Шинжлэх Ухаан Технологийн Их Сургууль
Мэдээлэл, Холбооны Технологийн Сургууль

Гантөмөрийн Алтай

Хувийн үүлэн хадгалалтын системийн зохиомж ба хэрэгжүүлэлт

Бакалаврын төгсөлтийн ажил

Улаанбаатар хот

Шинжлэх Ухаан Технологийн Их Сургууль
Мэдээлэл, Холбооны Технологийн Сургууль

Компьютерын ухааны салбар

Хувийн үүлэн хадгалалтын системийн зохиомж ба хэрэгжүүлэлт

Мэргэжлийн индекс: 071405000000002306
Мэргэжил: Программ хангамжийн инженерчлэл

Удирдагч: Магистр (M.Sc) Э.Батцэцэг
Зөвлөгч: Доктор (Ph.D), Дэд профессор Г.Ганбат
Гүйцэтгэгч: Г.Алтай

Улаанбаатар хот
2026 он 6 сар

Батлав. Компьютерын ухааны салбарын эрхлэгч:

..... /Доктор (Ph.D), дэд профессор А.Хүдэр/

Удирдагч:

..... /Магистр (M.Sc) Э.Батцэцэг/

ТӨГСӨЛТИЙН АЖЛЫН ТӨЛӨВЛӨГӨӨ

Сэдэв: "Хувийн үүлэн хадгалалтын системийн зохиомж ба хэрэгжүүлэлт"

№	Ажлын бүлэг, хэсгийн нэр	Эзлэх хувь	Дуусах хугацаа
1	Удиртгал хэсэг	5%	2026-2-4
2	Судалгааны хэсэг	15%	2026-2-21
3	Шинжилгээний хэсэг	15%	2026-3-4
4	Зохиомжийн хэсэг	20%	2026-4-1
5	Хөгжүүлэлтийн хэсэг	40%	2026-4-29
6	Дүгнэлт	5%	2026-5-27

Төлөвлөгөөг боловсруулсан оюутан: ..... /Г.Алтай/

ТӨГСӨЛТИЙН АЖЛЫН ЯВЦ

№	Хийж гүйцэтгэсэн ажил	Биелсэн хугацаа	Удирдагчийн гарын үсэг
1	Удиртгал	2026-2-4	
2	Судалгааны хэсэг	2026-2-21	
3	Шинжилгээний хэсэг	2026-3-4	
4	Зохиомжийн хэсэг	2026-4-1	
5	Хөгжүүлэлтийн хэсэг	2026-4-29	
6	Дүгнэлт	2026-5-27	

Ажлын товч дүгнэлт

. . . . . .

. . . . . .

. . . . . .

. . . . . .

. . . . . .

. . . . . .

. . . . . .

Удирдагч: . . . . . /Магистр (M.Sc) Э.Батцэцэг/

ЗӨВШӨӨРӨЛ

Оюутан Г.Алтай-н бичсэн төгсөлтийн ажлыг УШК-д хамгаалуулахаар тодорхойлов.

Салбарын эрхлэгч: . . . . . /Доктор (Ph.D), дэд профессор А.Хүдэр/

Шинжлэх Ухаан Технологийн Их Сургууль
Мэдээлэл, Холбооны Технологийн Сургууль

ШҮҮМЖИЙН ХУУДАС

Компьютерын ухааны салбарын төгсөх курсийн оюутан Г.Алтайгын "Хувийн үүлэн хадгалалтын системийн зохиомж ба хэрэгжүүлэлт" сэдэвт төгсөлтийн ажлын шүүмж:

1. Төслөөр дэвшүүлсэн асуудал, үүнтэй холбоотой онолын материал уншиж судалсан байдал; Энэ талаар хүмүүсийн хийсэн судалгаа, түүний үр дүнг уншиж тусгасан эсэх:

   .....
   .....
   .....
   .....
   .....
   .....
   .....

2. Төслийн ерөнхий агуулга, шийдсэн зүйлс, хүрсэн үр дүн; Өөрийн санааг гарган, харьцуулалт хийн, дүгнэж байгаа чадвар:

   .....
   .....
   .....
   .....
   .....
   .....
   .....

3. Эмх цэгцтэй, стандарт хангасан өөрөөр хэлбэл диплом бичих шаардлагуудыг биелүүлсэн эсэх; Төсөлд анзаарагдсан алдаанууд, зөв бичгийн болон өгүүлбэр зүйн гэх мэт /Хуудас дугаарлагдаагүй, зураг хүснэгтийн дугаар, тайлбар байхгүй, шрифт хольсон, хувилсан зүйл ихээр оруулсан/:

.....

.....

.....

.....

.....

.....

4. Төслөөр орхигдуулсан болон дутуу болсон зүйлс; Цаашид анхаарах хэрэгтэй зүйлс:

.....

.....

.....

.....

.....

.....

5. Төслийн талаар онцолж тэмдэглэх зүйлс:

.....

.....

.....

.....

.....

.....

6. Ерөнхий оноо (5 оноо)

.....

Шүүмж бичсэн: ..... /Магистр (M.Sc) О.Нэр/

Ажлын газар: .....

Хаяг (Утас) .....

Зохиогч эрхийн хамгаалал

Миний бие Г.Алтай, "Хувийн үүлэн хадгалалтын системийн зохиомж ба хэрэгжүүлэлт" сэдэвт энэ ажил нь минийх бөгөөд дараахыг нотолж байна. Үүнд:

- Горилогч энэ ажлыг тус сургуулиас боловсролын зэрэг авахаар бүхэлд нь буюу голлон хийсэн болно.
- Энэ ажлын аль нэг хэсгийг тус сургуульд эсвэл өөр байгууллагад боловсролын зэрэг, мэргэшил авахаар өмнө нь илгээсэн бол түүнийгээ тодорхой заасан болно.
- Бусад хүмүүсийн хэвлүүлсэн ажлаас зөвлөгөө авсан бол түүнийгээ үндэслэсэн болно.
- Бусад хүмүүсийн ажлаас ишлэл хийхдээ гол эх үүсвэрийг нь заасан болно.
- Миний ажилд тусалсан голлох бүх эх үүсвэрт талархаж байна.
- Ажлыг бусадтай хамтарсан бол алийг нь бусад хүмүүс хийсэн болохыг тодорхой заасан болно.

Гарын үсэг: _____

Огноо: _____

"*Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism.*"

Dave Barry

Шинжлэх Ухаан Технологийн Их Сургууль
Мэдээлэл, Холбооны Технологийн Сургууль

Хураангуй

Хувийн үүлэн хадгалалтын системийн зохиомж ба хэрэгжүүлэлт

Г.Алтай
bbyyydriver@gmail.com

Түлхүүр үгс: Үүлэн хадгалалт, end-to-end шифрлэлт, дамжуулагч серверийн архитектур

Энэхүү төгсөлтийн ажлын хүрээнд хэрэглэгч өөрийн өгөгдлийг бүрэн хянах боломжтой, нөөц багатай төхөөрөмж (Raspberry Pi) дээр ажиллах хувийн үүлэн хадгалалтын системийг зохиомжлон хэрэгжүүлэв. Уг систем нь гуравдагч этгээдийн хадгалалтын үйлчилгээнээс хамааралгүйгээр өгөгдлийн өмчлөл, нууцлалыг хангах зорилготой. Системийн үндсэн архитектур нь дотоод сүлжээнд серверийг автоматаар илрүүлэх болон шууд холболт боломжгүй үед дамжуулагч сервер (Relay Server) ашиглан интернэт орчинд автомат холболт тогтоох шийдлүүдээс бүрдэнэ.

Хөгжүүлэлтийн явцад олон хэрэглэгчийн зэрэг холболтыг дэмжих зэрэгцээ боловсруулалт бүхий дамжуулагч серверийн архитектурыг зохиож , хэрэглэгчийн гар утасны (IOS) программыг боловсруулсан. Аюулгүй байдлыг хангах үүднээс бүх өгөгдөл дамжуулалтад End-to-End шифрлэлт болон QR кодод суурилсан холболтын механизмыг ашигласан болно.

Талархал

Энэхүү дипломын ажлыг хийж дуусгахад хамгийн гол нөлөө үзүүлсэн, сэдэв бүрийг нягталж, үнэтэй зөвөлгөө өгч, алдааг маань засаж, удирдан чиглүүлж байсан удирдагч багш Э.Батцэцэг багшдаа хамгийн түрүүнд чин сэтгэлээсээ баярлалаа гэж хэлмээр байна. Мөн дипломын ажлын зөвлөх багш болох Г.Ганбат багшдаа үнэт цагаа зарцуулж, зааж, сургаж, орхигдуулсан зүйлсийг санууулж, хэрэгтэй зөвлөгөөнүүд өгч байсанд гүн талархлаа илэрхийлье.

Намайг бакалаврын дипломын ажил бичиж чадах хэмжээнд хүртэл зааж сургасан МХТС -н Компьютерын Ухааны салбарын багш нар, “Программ хангамж” мэргэжлийн хөтөлбөрийн хүрээнд хичээл зааж байсан ШУТИС -н багш нартаа баярлалаа. Суралцахад аюулгүй, цэвэрхэн орчныг бүрдүүлж өгч байсан МХТС -н ажилчдад баярлалаа.

Академик мэдлэгээ практик дээр туршин, дипломын ажлын хүрээнд шинэ технологиудыг судлан, мэргэжлийн хувьд туршлагажих боломж олгож байсан Грэйпсити Монгол компанийн “Технелогийн Газар” -н захирал болон ахлах инженертээ талархлаа илэрхийлмээр байна.

Эцэст нь, би дипломын ажилдаа их цаг зарцуулах боломжийг олгож, олон талаар чухал дэмжлэг үзүүлж байсан гэр бүлийнхэн болон урам зориг, эрч хүчээр урамшуулж байсан найзууддаа ч бас чин сэтгэлийн талархлаа илэрхийлэх ёстой.

Товчилсон үгс

LAN	**L**ocal **A**rea **N**etwork
E2EE	**E**nd-to-**E**nd **E**ncryption
API	**A**pplication **P**rogramming **I**nterface
CPU	**C**entral **P**rocessing **U**nit
OS	**O**perating **S**ystem
IDE	**I**ntegrated **D**evelopment **E**nvironment
SQL	**S**tructured **Q**uery **L**anguage
HTML	**H**ypertext **M**arkup **L**anguage
XML	E**x**tensible **M**arkup **L**anguage
JSON	**J**avascript **O**bject **N**otation

Гарчиг

Зургийн жагсаалт

Хүснэгтийн жагсаалт

Удиртгал

Өдгөө энгийн ухаалаг утас хэрэглэгч бүр үүлэн хадгалалт буюу Cloud Storage-ийг хэрэглэж байна. Хүн бүрт зураг, бичлэг, ажлын болон хувийн чухал файлууд гэх мэт өгөгдлүүдийг хадгалах хэрэгцээ гардаг бөгөөд түүнийгээ интернетийн тусламжтай хаанаас ч ашиглах боломж нь орчин үеийн мэдээллийн эрин үед амьдрахтай нь салшгүй нэг хэсэг болсон билээ. Гэвч энэхүү зах зээлийн ихэнх хувийг төвлөрсөн үүлэн хадгалалтын үйлчилгээ үзүүлдэг компаниуд удирдаж байгаа бөгөөд хэрэглэгчдийн өгөгдөл тухайн компанийн сервер дээр хадгалагддаг. Энэхүү хадгалалтын үйлчилгээ нь энгийнээр түрээсийн үйлчилгээ юм. Иймд хэрэглэгч өөрийн техник хангамжийг ашиглан өгөгдлөө хадгалах замаар түрээсийн төлбөр болон багтаамжийн хязгаарлалтаас ангид байх, үүний зэрэгцээ одоогийн үүлэн үйлчилгээнүүдтэй ижил түвшний хэрэглээний сэтгэл ханамжийг өгч чадах системийг хөгжүүлэх нь энэхүү ажлын гол зорилго юм.

Зорилго

Энэхүү төслийн зорилго нь хэрэглэгч өөрийн эзэмшлийн техник хангамжид суурилсан, гуравдагч этгээдээс хамааралгүй, бие даасан хувийн үүлэн хадгалалтын системийг хөгжүүлэхэд оршино. Тус систем нь өгөгдлийн өмчлөл болон нууцлалыг бүрэн хангахын зэрэгцээ дотоод болон гадаад сүлжээний орчинд төхөөрөмжүүд хооронд өгөгдлийг саадгүй, аюулгүй дамжуулах нэгдсэн шийдлийг бүрдүүлнэ.

Төслийн зорилтууд

Дэвшүүлсэн зорилгод хүрэхийн тулд дараах тодорхой зорилтуудыг хэрэгжүүлнэ. Үүнд:

1. Хувийн үүлэн хадгалалтын ижил төстэй системүүдийн архитектур болон технологийн харьцуулсан судалгаа хийх;
2. Нөөц багатай төхөөрөмж (Raspberry Pi г.м)-д зориулсан үүлэн хадгалалтын серверийн программ хангамжийг зохион бүтээх;
3. Файл удирдах үндсэн үйлдлүүд (upload, download, organize)-ийг дэмжих iOS үйлдлийн системд зориулсан хэрэглэгчийн аппликейшн боловсруулах;
4. Дотоод сүлжээнд өгөгдлийг хамгийн бага хоцрогдолтой дамжуулах ”LAN-first” механизмыг нэвтрүүлэх;

5. Интернэт орчинд олон хэрэглэгч зэрэг хандах боломжтой, зэрэгцээ боловсруулалт (parallel computing) бүхий дамжуулагч серверийн (relay server) архитектурыг шийдвэрлэх;
6. Өгөгдлийн нууцлал болон аюулгүй байдлыг хангах үүднээс төгсгөлийн цэг хоорондын (End-to-End) шифрлэлтийг хэрэгжүүлэх.

Судлах үндэслэл

Өнөөгийн дижитал шилжилтийн эрин үед үүлэн хадгалалт (Cloud Storage) нь хэрэглэгчдийн өдөр тутмын хэрэгцээ болж, 2025 оны байдлаар дэлхий даяар ашиглагчдын тоо 2.3 тэрбумыг [1] давсан үзүүлэлттэй байна. Гэвч энэхүү өсөлтийг дагаад дараах хоёр үндсэн асуудал тулгарч байна:

1. **Өгөгдлийн нууцлал ба хараат байдал:** Зах зээлийн дийлэнх хувийг Google, Apple, Microsoft зэрэг томоохон компаниуд эзэлж байгаа нь хэрэглэгчдийн хувийн мэдээлэл гуравдагч этгээдийн серверт төвлөрөхөд хүргэж байна. Энэ нь өгөгдөл эзэмших эрх (data ownership) болон нууцлалын аюулгүй байдалд эрсдэл дагуулдаг бөгөөд хэрэглэгч өөрийн датаг хянах боломжийг хязгаарладаг.
2. **Эдийн засгийн зардал ба хязгаарлагдмал нөөц:** Төвлөрсөн үүлэн үйлчилгээнүүд нь хадгалах багтаамжаас хамаарсан тогтмол (сарын эсвэл жилийн) төлбөр шаарддаг. Тухайлбал, Google One үйлчилгээний 100GB-аас 2TB хүртэлх багтаамж нь жилийн 20-100 долларын хооронд хэлбэлзэж байна. Их хэмжээний өгөгдөл хадгалах шаардлагатай хэрэглэгчдийн хувьд энэ нь урт хугацаандаа эдийн засгийн дарамт болдог.

Иймд хэрэглэгч өөрт байгаа техник хангамжийн нөөц боломжийг ашиглан, ямар нэгэн нэмэлт төлбөргүйгээр өөрийн өгөгдлийг бүрэн хянах, аюулгүй хадгалах боломжийг бүрдүүлсэн хувийн үүлэн системийг хөгжүүлэх нь технологийн болон практик ач холбогдолтой юм.

Судлагдсан байдал

Хувийн үүлэн хадгалалтын системийн чиглэлээр дэлхий дахинд Nextcloud, Seafile, Syncthing зэрэг нээлттэй эхийн төслүүд өргөнөөр судлагдаж, хөгжүүлэгдсээр ирсэн. Эдгээр системүүд нь хэрэглэгчийн өгөгдлийн нууцлалыг хангах үндсэн зорилготой боловч Raspberry Pi зэрэг нөөц багатай төхөөрөмж дээр ажиллахад системийн ачаалал ихсэх, тохиргоо хийхэд хэт нарийн мэргэжлийн мэдлэг шаарддаг зэрэг дутагдалтай талууд ажиглагддаг.

Сүлжээний хандалтын хувьд NAT traversal болон Reverse Proxy-д суурилсан шийдлүүдийг ашигладаг боловч олон хэрэглэгч зэрэг хандах үед дамжуулах хурд удаашрах асуудал тулгардаг. Иймд энэхүү төслөөр бага чадалтай төхөөрөмжид оновчтой (lightweight) ажиллах серверийн бүтэц болон зэрэгцээ боловсруулалт бүхий дамжуулагч серверийн архитектурыг ашиглан дээрх асуудлуудыг шийдвэрлэхээр зорьсон.

Шинэлэг тал

Хувийн үүлэн хадгалалтын системийн шинэлэг тал:

1. **Өгөгдлийн бүрэн эзэмшил (Data Governance)**: Хэрэглэгчийн өгөгдөл гуравдагч талын серверт бус, зөвхөн хэрэглэгчийн өөрийн техник хангамжид хадгалагдаж, нууцлал нь бүрэн хяналтад байна.
2. **Бага нөөцөд зориулсан шийдэл (Resource-Efficient)**: Raspberry Pi болон хуучин компьютер зэрэг бага чадалтай төхөөрөмжүүдэд зориулсан хөнгөн (lightweight) архитектуртай.
3. **Хялбар тохиргоо (Zero-Config Networking)**: Дамжуулагч серверийн тусламжтайгаар сүлжээний нарийн тохиргоо (Port forwarding, Static IP) шаардахгүйгээр шууд ашиглах боломжтой.

Технологийн ач холбогдол

Энэхүү систем нь технологийн хувьд дараах ач холбогдолтой:

1. **Сүлжээний бие даасан байдал**: Дамжуулагч серверийн шийдэл нь интернэтийн дурын орчноос (Public IP-гүй байсан ч) саадгүй холбогдох боломжийг олгоно.
2. **Зардал хэмнэлт**: Үүлэн хадгалалтын тогтмол төлбөрийг халж, байгаа нөөцөө ашиглан эдийн засгийн хэмнэлт гаргана.
3. **Аюулгүй байдлын шинэ стандарт**: End-to-End шифрлэлтийг ашигласнаар дамжуулагч сервер хүртэл өгөгдлийг унших боломжгүй болж, нууцлалын өндөр түвшинд хүрнэ.

Хамрах хүрээ

Тус төсөл нь дараах хүрээг хамарна:

- Өөрийн өгөгдлийн нууцлалыг эрхэмлэдэг хувь хүн болон гэр бүлийн хэрэглэгчид;
- Өндөр өртөг бүхий үүлэн үйлчилгээнээс татгалзаж, өөрийн нөөцийг ашиглах хүсэлтэй технологи сонирхогчид;

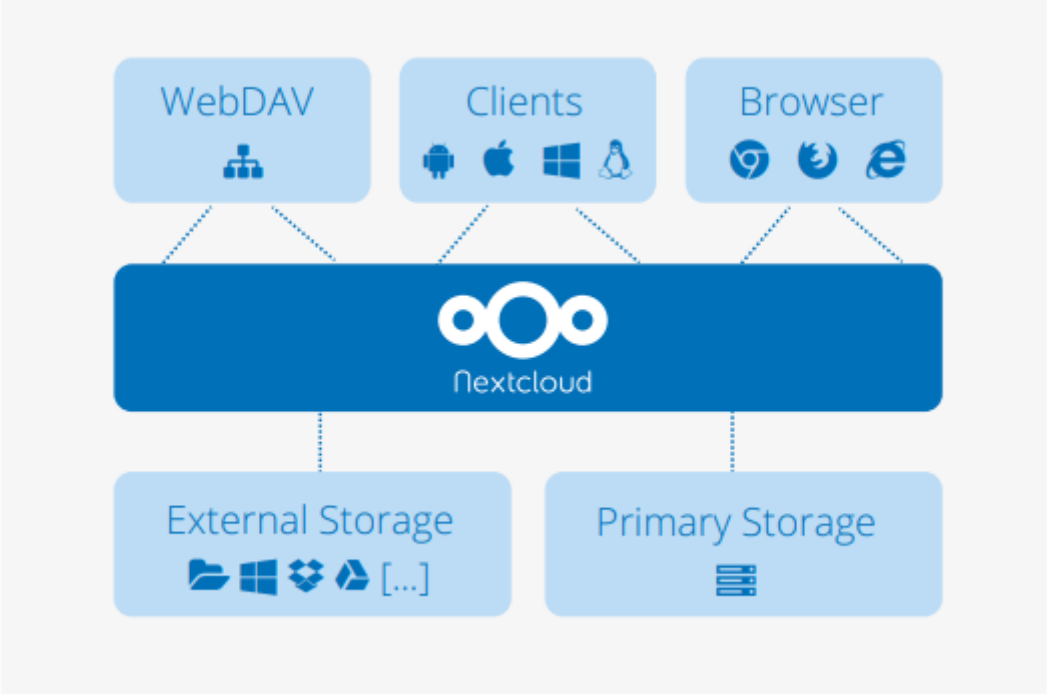
БҮЛЭГ 1

Хувийн үүлэн хадгалалтын системийн тухай онол, арга зүйн судалгаа

1.1 Ижил төст системийн судалгаа

Хувийн өгөгдөл хадгалах, синхрончлох системүүд нь сүүлийн жилүүдэд хэрэглэгчийн өгөгдөлд бүрэн хяналттай байх, аюулгүй байдал болон cloud-ын энгийн хэрэглээний туршлагыг хадгалах чиглэлд эрчимтэй хөгжиж байна. Энэ хүрээнд уламжлалт төвлөрсөн cloud системүүд болон peer-to-peer системүүдийг судалгаанд оруулж байна.

Nextcloud нь self-hosted cloud storage системүүдийн түгээмэл жишээ бөгөөд сервер төвтэй архитектур ашиглан файл хадгалалт, синхрончлол, хуваалцах болон хамтын ажиллагааны олон төрлийн боломжуудыг нэг дор нэгтгэдэг. Nextcloud нь Universal File Access [2] давхаргаар дамжуулан төрөл бүрийн storage эх үүсвэрүүдийг нэг интерфэйс дор нэгтгэх боломж олгодог. Гэсэн хэдий ч ийм өргөн хүрээтэй боломжууд нь системийг суулгах, тохируулах, удирдах процессыг төвөгтэй болгож, энгийн хэрэглэгчдэд хүндрэл учруулдаг.



Зураг 1.1: NextCloud системийн өгөгдлийн давхаргын нэгдсэн шийдэл

Нөгөө талаас Syncthing нь peer-to-peer зарчимд суурилсан файл синхрончлолын систем бөгөөд төв сервер шаардалгүйгээр төхөөрөмжүүдийг хооронд нь шууд холбодог. Syncthing нь Local Discovery, Global Discovery, Relay Protocol зэрэг механизмуудыг [3] ашиглан сүлжээний янз бүрийн орчинд төхөөрөмжүүдийг илрүүлэх, холбох

боломжийг олгодог. Мөн бүх өгөгдөл end-to-end encryption (E2EE)-ээр1.2 хамгаалагддаг нь аюулгүй байдлын чухал давуу тал юм. Гэвч Syncthing нь олон төхөөрөмж дээр нэг фолдерийг хуулбарлан хадгалах (replication) зарчимд суурилсан тул уламжлалт cloud storage-ийн өгөгдөл нэг газарт төвлөрсөн байдлаар хадгалагдана гэсэн ойлголттой нийцдэггүй.

Мөн хэрэглэгчийн хүлээлтийг тодорхойлогч consumer cloud storage системүүд болох Apple iCloud болон Google Drive нь автомат backup, background upload, минимал тохиргоо зэрэг UX-ийн өндөр стандартыг тогтоосон. Эдгээр системүүд нь хэрэглээний хувьд энгийн боловч өгөгдөл нь гуравдагч талын төв серверт хадгалагддаг.

1.1.1 Нийтийн үүлэн хадгалалтын системүүд

Нийтийн үүлэн хадгалалтын системүүдэд дараах өргөн хэрэглэгддэг платформууд орно:

1. Apple iCloud
2. Google Drive
3. Microsoft OneDrive

Нийтийн үүлэн хадгалалтын системүүд нь автомат синхрончлол, нөөцлөлт зэрэг үйлдлүүдийг хэрэглэгчийн оролцоогүйгээр гүйцэтгэх замаар өндөр түвшний хэрэглээний энгийн байдлыг хангадаг. Гэсэн хэдий ч өгөгдөл нь гуравдагч талын төв серверт хадгалагддаг тул хэрэглэгчийн өгөгдөлд бүрэн хяналттай байх боломж хязгаарлагддаг.

Зураг 1.2: Өргөн ашиглагдаж буй үүлэн хадгалалтын үйлчилгээнүүд

1.1.2 Хувийн үүлэн хадгалалтын системүүд

Self-hosted cloud системүүдийн түгээмэл жишээнүүд:

1. NextCloud (opensource)
2. OwnCloud (opensource)
3. Seafile (opensource)

Self-hosted cloud системүүд нь уламжлалт сервер төвтэй архитектур дээр суурилан ажилладаг бөгөөд хэрэглэгчийн өгөгдлийг өөрийн хяналт дор хадгалах боломжийг олгодог. Эдгээр системүүд нь хэрэглэгчийн интерфэйсийн хувьд харьцангуй боловсронгуй боловч сүлжээний орчин, серверийн тохиргоо, удирдлагын нэмэлт мэдлэг шаарддаг. Иймээс уг төрлийн шийдлүүд нь байгууллагын түвшинд хамтран ашиглахад илүү тохиромжтой бөгөөд хувийн хэрэглээнд ашиглахад техникийн хүндрэл үүсгэх магадлалтай.

Зураг 1.3: Хувийн сервер дээр ашиглах боломжтой үүлэн хадгалалтын платформууд

1.1.3 Peer-to-Peer өгөгдөл синхрончлолын системүүд

Peer-to-peer (P2P) зарчимд суурилсан өгөгдөл синхрончлолын системүүд:

1. Syncthing (opensource)
2. Resilio Sync

Эдгээр системүүд нь төхөөрөмж бүрийг давтагдашгүй таних тэмдэг (device identity)-ээр тодорхойлж, төв сервер шаардахгүйгээр хооронд нь шууд холбох боломжийг олгодог. Төхөөрөмжүүдийг илрүүлэхдээ дотоод сүлжээнд LAN discovery механизмыг ашиглах бөгөөд шаардлагатай тохиолдолд relay fallback [3] шийдлийг ашиглан холболтыг үргэлжлүүлэх боломжтой. Мөн өгөгдөл дамжуулалт нь end-to-end encryption (E2EE)-д суурилсан zero-trust загвараар хамгаалагддаг нь аюулгүй байдлын чухал давуу тал болдог.

Гэсэн хэдий ч эдгээр системүүд нь төхөөрөмжүүдийн хооронд файлыг харилцан хуулбарлан хадгалах (replication) зарчимд суурилдаг тул өгөгдөл нэг төв байршилд хадгалагдах уламжлалт хувийн cloud storage загвараас ялгаатай. Иймээс тэдгээрийн хэрэглээний зорилго нь бүрэн нийцэхгүй боловч дотооддоо ашиглаж буй холболт, илрүүлэлт болон аюулгүй дамжуулалтын технологиуд нь судлан хэрэгжүүлэхэд чухал ач холбогдолтой юм.

1.1.4 Харьцуулсан шинжилгээ

Дараах хүснэгтээс харахад одоо байгаа шийдлүүд нь хэрэглээний энгийн байдал, өгөгдлийн хяналт болон сүлжээний уян хатан байдлын хооронд тодорхой зөрчил (trade-off) үүсгэж байна. Нийтийн үүлэн системүүд нь хэрэглээний хувьд энгийн боловч хэрэглэгчийн өгөгдөлд бүрэн хяналт олгодоггүй. Харин self-hosted болон P2P системүүд нь өгөгдлийн хяналт болон аюулгүй байдлыг сайжруулдаг ч суурилуулалт болон ашиглалтын төвөгшил нэмэгддэг. Иймээс эдгээр системүүд нь хэрэглээний энгийн байдал, аюулгүй байдал болон сүлжээний уян хатан байдлыг зэрэг хангах нэгдсэн шийдлийг бүрэн санал болгож чадахгүй байна.

Систем	**Архитектур**	**LAN Auto Discovery** 1.3	**Relay Fallback**	**E2EE** 1.2	**UX энгийн байдал**
Nextcloud	Төвлөрсөн self-hosted сервер	Байхгүй	Байхгүй	Хэсэгчлэн	Дунд
Syncthing	Peer-to-Peer (P2P)	Байгаа	Байгаа	Байгаа	Харьцангуй төвөгтэй
iCloud / Google Drive	Төвлөрсөн нийтийн cloud	Байхгүй	Байхгүй	Хязгаар -лагдмал	Өндөр
Санал болгож буй систем	Hybrid (Server + Relay)	Байгаа	Байгаа	Байгаа	Өндөр

Хүснэгт 1.1: Ижил төст системүүдийн харьцуулалт

1.2 End-to-End шифрлэлт (E2EE)

End-to-End шифрлэлт гэдэг нь нэг цэгээс нөгөө цэгд өгөгдлийг дамжуулахдаа шифрлэж илгээх процессийг хэлдэг. Өгөгдөл нь илгээгдэх үедээ шифрлэлттэй илгээгдэж хүлээн авагч нь л тайлж уншдаг. Мессеж чат бичих аппууд болон бусад харилцаа

холбооны үйлчилгээнүүд E2EE-г ашиглаж мессежүүдийг зөвшөөрөлгүй хандалтаас сэргийлдэг. сүлжээгээр харилцах хамгийн аюулгүй арга гэж тооцогддог. [4]

E2EE нь бусад шифрлэлтийн аргуудаас ялгаатай нь өгөгдлийг эхлэлээс төгсгөл хүртэл хамгаалдгаараа онцлог юм. Энэ нь өгөгдлийг илгээгчийн төхөөрөмж дээр шифрлэж, дамжуулалтын турш шифрлэгдсэн хэвээр хадгалан, зөвхөн хүлээн авагчийн төхөөрөмж дээр тайлдаг. Ийнхүү дамжуулалтыг зуучилж буй үйлчилгээ үзүүлэгчид, жишээлбэл WhatsApp зэрэг системүүд, дамжуулж буй мэдээллийн агуулгад нэвтрэх боломжгүй болдог. Өөрөөр хэлбэл, зөвхөн илгээгч болон зорилтот хүлээн авагч л мэдээллийг унших боломжтой байдаг.

Үүнтэй харьцуулахад дамжуулалтын үеийн шифрлэлт (encryption in transit) нь өгөгдлийг зөвхөн дамжиж буй хугацаанд хамгаалдаг. Жишээлбэл, Transport Layer Security (TLS) протокол нь өгөгдлийг клиент болон серверийн хооронд дамжих үед шифрлэдэг. Гэвч энэ нь өгөгдлийг дамжуулалтын дараах шатанд хамгаалахгүй бөгөөд програмын серверүүд эсвэл сүлжээний үйлчилгээ үзүүлэгчид зэрэг завсрын оролцогчид өгөгдөлд нэвтрэх боломжтой хэвээр байдаг.

1.2.1 E2EE хэрхэн ажилладаг вэ?

End-to-End шифрлэлтийн процесс нь уншигдах боломжтой өгөгдлийг уншигдах боломжгүй хэлбэрт хувиргаж, аюулгүй байдлаар дамжуулан, хүрэх цэг дээр нь дахин анхны хэлбэрт нь сэргээх үйл явцыг агуулдаг.

Нарийвчилбал E2EE дараах үндсэн 4 хэсгийг агуулдаг: [4]

1. Шифрлэлт
2. Дамжуулалт
3. Өгөгдлийг тайлах
4. Баталгаажуулалт

Шифрлэлт

E2EE нь нууц өгөгдлийг шифрлэх алгоритмыг ашигласнаар эхэлдэг. Энэхүү алгоритм нь нарийн төвөгтэй математик функцуудыг ашиглан өгөгдлийг уншигдах боломжгүй хэлбэрт хувиргадаг бөгөөд үүнийг шифрлэгдсэн текст (ciphertext) гэж нэрлэдэг. Зөвхөн тайлах түлхүүр (decryption key) бүхий эрх бүхий хэрэглэгчид л уг мэдээллийг унших боломжтой байдаг.

E2EE нь өгөгдлийг шифрлэх болон тайлахад хоёр өөр түлхүүр ашигладаг тэгш хэмт бус (asymmetric) шифрлэлтийн схемийг, эсвэл нэг ижил нууц түлхүүр ашигладаг тэгш хэмт (symmetric) шифрлэлтийн схемийг ашиглаж болно. Ихэнх E2EE хэрэгжилтүүд нь эдгээр хоёр аргыг хослуулан ашигладаг. 1.2.2 хэсэгт дэлгэрүүлж бичсэн байгаа.

Дамжуулалт

Шифрлэгдсэн өгөгдөл (ciphertext) нь интернет болон бусад сүлжээ зэрэг харилцаа холбооны сувгаар дамжин зорьсон газартаа хүрдэг. Дамжуулалтын явцад уг мэдээлэл нь програмын серверүүд, интернет үйлчилгээ үзүүлэгчид (ISP), халдагчид болон бусад этгээдүүдэд уншигдах боломжгүй хэвээр байна. Хэрэв дамжуулалтын үеэр хэн нэгэн мэдээллийг барьж авсан тохиолдолд энэ нь санамсаргүй, ойлгомжгүй тэмдэгтүүдийн дараалал мэт харагдах болно.

Өгөгдлийг тайлах

Шифрлэгдсэн өгөгдөл хүлээн авагчийн төхөөрөмжид хүрэхэд асимметрик шифрлэлтийн үед хүлээн авагчийн хувийн түлхүүр, харин симметрик шифрлэлтийн үед урьдчилан хуваалцсан нууц түлхүүрийг ашиглан тайлагдана. Өгөгдлийг тайлахад шаардлагатай хувийн түлхүүрийг зөвхөн хүлээн авагч эзэмшдэг.

Баталгаажуулалт

Тайлсан өгөгдлийн бүрэн бүтэн байдал болон жинхэнэ эх сурвалжийг баталгаажуулах зорилгоор шалгалт хийгддэг. Энэ шатанд илгээгчийн дижитал гарын үсэг эсвэл бусад баталгаажуулах мэдээллийг шалгаж, дамжуулалтын явцад өгөгдөлд ямар нэгэн өөрчлөлт орсон эсэхийг тодорхойлно.

1.2.2 Шифрлэлтийн алгоритм ба тэгш хэмт ба тэгш хэмт бус шифрлэлт (Symmetric vs asymmetric)

Шифрлэлтийн үндсэн хоёр арга болох тэгш хэмт (symmetric) болон тэгш хэмт бус (asymmetric) шифрлэлт нь нууц түлхүүрийг ашиглах зарчмаараа ялгаатай. Тэгш хэмт шифрлэлт нь өгөгдлийг шифрлэх болон тайлахад нэг ижил нууц түлхүүр ашигладаг бөгөөд хурд болон үр ашгийн хувьд давуу талтай боловч түлхүүрийн аюулгүй удирдлагыг шаарддаг. Харин тэгш хэмт бус шифрлэлт нь хоёр өөр криптограф түлхүүр болох нийтийн түлхүүр (public key) болон хувийн түлхүүр (private key)-ийг

ашигладаг бөгөөд түлхүүрийг аюулгүйгээр хуваалцах асуудлыг шийдвэрлэдэг боловч боловсруулах хурд харьцангуй удаан байдаг.

- **Тэгш хэмт шифрлэлт (Symmetric Encryption):** Өгөгдөл M-ийг K түлхүүрээр шифрлэх (E) ба тайлах (D) процесс нь ижил түлхүүр ашиглана:

$$C = E_K(M), \quad M = D_K(C) \tag{1.1}$$

- **Тэгш хэмт бус шифрлэлт (Asymmetric Encryption):** Илгээгч нь хүлээн авагчийн нийтийн түлхүүр K_{pub}-аар шифрлэх ба зөвхөн хүлээн авагч өөрийн хувийн түлхүүр K_{priv}-ээр тайлна:

$$C = E_{K_{pub}}(M), \quad M = D_{K_{priv}}(C) \tag{1.2}$$

Дээрх онолын аргуудыг бодит системд хэрэгжүүлэхдээ гүйцэтгэл болон аюулгүй байдлын тэнцвэрийг хадгалах дараах алгоритмуудыг ашиглах нь тохиромжтой байна:

1. **AES-GCM (Advanced Encryption Standard - Galois/Counter Mode):** *Төрөл: Тэгш хэмт (Symmetric).* Өнөөдрийн байдлаар дэлхийн хамгийн найдвартай стандарт юм. Гүйцэтгэлийн хувьд Raspberry Pi 4 болон орчин үеийн ухаалаг утасны процессорууд дахь техник хангамжийн хурдасгуурыг (hardware acceleration) ашигладаг тул маш хурдан ажилладаг. Мөн өгөгдлийг шифрлэхийн зэрэгцээ түүний бүрэн бүтэн байдлыг (integrity) шалгах AEAD горимыг дэмждэг.
2. **ChaCha20-Poly1305:** *Төрөл: Тэгш хэмт (Symmetric).* Google-ийн санал болгосон алгоритм бөгөөд техник хангамжийн хурдасгуургүй бага хүчин чадалтай төхөөрөмжүүд дээр AES-ээс илүү хурдан ажилладаг. Программ хангамжийн түвшинд хэрэгжүүлэхэд хялбар, аюулгүй байдлын хувьд AES-тэй ижил түвшинд үнэлэгддэг тул мобайл аппликейшнд өргөн ашигладаг.
3. **X25519 (Curve25519):** *Төрөл: Тэгш хэмт бус (Asymmetric / Key Exchange).* Төхөөрөмжүүд хоорондоо нууц түлхүүрээ аюулгүй солилцоход ашигладаг Elliptic Curve Diffie-Hellman (ECDH) протоколын хамгийн түгээмэл хувилбар юм. RSA-тай харьцуулахад түлхүүрийн хэмжээ маш бага (256-бит) боловч аюулгүй байдал нь өндөр тул сүлжээний зурвасын өргөнийг хэмнэдэг.

Иймээс End-to-End Encryption (E2EE)-ийг хэрэгжүүлдэг системүүд ихэвчлэн эдгээр хоёр аргыг хослуулан ашигладаг. Жишээлбэл, хэрэглэгчид хооронд харилцаа

эхлэх үед тухайн session-д зориулсан өвөрмөц session түлхүүр үүсгэж, мессежүүдийг symmetric аргаар шифрлэдэг. Энэхүү session түлхүүрийг asymmetric шифрлэлтийн аргаар дамжуулан хуваалцдаг бөгөөд хүлээн авагчийн нийтийн түлхүүрээр шифрлэгдэж, зөвхөн түүний хувийн түлхүүрээр тайлагддаг. Ингэснээр систем нь asymmetric шифрлэлтийн аюулгүй байдлыг symmetric шифрлэлтийн өндөр үр ашигтай хослуулсан хамгаалалтын механизмыг бүрдүүлдэг.

1.2.3 E2EE-ийн нийтлэг хэрэглээний тохиолдлууд

E2EE-ийн нийтлэг хэрэглээнд дараах хувийн эмзэг мэдээлэл хадгалах хэрэглээнүүд ордог:

- **Найдвартай харилцаа холбоо:** Apple iMessage, WhatsApp аппууд нь E2EE ашиглаж хэрэглэгчдийн хооронд мессеж илгээдэг.
- **Нууц үгийн зохицуулга:** 1Password, Bitwarden, Dashlane, LastPass гэх мэт хэрэглэгчийн нууц үг зохицуулгын үйлчилгээнүүд хэрэглэгчийн төхөөрөмж хооронд E2EE ашиглан нууц үгийг илгээдэг.
- **Файл хуваалцах:** E2EE нь өгөгдлийг дамжуулах явцад зөвшөөрөлгүй этгээдүүд нэвтрэхээс хамгаалдаг бөгөөд P2P файл солилцоо, шифрлэгдсэн cloud хадгалалт болон хамгаалагдсан файл дамжуулах үйлчилгээнд өргөн ашиглагддаг.

Хувийн үүлэн хадгалалтын системийн аюулгүй байдлын үндсэн зарчим нь хэрэглэгчийн мобил төхөөрөмж болон хувийн сервер хооронд дамжих бүх өгөгдлийг *End-to-End Encryption (E2EE)* аргаар хамгаалахад оршино. Систем нь сүлжээний NAT болон галт ханын хязгаарлалтыг давахын тулд дамжуулагч сервер (*Relay Server*) ашиглах бөгөөд энэ нөхцөлд өгөгдөл замаас задрахгүй байх нь нэн чухал юм. Иймд *No-Trust* буюу үл итгэх загварыг баримтлан, дамжуулагч серверт өгөгдлийг тайлах ямар ч боломж олгохгүйгээр зөвхөн шифрлэгдсэн пакетуудыг дамжуулах бөгөөд энэ нь системийн нууцлалыг хамгийн дээд түвшинд хангана.

1.3 Дотоод сүлжээн дэх төхөөрөмж илрүүлэх механизм (Device Discovery Mechanisms in LAN)

Хувийн үүлэн хадгалалтын системийн хэрэглэгчийн туршлага (UX)-ийг сайжруулах нэг гол хүчин зүйл нь төхөөрөмжүүд хоорондоо ямар нэгэн гар тохиргоо (Manual configuration) шаардахгүйгээр холбогдох боломж юм. Хэрэглэгч өөрийн гэрийн дотоод сүлжээнд (LAN) байгаа тохиолдолд өгөгдлийг заавал гадаад дамжуулагч сервер

(Relay Server) рүү илгээх нь зурвасын өргөнийг үр ашиггүй зарцуулж, дамжуулах хурдыг хязгаарладаг. Иймд систем нь дотоод сүлжээн дэх серверийг автоматаар илрүүлж, шууд холболт тогтоох *Service Discovery* механизмыг хэрэгжүүлэх шаардлагатай.

1.3.1 Zero-Configuration Networking (Zeroconf)

Локал сүлжээнд төхөөрөмжийг IP хаяг болон портын дугаар ашиглан гараар бүртгэх нь динамик IP (DHCP) олголттой орчинд тохиромжгүй юм. Үүнийг шийдвэрлэхэд *Zero-Configuration Networking* буюу *Zeroconf* стандартыг ашиглана. Энэхүү стандарт нь дараах гурван үндсэн асуудлыг шийдвэрлэдэг:

1. **IP хаяг олголт:** Төхөөрөмж өөртөө автоматаар IP хаяг олгох.
2. **Нэр өгөх (Naming):** IP хаяг ашиглалгүйгээр `mycloud.local` гэх мэт хүн уншихад хялбар нэрээр хандах.
3. **Үйлчилгээ илрүүлэх (Service Discovery):** Сүлжээнд ямар үйлчилгээ (жишээ нь: файл хадгалалт) байгааг илрүүлэх.

1.3.2 mDNS (Multicast DNS) болон DNS-SD

mDNS нь төвлөрсөн DNS сервер ашиглахгүйгээр дотоод сүлжээний 5353 портоор дамжуулан *Multicast* пакет илгээх замаар төхөөрөмжийн IP-г олох протокол юм. Энэ нь Apple-ийн *Bonjour* болон Linux-ийн *Avahi* сервист ашиглагддаг хамгийн найдвартай стандарт юм. Манай системийн хувьд iOS (Swift) болон Raspberry Pi (Linux) хооронд холболт тогтооход хамгийн тохиромжтой шийдэл юм.

1.3.3 UDP Broadcast илрүүлэлт

mDNS-ээс гадна илүү хөнгөн бөгөөд "custom"шийдэл нь *UDP Broadcasting* юм. Клиент аппликейшн нь сүлжээний тодорхой порт руу (жишээ нь: 255.255.255.255) "Сервер байна уу?"гэсэн хүсэлт илгээхэд, сервер өөрийн IP хаяг болон төхөөрөмжийн таних тэмдэг (Identity) бүхий хариуг илгээдэг. Энэ нь нэмэлт нүсэр сан (Library) шаарддаггүй тул бага нөөцтэй Raspberry Pi төхөөрөмжид нэн тохиромжтой.

1.3.4 Аюулгүй байдал ба Баталгаажуулалт (Manual Pairing)

Автоматаар илрүүлэх механизм нь сүлжээнд байгаа дурын төхөөрөмж холбогдох эрсдэлийг (Unauthorized access) дагуулдаг. Үүнээс сэргийлэхийн тулд системд *Manual Confirmation* буюу хэрэглэгчийн зөвшөөрлийн логикийг нэмж судалж байна.

- **Pairing Handshake:** Клиент серверт холбогдох хүсэлт илгээхэд серверийн UI дээр хэрэглэгчийн зөвшөөрөл (Accept/Decline) нэхэх.
- **Key Exchange:** Зөвшөөрсөн тохиолдолд төхөөрөмжүүд бие биенийхээ *Public Key*-ийг хадгалж авах бөгөөд дараагийн холболтууд автоматаар баталгаажна.
- **Man-in-the-Middle (MITM) хамгаалалт:** Хоёр төхөөрөмжийн дэлгэц дээр ижилхэн 6 оронтой тоон код харуулснаар хэрэглэгч зөв серверт холбогдож байгаагаа нүдээр баталгаажуулна.

1.4 Raspberry Pi-д зориулсан программ хангамж

1.5 Ашиглах технелогиуд

1.6 Бүлгийн дүгнэлт

БҮЛЭГ 2

... системийн шинжилгээ

2.1 Системийн үйл ажиллагааны тухай дэлгэрэнгүй

Энэхүү систем нь хэрэглэгчээс зурган файлыг авч тухайн зурагт дүрслэгдсэн объектуудын нэрийг олж, тэдгээрийн нэрийг байршилтай нь хамт тухайн зураг дээр нь хүрээлсэн хайрцагт хийж тодотгож харуулдаг объект илрүүлэлт хийдэг веб юм. Объектын илрүүлэлтийг хийж, зургийг хэрэглэгчид буцаан харуулахаас гадна хэрэглэгчид дүрслэгдсэн объектуудыг, илэрсэн тоотой нь хамт текст болон монгол, англи хэл дээрх аудио файл болгон хөрвүүлж өгнө. Хэрэглэгчид утасны дугаар эсвэл өөрийн имейл хаягаар бүртгэл үүсгэж системд нэвтэрвэл тухайн хэрэглэгчийн хэрэглэгчийн бүртгэлээрээ нэвтэрсэн үедээ хийсэн объект илрүүлэлтүүд хадгалагдсан байна. Үүнд объект илрүүлэлт хийсний дараах шинэчлэгдсэн зураг, зурагт хамаарах объект илрүүлэлтийн мэдээллийг агуулсан текст болон аудио файлууд огноотойгоо хамт хадгалагдсан байна. Системийн объект илрүүлэлт нь илрүүлсэн объектоо дараах 80 төрлийн зүйлд хуваан танина. Хүн, унадаг дугуй, машин, мотоцикл, онгоц, автобус, галт тэрэг, ачааны машин, завь, гэрлэн дохио, галын цорго, зогсох тэмдэг, зогсоолын тоолуур, вандан, шувуу, муур, нохой, морь, хонь, үхэр, заан, баавгай, тахь, анааш, үүргэвч, шүхэр, гар цүнх, зангиа, чемодан, фрисби, цана, снөүбоард, спортын бөмбөг, цаасан шувуу, бейсболын цохиур, бейсболын бээлий, скейтборд, серфингийн самбар, теннисний цохиур, лонх, дарсны шил, аяга,с эрээ, хутга, халбага, аяга, банана, алим, сэндвич, жүрж, брокколи, лууван, хот дог, пицца, донат, бялуу, сандал, буйдан, савтай ургамал, ор, хоолны ширээ, бие засах газар, ТВ монитор, зөөврийн компьютер, хулгана, удирдлага, гар, гар утас, богино долгионы зуух, зуух, талх шарагч, угааалтуур, хөргөгч, ном, цаг, ваар, хайч, бамбарууш, үс хатаагч, шүдний сойз.

2.2 Системийг ашиглах хэрэглэгчид

Тус системийг нас, хүйс хамаарахгүйгээр объект илрүүлэлтийг ашиглаж үзэхийг хүссэн, интернэтэд холбогдсон хэн бүр ашиглаж болно. Системийг ашиглаж буй хэрэглэгчдийг бүртгэлтэй болон бүртгэлгүй гэж 2 ангилна.

2.3 Функцийн шаардлага

Хэрэглэгчийн функцийн шаардлага

1. Хэрэглэгч объект илрүүлэлтэд ашиглахыг хүссэн зургаа веб -д байршуулна.
2. Аудио файлын хэлийг монгол эсвэл англи хэлний аль нэгээр нь сонгоно.

Бүртгэлтэй хэрэглэгчийн функцийн шаардлага

1. Хэрэглэгч системд овог нэр, утасны дугаар, имейл хаяг, нууц үг гэсэн талбаруудыг бөглөж бүртгүүлнэ
2. Имейл хаягаар илгээгдэх OTP кодоор бүртгэлээ баталгаажуулна
3. Нэр, утас, имейл хаягийн мэдээллийг агуулсан хэрэглэгчийн бүртгэлийн хэсэгт бүртгэлээ засдаг байна.
4. Бүртгэлтэй хэрэглэгч өөрийн бүртгэлээ устгаж болно, энэ тохиолдолд бүртгэлтэй хэрэглэгчид хамааралтай бүх файл устана.
5. Хэрэглэгчийн бүртгэлээр нэвтэрсэн үедээ ашигласан зургуудын объект илрүүлэлтийг агуулсан шинэчилсэн зургууд, тэдгээрт хамаарах дүрслэгдсэн объектуудын мэдээллийг агуулсан текст болон аудио файлууд огноотойгоо хадгалагдсан байна.
6. Хэрэглэгч өөрт бүртгэлтэй байгаа объект илрүүлэлтийг агуулсан зургууд, тэдгээрт хамаарах текст болон аудио файлтай нь хамт устгаж болно.

Системийн функцийн шаардлага

1. Хэрэглэгчийн байршуулсан зураг дээр объект илрүүлэлт хийж, объект илрүүлэлтийг агуулсан зургийг хэрэглэгчид буцаан харуулдаг байна.
2. Объект илрүүлэлтэд илэрсэн объектуудын нэр болон тоо хэмжээг агуулсан текстэн мэдээллийг хэрэглэгчид харуулна.
3. Объект илрүүлэлтэд илэрсэн объектуудын нэр болон тоо хэмжээг илэрхийлсэн монгол эсвэл англи/хэрэглэгчийн сонголтоор/ хэл дээрх mp4 өргөтгөлтэй аудио файлыг буцаана.
4. Бүртгэл үүсгэж буй хэрэглэгчийн бүртгэлийг баталгаажуулахын тулд имейл хаягаар OTP код илгээнэ.

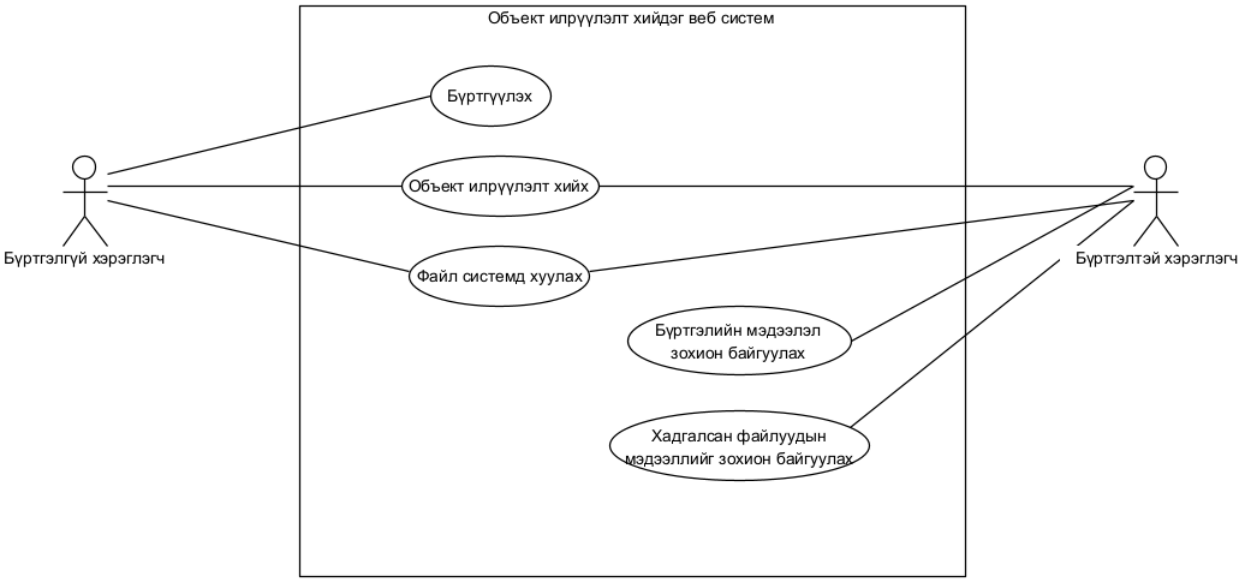
2.4 Функцийн бус шаардлага

1. Хэрэглэгч болон бүртгэлтэй хэрэглэгчийн интерфейс ойлгомжтой, минимал загвартай байна.
2. Бүртгэлтэй хэрэглэгчийн нууц үг хамгийн багадаа 6 ширхэг том, жижиг үсэг, тоо, тэмдэгтийг агуулсан байна.
3. 200MB хүртэлх хэмжээтэй файлыг веб -д хуулна.
4. Веб -д хуулах файл нь jpg, jpeg өргөтгөлтэй байна.
5. Front-end болон back-end системүүдийн хооронд мэдээлэл солилцох API -н хүсэлт нь multipart-form-data хэлбэртэй байх бол хариу нь Application JSON форматтай байна.

6. Package ба Import statements - Пакеж доторх классуудтай холбоотой нэр үгээр пакежийг нэрлэнэ.
7. Class Header ба Declaration - Классын нэрийг тухайн класс дотор хийгдэж байгаа үйл ажиллагаатай холбоотой нэр үгээр нэрлэнэ
8. Method Headers ба Declarations - Метод буюу дэд функцийг тухайн функцийн гүйцэтгэх үйл үгээр нэрлэнэ
9. Функцийн дээд мөрөнд функцийн тайлбарыг коммент хэлбэрээр бичсэн байна.
10. Хөгжүүлэлтийн туршид хөгжүүлэлтийн нэг phase бүр дээр нэгжийн тестийг хийдэг байна.

2.5 Юзкейс диаграмм

Зураг 2.1 -д дүрслэгдсэн объект илрүүлэлт хийдэг веб системийн юзкейс диаграмм нь бүртгэлтэй болон бүртгэлгүй хэрэглэгч гэсэн тоглогчид болон тэдгээрийн ашиглаж болон нийт 5 юзкейсыг агуулсан.



Зураг 2.1: Юзкейс диаграмм

2.6 Юзкейсийн тодорхойлолт

Хүснэгт 2.1: Системд файл хуулах юзкейсийн тодорхойлолт

Юзкейс:	Системд файл хуулах

ID:	1
Үүсгэсэн:	Хүслэн
Үүсгэсэн огноо:	2023.03.15
Үндсэн тоглогч:	Бүртгэлтэй хэрэглэгч эсвэл бүртгэлгүй хэрэглэгч
Нэмэлт тоглогч:	Байхгүй
Товч тайлбар:	Хэрэглэгч объект илрүүлэлтэд ашиглах файлыг веб хуудас руу хуулна.
Өмнөх нөхцөл:	1. Хэрэглэгч объект илрүүлэлт хийдэг вебийг броузер дээрээ нээсэн байна. 2. Веб хуудас руу хуулах гэж файл нь веб хуудас руу хандалт хийж байгаа төхөөрөмж дээр байна.
Үндсэн урсгал:	1. Хэрэглэгч веб хуудасны "Байршуулах"button -г дарна. 2. Төхөөрөмж дээр байх файлуудыг харуулсан "Open"цонх хэрэглэгчид харагдана. 3. Хэрэглэгч веб хуудас руу хуулах файлаа сонгоод "Open"цонхны "Open"button -г дарна. 4. "Open"цонх хаагдана. 5. if Файл хуулалт амжилттай болсон бол 5.1. Вебийн файл байршуулах хэсэгт файл байршсан байна. 6. if Файл хуулалт амжилтгүй болсон бол 6.1. "Амжилтгүй, дахин оролдоно уу + алдааны мессеж"гэсэн контексттэй pop-up -г хэрэглэгчид харуулна. 6.2. Файл байршуулах хэсэгт файл байршаагүй байна.
Дараах нөхцөл:	1. Системд файл хуулагдсан байна.
Альтернатив урсгал:	Байхгүй.

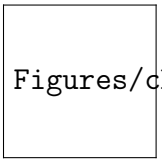
Хүснэгт 2.2: Объект илрүүлэлт хийх юзкейсийн тодорхойлолт

Юзкейс:	Объект илрүүлэлт хийх
ID:	2
Үүсгэсэн:	Хүслэн

Үүсгэсэн огноо:	2023.03.15
Үндсэн тоглогч:	Бүртгэлтэй хэрэглэгч эсвэл бүртгэлгүй хэрэглэгч
Нэмэлт тоглогч:	Байхгүй
Товч тайлбар:	Веб хуудсанд байршуулсан файлд объект илрүүлэлт хийх хүсэлт явуулна.
Өмнөх нөхцөл:	1. Веб хуудсанд объект илрүүлэлтэд ашиглах файлыг хуулсан байна..
Үндсэн урсгал:	1. Хэрэглэгч объект илрүүлэлтийн мэдээллийг хүлээн авах хэлээ сонгосон байна. 2. Хэрэглэгч "Объект илрүүлэх"button дарна. 3. if Объект илрүүлэлт амжилттай бол 3.1. Объект илрүүлэлт хийсэн файлыг хэрэглэгчид файл байршуулах хэсэгт буцаан харуулна. 3.2. Файлд илэрсэн объектуудын нэрийг текст мэдээлэл болгон сонгосон хэл руу хөрвүүлэн "Текст мэдээлэл"хэсэгт харуулна. 3.3. Текст мэдээллийг аудио руу хөрвүүлэн "Аудио мэдээлэл"хэсэгт харуулна. 3.4. Объект илрүүлэлтэд ашигласан файл түүний үр дүнг агуулсан текст болон аудио файлуудыг систем бүртгэлтэй хэрэглэгчийн "Хадгалагдсан файлууд"руу нэмнэ. 4. if Объект илрүүлэлт амжилтгүй болсон бол 4.1. "Амжилтгүй, дахин оролдоно уу + алдааны мессеж"гэсэн контексттэй pop-up -г хэрэглэгчид харуулна. 4.2. Файл байршуулах хэсэгт файл харагдахгүй, хэлний сонголт арилсан байна.
Дараах нөхцөл:	1. Объект илрүүлэлтийн үр дүнг хэрэглэгч харсан байна.
Алтернатив урсгал:	Байхгүй.

2.7 Шинжилгээний класс диаграмм

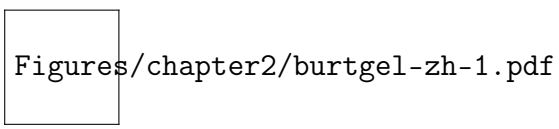
Зураг 2.2 дээрх системийн шинжилгээний шатны класс диаграмм нь системд үндсэн үүрэг гүйцэтгэх "control"төрөлтэй байж болох 6 классыг тодорхойлж тэдгээрийн хоорондын холбоо хамаарлыг дүрсэлсэн. Үр дүнгийн боловсруулалт класс нь объект илрүүлэлт хийх хүсэлтийг хүлээн авч, үр дүнг нэгтгэн боловсруулаад буцаах үүрэгтэй класс, Хэлний төрөл класс нь хөрвүүлэлт хийж болох хэлний төрлүүдийг агуулсан класс, Объект илрүүлэлт класс нь хэрэглэгчийн хуулсан файлд объект илрүүлэлт хийж илэрсэн объектуудын төрөл болон объект илрүүлэлт хийсэн шинэ файлыг Үр дүнгийн боловсруулалт класс руу илгээдэг. Хэрэглэгчийн файл болон Хэрэглэгч классууд нь хэрэглэгчийн мэдээлэл болон хэрэглэгчийн файлын мэдээлэл зохион байгуулдаг классууд, Бүртгэл класс нь бүртгэлтэй холбоотой ОТР үүсгэж, шалгаж, бүртгэл үүсгэдэг класс юм.



Зураг 2.2: Шинжилгээний класс диаграмм

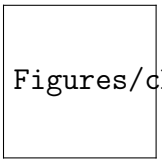
2.8 Шинжилгээний дарааллын диаграмм

Шинжилгээний класс диаграммд тодорхойлсон control төрлийн 5 классын операторуудыг ашиглан файл системд хуулах, объект илрүүлэлт хийх, бүртгүүлэх, хэрэглэгчийн файлын мэдээлэл зохион байгуулах, бүртгэлийн мэдээлэл зохион байгуулах гэсэн юзкейсүүдэд шинжилгээний шатны дарааллын диаграммыг гаргасан.
Бүртгэлтэй хэрэглэгч бүртгэлийн мэдээллээ шинэчлэх эсвэл устгах үед дараах дарааллаар процесс явагдана. Бүртгэлтэй хэрэглэгчийн хүсэлт хэрэглэгчийн мэдээллийг удирдах хэрэглэгч классаар гүйцэтгэгдэнэ. Зураг 2.3 -д дээрх дарааллыг агуулсан хэрэглэгчийн бүртгэлийн мэдээлэл зохион байгуулах шинжилгээний дарааллын диаграммыг харуулж байна.



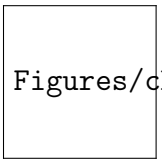
Зураг 2.3: Хэрэглэгчийн бүртгэлийн мэдээлэл зохион байгуулах шинжилгээний дарааллын диаграмм

Хэрэглэгч системд шинээр бүртгэл үүсгэх хүсэлтийг бүртгэл класс хүлээн авч системд бүртгэлтэй эсэхийг шалгаад хэрэв бүртгэлгүй бол бүртгэлийн процессыг дуусгаад хэрэглэгч классаар дамжуулан шинэ хэрэглэгчийг системд бүртгэх хүсэлтийг явуулна. Дээрх процессыг Зураг 2.4 дээрх шинжилгээний дарааллын диаграммд дүрсэлсэн.



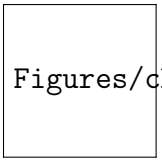
Зураг 2.4: Бүртгэл үүсгэх шинжилгээний дарааллын диаграмм

Зураг 2.5 -н хадгалагдсан файлуудын мэдээллийг зохион байгуулах шинжилгээний дарааллын диаграммд үзүүлсний дагуу системд бүртгэлтэй хэрэглэгч дээр бүртгэгдсэн байгаа объект илрүүлэлт хийсэн файлын мэдээллийг хэрэглэгчид харуулах болон устгах процессыг хэрэглэгчийн файл класс гүйцэтгэнэ.



Зураг 2.5: Хадгалагдсан файлуудын мэдээллийг зохион байгуулах шинжилгээний дарааллын диаграмм

Зураг 2.6 -д үзүүлсний дагуу хэрэглэгчийн хуулсан файлыг үр дүнгийн боловсруулалт класс локал сервер дээрээ хүлээн авч, объект илрүүлэлт хийхэд бэлддэг.



Зураг 2.6: Файл системд хуулах шинжилгээний дарааллын диаграмм

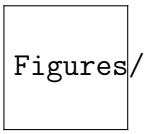
Объект илрүүлэлт хийхийн тулд хэрэглэгч үр дүнгийн боловсруулалт класст файлын мэдээллийг явуулна, дараа нь тус файлд объект илрүүлэлт класс объект илрүүлэлт хийгээд үр дүнг буцаана. Үр дүнгийн боловсруулалт класс текст болон аудио мэдээллийг боловсруулаад, бүртгэлтэй хэрэглэгчийн хүсэлт эсэхийг хэрэглэгч классаар дамжуулан шалгаад, хэрэв тийм бол хэрэглэгчийн файл класс руу объект илрүүлэлт хийсэн файлын мэдээллийг системд нэмэх хүсэлтийг илгээгээд үр дүнг хэрэглэгчид буцаана, бүртгэлгүй хэрэглэгчийн хүсэлт бол үр дүнг шууд буцаана. Зураг 2.7 нь

дээрх процессыг агуулсан объект илрүүлэлт хийх шинжилгээний дарааллын диаграммыг харуулж байна.

Зураг 2.7: Объект илрүүлэлт хийх шинжилгээний дарааллын диаграмм

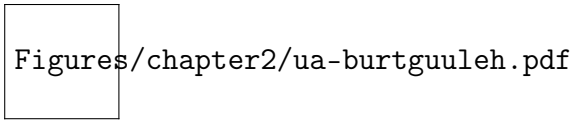
2.9 Үйл ажиллагааны диаграмм

Объект илрүүлэлт хийх веб системийн юзкейс диаграммд тодорхойлогдсон юзкейсүүдийн эхлэлээс төгсгөл хүртэлх үйлдлүүдийг дүрсэлсэн үйл ажиллагааны диаграммуудыг доор харуулав.
Хэрэглэгч бүртгэлийн мэдээллээ зохион байгуулах үйл ажиллагааг эхлэхийн тулд эхлээд системд нэвтэрсэн байна. Хэрэглэгч мэдээлэл шинэчлэх хэсгийг сонгосноор өөрийн бүртгэлийн мэдээллээ харна. Хэрэглэгч шинэчлэхийг хүссэн мэдээллээ шинэчилж оруулаад шинэчлэх хүсэлт илгээнэ. Систем хүсэлтийг хүлээн аваад мэдээллийг шинэчлээд хэрэглэгчид бүртгэгдсэн эсэх мэдээллийг хүргэнэ, хэрэв бүртгэгдээгүй бол хэрэглэгч хуучин бүртгэлийн мэдээллээ дахин харснаар дээрх процесс давтагдаж болно. Дээрх процессыг Зураг 2.8 -н хэрэглэгчийн бүртгэлийн мэдээлэл зохион байгуулах үйл ажиллагааны диаграммаас харж болно.



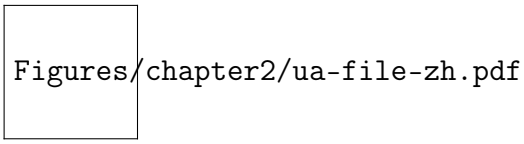
Зураг 2.8: Хэрэглэгчийн бүртгэлийн мэдээлэл зохион байгуулах үйл ажиллагааны диаграмм

Зураг 2.9 -н үйл ажиллагааны диаграмм нь дараах процессыг дүрсэлнэ. Хэрэглэгч бүртгэлийн мэдээллээ бөглөж оруулаад системд бүртгэлтэй эсэхээ шалгуулна, бүртгэлтэй бол хэрэглэгчид бүртгэлтэй гэсэн мэдээллийг хүргээд үйл ажиллагаа дуусна, бүртгэлгүй бол систем хэрэглэгчийн имейл руу OTP -г илгээж, баталгаажуулалтыг хүлээнэ. Баталгаажуулалт амжилттай бол бүртгэл үүсэж үйл ажиллагаа дуусна харин амжилтгүй бол хэрэглэгчээс код дахин илгээх эсэхийг лавлаж үйл ажиллагаа OTP илгээх үйлдэл руу буцна.



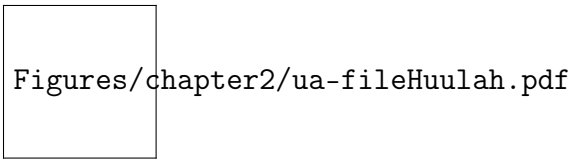
Зураг 2.9: Бүртгэл үүсгэх үйл ажиллагааны диаграмм

Хадгалагдсан файлуудын мэдээллийг зохион байгуулахын тулд хэрэглэгч нэвтэрсэн байна. Дараа нь мэдээлэл харах хэсгээс өөрт байгаа файлуудын мэдээллийг харна. Устгахыг хүссэн файлынхаа ард байрлах устгах товчийг дарна, систем хэрэглэгчийг устгах эсэхийг лавлаж асууна. Хэрэглэгч файлыг устгана гэдгийг баталгаажуулснаар файл устаж үйл ажиллагаа дуусна. Харин хэрэглэгч лавлах асуултад цуцлах хэсгийг сонгож файлыг устгахгүйгээр үйл ажиллагааг дуусгаж болно. Зураг 2.10 дээрх хадгалагдсан файлуудын мэдээллийг зохион байгуулах үйл ажиллагааны диаграмм дээрх процессыг дүрсэлж байна.



Зураг 2.10: Хадгалагдсан файлуудын мэдээллийг зохион байгуулах үйл ажиллагааны диаграмм

Зураг 2.11 -д үзүүлсэн файл системд хуулах үйл ажиллагааны диаграммын дагуу хэрэглэгчийн сонгосон файл эхлээд веб хуудсанд хуулагдана, дараа нь объект илрүүлэх хүсэлттэй хамт илгээгдэж ирснээр back-end сервер дээр хуулагдана.



Зураг 2.11: Файл системд хуулах үйл ажиллагааны диаграмм

Объект илрүүлэлт хийх үйл ажиллагаа эхлэхийн өмнө файл системд хуулагдсан байна. Файлд эхлээд объект илрүүлэлт хийгээд сонгогдсон хэл дээр текст болон аудио мэдээллийг боловсруулна. Хэрэглэгч системд нэвтэрсэн эсэхийг шалгаад үнэн бол үр дүнг хэрэглэгчийн файлын санд хадгалаад дараа нь үр дүнг хэрэглэгчид буцаасснаар үйл ажиллагаа дуусна. Хэрэв хэрэглэгч нэвтэрч ороогүй байвал үр дүнг шууд буцааснаар үйл ажиллагаа дуусгавар болно. Зураг 2.12 -д дараах процессыг үйл ажиллагааны диаграммаар дүрсэлсэн.

Figures/chapter2/ua-objectDetection.pdf

Зураг 2.12: Объект илрүүлэлт хийх үйл ажиллагааны диаграмм

2.10 Бүлгийн дүгнэлт

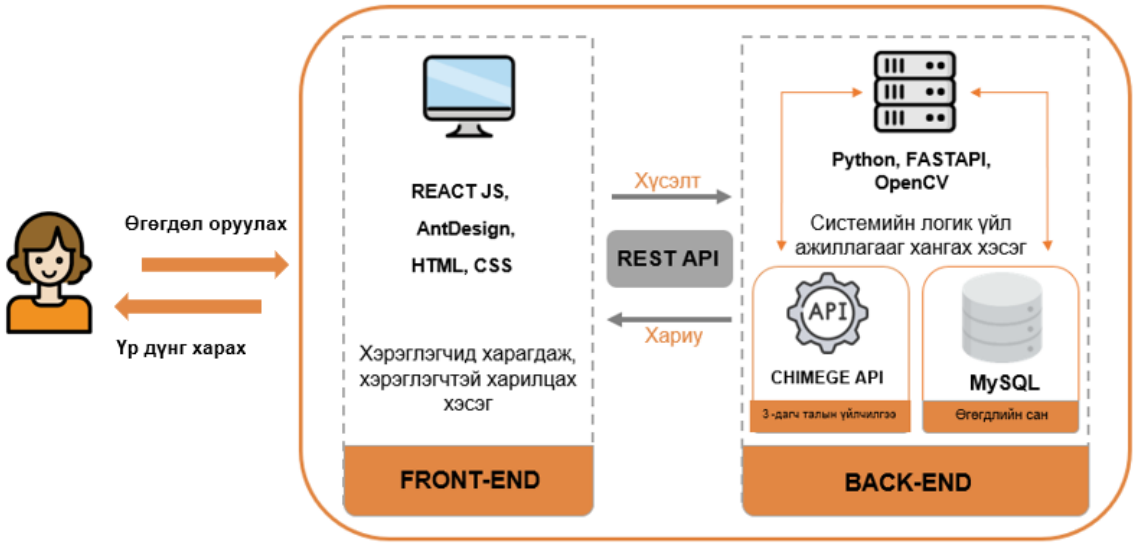
Энэ бүлэгт системийн үйл ажиллагааг дэлгэрэнгүй тайлбарлаж, системийг ашиглах хэрэглэгчид болон шаардлагуудыг тодорхойлж, юзкейс, юзкейсийн тодорхойлолт болон шинжилгээний шатны класс, дараалал, үйл ажиллагааны диаграммуудыг гаргасан. Эхлээд системийг хэрэглэх хэрэглэгчдийг тодорхойлоод системд оролцогч тал бүрийн функцийн шаардлагыг тодорхойлж мөн системийн функцийн бус шаардлагыг ч тодорхойлсон. Системд оролцогчдын функцийн шаардлага дээр үндэслэн 2 тоглогчтой 5 юзкейс бүхий юзкейс диаграммыг байгуулж, юзкейс бүрд тодорхойлолт гаргасан. Юзкейсийг хэрэгжүүлж болох шинжилгээний шатны 6 классуудыг гаргаж тэдгээрийн холбоо хамаарлыг тодорхойлсон. Тодорхойлогдсон классууд болон юзкейсийн тодорхойлолтын дагуу дарааллын диаграмм болон үйл ажиллагааны диаграммуудыг гаргаж системийн үйл ажиллагаа, оролцогч талууд, процессын дарааллыг шинжилгээний түвшинд ойлгох боломжийг олгосон.

БҮЛЭГ 3

... системийн зохиомж ба хөгжүүлэлт

3.1 Системийн архитектур

Зургаас объект илрүүлэлт хийдэг веб систем нь Back-end болон Front-end гэсэн үндсэн хоёр хэсгээс бүрдэнэ. Зураг 3.1 -д үзүүлсний дагуу Front-end хэсэг нь хэрэглэгчид харагдаж, хэрэглэгчтэй харилцах хэсэг бөгөөд хэрэглэгчийн оролтын мэдээллийг авч Back-end хэсгээс гаргасан сервисийг ашиглан Back-end хэсэг рүү хүсэлт явуулж, хариуг аван хэрэглэгчийг гаралтын мэдээллээр хангана. Back-end хэсгийн хувьд системийн логик үйл ажиллагааг хангаж ажилладаг. Хэрэглэгчийн нэвтрэлт, бүртгэл, объект илрүүлэлттэй холбоотой сервисүүдийг гаргаж, монгол хэл дээрх текстийг аудио руу хөрвүүлж өгөх 3-дагч талын сервис болох ChimegeAPI, хэрэглэгчдийн мэдээлэл болон тэдгээрт хамааралтай объект илрүүлэлтэд ашигласан файлын мэдээллийг хадгалах MySQL өгөгдлийн сантай ажиллана.



Зураг 3.1: Системийн архитектур

3.2 Зохиомжийн шатны класс диаграмм

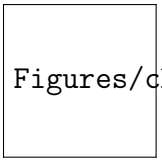
Объект илрүүлэлт хийдэг веб системийн зохиомжийн шатны класс диаграммд control төрлийн Хэрэглэгч, Үр дүнгийн боловсруулалт, Хэлний төрөл, Хэрэглэгчийн файл , Объект илрүүлэлт, Бүртгэл гэсэн 6-н класс ,entityтөрлийн Хэрэглэгчийн файлын мэдээлэл, Хэрэглэгчийн мэдээлэл гэх 2 класс, boundary төрлийн Объект илрүүлэлтийн дэлгэц,Бүртгэлийн дэлгэц, OTP шалгах дэлгэц, Хэрэглэгчийн файлын мэдээллийн дэлгэц, Хэрэглэгчийн мэдээллийн дэлгэц гэсэн 5-н класс буюу нийт 13

классын аттрибут болон операторуудыг тодорхойлсон. Зураг 3.2 -д тэдгээр классуудын стерео төрлийг тодорхойлж, класс хоорондын холбоо хамаарлыг ассосиэйшн болон бүрдмэл холбоо хамаарлаар дүрсэлж, зарим класс хоорондын холбоо хамааралд ашиглалтын хараат байдлыг дүрсэлсэн.

Зураг 3.2: Класс диаграмм

3.3 Өгөгдлийн ерөнхий схем

Объект илрүүлэлт хийдэг веб системийн өгөгдлийн сан нь хэрэглэгч болон хэрэглэгчийн файл гэсэн хоёр хүснэгтийг агуулна. Хэрэглэгч хүснэгт нь системд бүртгүүлсэн хэрэглэгчдийн овог нэр, имейл, нууц үг болон хэрэглэгчийн кодыг анхдагч түлхүүрээр хадгална. Хэрэглэгчийн файл хүснэгт нь файлын нэр, локал сервер дээр хадгалагдсан байршил, текст, аудио мэдээлэл, огноо, хэрэглэгчийн кодыг гадаад түлхүүрээр, файлын кодыг анхдагч түлхүүрээр хадгална. Зураг 3.3 -д системийн хөгжүүлэлтийн хүрээнд үүсгэсэн өгөгдлийн сангийн "Хэрэглэгч", "Хэрэглэгчийн файл"гэсэн хүснэгтүүд, тэдгээрийн аттрибутууд болон тэдгээр хүснэгтүүд "1 : олон"харьцаагаар дүрслэгдсэнийг харуулж байна.



Зураг 3.3: Өгөгдлийн ерөнхий схем

3.4 Системийн прототип

Системийн үндсэн хуудас, бүртгүүлэх болон нэвтрэх хуудас, бүртгэлтэй хэрэглэгчийн мэдээллээ шинэчлэх болон файлын мэдээллүүдээ зохион байгуулах хуудаснуудын прототип загварыг дүрслэн харуулна.
Зураг 3.4 -д зургаас объект илрүүлэх вебийн хэрэглэгчийн бүртгэлээр ороогүй байх үеийн объект илрүүлэлт хийх нүүр хуудас харагдаж байна.

Зураг 3.4: Зургаас объект илрүүлэлт хийх хуудасны прототип

Зураг 3.4 -д үзүүлсэн хэрэглэгчийн бүртгэлээр нэвтрээгүй байх үеийн нүүр хуудасны толгой хэсэгт байрлах "Бүртгүүлэх"лабел -г дарснаар тус хуудас гарч ирнэ. Зураг 3.5 -д хэрэглэгчийн бүртгэлийн мэдээллийг авч бүртгэл үүсгэх хуудсыг харуулж байна.

Зураг 3.5: Бүртгэл үүсгэх хуудасны прототип

Нүүр хуудасны толгой хэсэгт байрлах "Нэвтрэх"лабел -г дарснаар тус хуудас гарч ирнэ. Зураг 3.6 -д хэрэглэгчийн имейл хаяг, нууц үгээр вебд нэвтрэх хуудсыг харуулж байна. Тус хуудас дээрх "Нууц үг сэргээх"лабел нь нууц үг сэргээх хуудас руу чиглүүлнэ.

Зураг 3.6: Нэвтрэх хуудасны прототип

Хэрэглэгчийн бүртгэлээр нэвтрээд хуудасны толгой хэсэгт байрлах жагсаалтыг илэрхийлсэн icon дээр дарснаар Зураг 3.7 -д үзүүлсэн хуудас харагдана. Тус зурагт хэрэглэгчид бүртгэлтэй объект илрүүлэлтэд ашигласан зураг, түүний үр дүнгийн текст болон аудио файлууд харагдана. Жагсаалт дахь файл бүрийн ард байрлах устгах icon нь харгалзах зургийн мэдээллээ устгах бол "Бүгдийг устгах"лабел -тэй icon нь бүх бүртгэлтэй файлыг устгана.

Зураг 3.7: Бүртгэлтэй файлын мэдээлэл харах хуудасны прототип

Хэрэглэгчийн бүртгэлээр нэвтрээд хуудасны толгой хэсэгт байрлах хэрэглэгчийг илэрхийлсэн icon дээр дарснаар Зураг 3.8 -д үзүүлсэн дэлгэц харагдана. Дэлгэцэд "Бүртгэлийн мэдээлэл шинэчлэх", "Гарах"гэсэн сонголтуудыг идэвхжүүлж харуулна.

Зураг 3.8: Хэрэглэгчийн сонголт харуулах дэлгэцийн прототип

Зураг 3.8 -д үзүүлсэн дэлгэц дээрх сонголтуудаас ¨үртгэлийн мэдээлэл шинэчлэх¨онголтыг сонгосноор Зураг 3.9 -д үзүүлсэн хуудас гарч ирнэ. Тус зурагт үзүүлсэн хуудас нь

хэрэглэгчийн бүртгэлийн мэдээллийг харуулаад, шинэчилсэн мэдээллийг аваад хадгална.

Figures/chapter3/Frame 6.pdf

Зураг 3.9: Бүртгэлийн мэдээлэл шинэчлэх хуудасны прототип

Зураг 3.10 -д бүртгэлийн мэдээлэл амжилттай шинэчлэгдсэний дараах pop-up хэлбэрийн мэдээлэл хэрэглэгчид хэрхэн харагдахыг харуулж байна.

Figures/chapter3/Frame 7.pdf

Зураг 3.10: Бүртгэлийн мэдээлэл амжилттай шинэчлэгдсэнийг харуулах дэлгэцийн прототип

Зураг 3.11 -д бүртгэлийн мэдээлэл оруулсны дараа хэрэглэгчийн бүртгэлдээ ашигласан имейл хаяг руу илгээгдэх бүртгэл баталгаажуулах OTP оруулах хуудсыг харуулж байна.

Figures/chapter3/Frame 8.pdf

Зураг 3.11: OTP оруулах хуудасны прототип

Хэрэглэгч OTP -г зөв оруулснаар бүртгэл баталгаажна. Зураг 3.12 -д бүртгэл амжилттай үүссэнийг харуулах pop-up мэдээлэл хэрэглэгчид хэрхэн харагдахыг харуулж байна.

Figures/chapter3/Frame 9.pdf

Зураг 3.12: Бүртгэл амжилттай үүссэнийг харуулах дэлгэцийн прототип

Зураг 3.13 -д хэрэглэгч буруу OTP оруулсны дараах дахин OTP авах эсвэл бүртгэлийг цуцлах сонголтуудыг агуулсан pop-up мэдээлэл хэрхэн харагдахыг харуулсан.

Figures/chapter3/Frame 10.pdf

Зураг 3.13: Код дахин илгээх эсэхийг лавлах дэлгэцийн прототип

3.5 Хөгжүүлсэн системийн интерфейс

Хөгжүүлэгдсэн системийн объект илрүүлэлт хийх, нэвтрэх, бүртгүүлэх, OTP шалгах хуудаснууд болон хэрэглэгчийн бүртгэлээр нэвтэрч орсон үеийн объект илрүүлэлт хийх хуудас , өөрт бүртгэлтэй зургуудын жагсаалт харах хуудасны интерфейсийг энэ хэсэгт тайлбарлана.
Зураг 3.14 -д харуулсан интерфейстэй хуудас нь хэрэглэгчийг веб хуудас руу хэрэглэгчийн бүртгэлгүйгээр хандах үед харагдана. Хуудасны зураг оруулах хэсэгт зургийг оруулснаар зургийн нэр доод хэсэгт харагдана. Англи болон монгол гэсэн сонголтуудаас хэлний сонголтоо хийгээд "Илрүүлэлт хийх"товчийг дарснаар текст болон аудио мэдээлэл хэсгүүдэд үр дүн харагдана.

Figures/chapter3/home.pdf

Зураг 3.14: Объект илрүүлэлт хийх хуудасны интерфейс

Зураг 3.15 -д системийн нэвтрэх хуудасны интерфейсийг харуулсан. Бүртгэлтэй хэрэглэгч имейл хаяг болон нууц үгээ оруулаад "Нэвтрэх"товчийг дарж системд нэвтэрнэ. Хэрэглэгчийн оруулсан имейл хаяг бүртгэлгүй эсвэл нууц үг буруу тохиолдолд алдааны мессежийг хэрэглэгчид харуулна. Хэрэв бүртгэлгүй бол "Одоо бүртгүүлэх"гэсэн лабелийг дарж бүртгүүлэх хуудас руу очиж болно.

Figures/chapter3/login.pdf

Зураг 3.15: Нэвтрэх хуудасны интерфейс

Зураг 3.16 -д хэрэглэгчийн бүртгэл үүсгэх хуудасны интерфейс харагдаж байна. Хэрэглэгч бүртгэлийн мэдээллээ оруулаад "Бүртгүүлэх"товч дарна. Хэрэв хэрэглэгчийн бүртгүүлсэн имейл хаяг бүртгэлтэй байвал "Бүртгэлтэй имейл байна"гэсэн алдааны мессежийг хэрэглэгчид харуулна, имейл хаяг бүртгэлгүй байвал хэрэглэгчийн имейл хаяг руу OTP явуулаад OTP шалгах хуудас руу чиглүүлнэ.

Figures/chapter3/signup.pdf

Зураг 3.16: Бүртгэл үүсгэх хуудасны интерфейс

Зураг 3.17 -д хэрэглэгч рүү илгээсэн OTP -г шалгах хуудасны интерфейсийг харуулж байна. Хэрэглэгч OTP -г оруулаад "OTP шалгах"товчийг дарна. Хэрэв OTP зөв байвал хэрэглэгчийн бүртгэл амжилттай үүссэнийг хэрэглэгчид мессежээр мэдэгдээд

нэвтрэх хуудас руу чиглүүлнэ. Буруу бол "Таны оруулсан OTP буруу байна"гэсэн мессежийг хэрэглэгчид хүргээд хуудас солигдохгүй. Хуудасны "Арилгах"товч оруулсан OTP -г устгадаг, "Бүртгэл цуцлах"товч таны оруулсан мэдээллүүдийг устгаж бүртгэл үүсгэх процессыг цуцлаад бүртгүүлэх хуудас руу чиглүүлнэ.

Зураг 3.17: OTP шалгах хуудасны интерфейс

Зураг 3.18 -д системд хэрэглэгчийн бүртгэлээр нэвтэрсэн байх үеийн объект илрүүлэлт хийх хуудасны интерфейсийг харуулж байна. Хуудасны толгой хэсэгт жагсаалт болон хэрэглэгчийг дүрсэлсэн icon -ууд харагдана. Тус хуудаснаас жагсаалтыг дүрсэлсэн icon дээр дарвал хэрэглэгчид бүртгэлтэй, объект илрүүлэлт илрүүлэлт хийсэн зургуудын жагсаалт харагдана. Хэрэглэгчийг дүрсэлсэн icon дээр дарвал "Бүртгэл шинэчлэх"болон "Гарах"гэсэн сонголтууд харагдана.

Зураг 3.18: Хэрэглэгчийн бүртгэлээр орсон үеийн объект илрүүлэлт хийх хуудас

Зураг 3.19 -д хэрэглэгчид бүртгэлтэй зургуудын жагсаалтыг харуулдаг хуудасны интерфейсийг дүрсэлсэн. Жагсаалтын нэг item бүрд объект илрүүлэлт хийсэн зураг, текст болон аудио үр дүн, объект илрүүлэлт хийсэн огноо болон "Устгах"icon байна. Хэрэглэгч "Устгах"icon -г дарснаар тус icon -д харгалзах зургийг устгах боломжтой.

Зураг 3.19: Бүртгэлтэй зургуудын жагсаалт харуулах интерфейс

3.6 Системд хийгдсэн тест

3.7 Системийн нэвтрүүлэлт

3.8 Бүлгийн дүгнэлт

Энэ бүлэгт ерөнхийдөө системийн зохиомжийг гаргасан. Системийн архитектур, класс диаграмм, дарааллын диаграмм, өгөгдлийн ерөнхий схем, системийн прототип

загвар болон хөгжүүлэгдсэн системийн интерфейсийг багтаасан. Системийн архитектурыг front-end болон back-end гэсэн үндсэн 2 хэсэгт хувааж, тэдгээр хэсгүүд дээр хийгдэх үйл ажиллагаануудыг бүдүүвч байдлаар дүрсэлсэн. Ингэхдээ front-end хэсгийг хэрэглэгчид харагдаж, хэрэглэгчтэй харилцах хэсэг, back-end хэсгийг системийн логик үйл ажиллагааг хангах хэсэг гэж тодорхойлоод REST API ашиглан тэдгээрийн дунд холболт үүсгэхээр дүрсэлсэн. Класс диаграмм дээр шинжилгээний шатны control төрлийн 6-н класс дээр нэмээд boundary төрлийн 5-н класс, entity төрлийн 2 классыг гаргаж тэдгээрийн харьцаа хамаарлыг тодорхойлсон. Шинээр тодорхойлсон классуудыг оролцуулан илүү дэлгэрэнгүй, ойлгомжтой дарааллын диаграммуудыг системийн юзкейс болгон дээр үүсгэсэн. Өгөгдлийн ерөнхий схемийг зохиомжлохдоо Хэрэглэгч, Хэрэглэгчийн файл гэсэн 2 хүснэгтүүдийг үүсгэн аттрибутуудыг тодорхойлоод, тэдгээрийг (1 : олон) харьцаагаар тодорхойлсон. Хөгжүүлэлт эхлэхээс өмнөх системийн прототип загварыг boundary төрлийн классуудын хувьд гаргасан. Мөн хөгжүүлэлтийн дараах системийн интерфейсийг хөгжүүлэгдсэн вебийн хуудас бүрээр оруулсан.

Ерөнхий дүгнэлт

“Зургаас объект илрүүлэлт хийх веб хөгжүүлэх нь” сэдэвтэй дипломын ажлын зорилго нь зурагт дүрслэгдэж буй объектуудыг таних, тэдгээр объектын нэрсийг тодорхойлох, байршлыг дүрслэн харуулах, дүрслэгдсэн объектуудын нэрийг текст болон монгол, англи хэл дээрх аудио хэлбэр рүү хөрвүүлэх, мөн хэрэглэгчид өмнө нь хөрвүүлсэн зураг, тухайн зургийн текст, аудио мэдээллийг хадгалах чадвартай веб сайтыг хөгжүүлэх юм. Тус веб сайтыг хөгжүүлснээр хэрэглэгчид зурагт агуулагдаж буй объектуудыг ялгаж, таньж мэдэх, илэрсэн объектуудыг нэрээр нь ангилан тоолох, зурагт дүрслэгдсэн жижиг объектуудыг олж харах, харааны бэрхшээлтэй иргэд илэрсэн объектуудын мэдээллийг аудио хэлбэрээр сонссоноор зургийн агуулгыг төсөөлөх гэх мэт боломжийг олгоно.

Системийг хөгжүүлэхийн тулд объект илрүүлэлт гэж юу болох талаар судалж, хэрэглэгч болон системийн шаардлагуудыг тодорхойлж, системийг хөгжүүлэхэд шаардлагатай ЮМЛ -н диаграммуудыг гаргасан. Системийн архитектур нь хэрэглэгчтэй харилцах хэсэг болох front-end болон системийн логик үйл ажиллагааг хангах хэсэг болох back-end гэсэн 2 хэсгээс бүрдэж хоорондоо REST API ашиглан холбогддог байхаар зохиомжилсон. MobileNet SSD V3 архитектурын хөлдөөсөн моделийг авч OpenCV -н DetectionModel классаар объект илрүүлэлт хийх сүлжээ үүсгэж, түүгээрээ дамжуулан зургаас объект илрүүлэлтийг гүйцэтгэж байгаа юм. Текст мэдээллийг аудио руу хөрвүүлэхдээ Python gTTS сан, Chimege API -г ашигласан. Системийн хэрэглэгчтэй харилцах хэсгийг ReactJS фреймворк ашиглан хөгжүүлсэн.

Энэхүү дипломын ажил нь зургаас объект илрүүлэлт хийх веб хөгжүүлэхэд ашигласан технологиуд, хэд хэдэн объект илрүүлэлтийн архитектурын талаарх мэдээллийг агуулсан бичиг баримт ба зургаас объект илрүүлэлт хийх веб сайтыг багтаасан болно.

Чухал кодын хэсгүүд

Хэрэглэгчийн бүртгэл үүсгэх кодын хэсэг

```
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

                if M is None:
                    M = np.copy(VT)
                else:
                    M = np.concatenate((M, VT), 1)

                VT = np.zeros((n*m,1), int)

    return M
```

Ашигласан материалын жагсаалт

[1] Threadgold consulting. *Personal Cloud Storage Usage*. Personal Cloud Storage Usage report. 2025. URL: https://threadgoldconsulting.com/research/personal-cloud-storage-usage (**urlseen** 01/03/2026).

[2] Nextcloud GmbH. *Nextcloud Architecture Whitepaper*. Whitepaper. Accessed: 2026-03-01. Nextcloud, **july** 2018. URL: https://nextcloud.com/c/uploads/2022/03/Architecture-Whitepaper-WebVersion-072018.pdf.

[3] Syncthing Project. *Syncthing Protocol Specifications*. Syncthing Documentation. 2024. URL: https://docs.syncthing.net/specs/index.html (**urlseen** 01/03/2026).

[4] IBM. *What is end-to-end encryption (E2EE)?* Information about E2EE. 2026. URL: https://www.ibm.com/think/topics/end-to-end-encryption (**urlseen** 01/03/2026).