт*

- ❖ Харилцааны диаграмд зурвасын дэс дарааллыг шаталсан дугаарлалтаар тогтоодог. Харин дарааллын диаграмд зурвасын дэс дарааллыг дээрээсээ доошоо босоо чиглэлтэй шугамаар тогтоодог. Дарааллын диаграмд зурvas болон түүгээр өдөөгдсөн үйлдэл дэхь хяналтын урсгал илгээгч объектоос зурvas явуулснаар хүлээн авагч

объектод шилждэг. Дараа нь хүлээн авагч объект үйдлийг эхлүүлдэг, түүний амьдрах хугацааг нарийхан тэгш өнцөгтөөр объектын амьдралын шугамд дүрсэлж өгнө. Мөн үйлдлийн төгсгөлд сумтай тасархай шулууныг илгээгч объект руу буцаан заадаг. Ингэснээр хяналтын урсгал илгээгч объектын үйлдэлд буцаж ирдэг.

15. Эх сурвалж

- ❖ <https://www.youtube.com/watch?v=HAAHCFm-zn4>
- ❖ <https://www.youtube.com/watch?v=Phw1UbxRMOU>
- ❖ <http://itechnology1234.blogspot.com/>
- ❖ <https://www.slideshare.net/dhsemfk/6-7-27361995>
- ❖ <https://slideplayer.com/slide/5268319/>
- ❖ <https://www.slideshare.net/UnuboldTumenbayar/ss-60389201>
- ❖ [Dick,-Jeremy;-Hull,-Elizabeth;-Jackson,-Ken]-Requirements Engineering
- ❖ Software-Engineering-9th-Edition-by-Ian-Sommerville