

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

Нямдоржын Энхболд

Урамшууллын системийн цогц хөгжүүлэлт
(Full stack loyalty system developing)

Програм хангамж (D061302)
Үйлдвэрлэлийн дадлагын тайлан

Улаанбаатар

2023 оны 09 сар

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

Урамшууллын системийн цогц хөгжүүлэлт
(Full stack loyalty system developing)

Програм хангамж (D061302)
Үйлдвэрлэлийн дадлагын тайлан

Гүйцэтгэсэн: _____ Н.Энхболд (20B1NUM0102)

Улаанбаатар
2023 оны 09 сар

Зохиогчийн баталгаа

Миний бие Нямдоржын Энхболд "Урамшууллын системийн цогц хөгжүүлэлт" сэдэвтэй судалгааны ажлыг гүйцэтгэсэн болохыг зарлаж дараах зүйлсийг баталж байна:

- Ажил нь бүхэлдээ эсвэл ихэнхдээ Монгол Улсын Их Сургуулийн "Үйлдвэрлэлийн дадлага (INTE401)" хичээлийг тооцуулахаар дэвшүүлсэн болно.
- Энэ ажлын аль нэг хэсгийг эсвэл бүхлээр нь ямар нэг их, дээд сургуулийн зэрэг горилох, хичээл тооцуулахаар оруулж байгаагүй.
- Бусдын хийсэн ажлаас хуулбарлаагүй, ашигласан бол ишлэл, зүүлт хийсэн.
- Ажлыг би өөрөө (хамтарч) хийсэн ба миний хийсэн ажил, үзүүлсэн дэмжлэгийг дипломын ажилд тодорхой тусгасан.
- Ажилд тусалсан бүх эх сурвалжид талархаж байна.

Гарын үсэг: _____

Огноо: _____

ГАРЧИГ

УДИРТГАЛ	1
1. ТӨЛӨВЛӨГӨӨ	2
2. БАЙГУУЛЛАГЫН ТУХАЙ.....	3
2.1 Компаний тухай.....	3
2.2 Үйл ажиллагаа	4
2.3 Технологиуд болон системүүд	4
3. НООМЕ ТӨСӨЛ	6
3.1 Хотхон	6
3.2 Контор	7
3.3 Мобайл аппликейшн	7
4. ТЕХНОЛОГИЙН СУДАЛГАА	9
4.1 BLoC	9
5. УРАМШУУЛЛЫН СИСТЕМ	11
6. ИШЛЭЛ, ЗҮҮЛТ	12
6.1 Ишлэл	12
6.2 Зүүлт	12
7. ЗУРАГ	13
7.1 Зургийн хэмжээ өөрчлөх	13
7.2 Зураг эргүүлэх	14
7.3 Зургийн нэр.....	15
7.4 Зураг голлуулах	16
7.5 Зургийн чанар	16
8. ХҮСНЭГТ ОРУУЛАХ.....	17
8.1 Хүснэгт зурах хэрэгсэл	17
9. КОД БА АЛГОРИТМ ОРУУЛАХ	18

ДҮГНЭЛТ	20
НОМ ЗҮЙ	20
ХАВСРАЛТ.....	21
А. ШИНЖИЛГЭЭ ЗОХИОМЖ	22
В. КОДЫН ХЭРЭГЖҮҮЛЭЛТ.....	23

ЗУРГИЙН ЖАГСААЛТ

3.1	Hoome - Коммунити нүүр.	8
3.2	Hoome - Cloud Car Parking нүүр.	8
4.1	BLoC	10
7.1	Зураг голлуулах	15
7.2	Зургийн нэрийг энд бичнэ	16

ХҮСНЭГТИЙН ЖАГСААЛТ

1.1	Үйлдвэрлэлийн дадлагын төлөвлөгөө	2
8.1	Хүснэгтийн нэр. Хүснэгтийн нэр хүснэгтийн дээд талд байрлана.	17

Кодын жагсаалт

9.1	С хэлний кодын жишээ	18
9.2	Кодын файлаас хэсэгчилж оруулах	18

УДИРТГАЛ

Миний бие Н. Энхболд нь үйлдвэрлэлийн дадлагын хугацааны хүрээнд "MOGUL" группийн салбар компани болох "Nomadic Software Solution" ХХК-д хөгжүүлэгчдийн ашигладаг технологиудыг судалж, сурсан мэдлэгээрээ HPoint төслийг эхлүүлсэн бөгөөд нийт 5 хүний бүрэлдэхүүн бүхий багийг ахлаж төслийг эхний байдлаар хэрэгжүүлсэн билээ.

HPoint төслийн гол зорилго нь "Hoome" платформд урамшууллын програмыг нэвтрүүлснээр хэрэглэгчдийн тоог үнэмлэхүйц байдлаар өсгөх юм.

Төслийн баг нь архитектор, back-end хөгжүүлэгч, програм хангамж хөтөлбөрийн 2 дадлагын оюутнуудаас бүрдэх бөгөөд төслийн хүрээнд мобайл хөгжүүлэлт, микросервис хөгжүүлэлт, өгөгдлийн сангийн зохиомж, хэрэглэгчийн интерфэйсийн зохиомж зэрэг ажлууд өрнөсөн болно.

Үйлдвэрлэлийн дадлагын хугацаанд эдгээр ажлуудад ашиглагдах технологиудыг судалж, хэрэгжүүлэлтүүдийг гүйцэтгэсэн билээ.

1. ТӨЛӨВЛӨГӨӨ

Table 1.1: Үйлдвэрлэлийн дадлагын төлөвлөгөө

№	Гүйцэтгэх ажил	Хугацаа	Төлөв	Удирдагчийн үнэлгээ
1	Spring boot технологи болон JHipster-ийн талаар судлах	1 өдөр	Гүйцэгтэсэн	10/10
2	Keycloak технологийн талаар судлах	1 өдөр	Гүйцэгтэсэн	10/10
3	Өмнө хөгжүүлэгдсэн микросервис дээр feature нэмэх	3 өдөр	Гүйцэгтэсэн	10/10
4	Flutter-ийн BLoC технологийн талаар судлах	1 өдөр	Гүйцэгтэсэн	10/10
5	Noome мобайл аппликейшн дээр зогсоолын төлбөр төлөх хэсэг дээр bug засах, карт холбох feature нэмэх	4 өдөр	Гүйцэгтэсэн	10/10
6	Noome платформын хэрэглэгчийн тоог өсгөх шийдэл олох, зохиомжийг гаргах	1 өдөр	Гүйцэгтэсэн	10/10
7	Микросервис хөгжүүлэлтийг хийх API endpoint-уудыг бэлдэх (Багаар)	6 өдөр	Гүйцэгтэсэн	10/10
8	”Noome” мобайл аппликейшн дээр шийдлээ оруулж өгөх, дизайныг хэрэгжүүлэх	2 өдөр	Гүйцэгтэсэн	10/10
9	Микросервис дээр integration тест бичих	2 өдөр	Гүйцэгтэсэн	10/10

2. БАЙГУУЛЛАГЫН ТУХАЙ

”MOGUL” групп нь 1997 онд компьютер, компьютерийн тоног төхөөрөмж нэвтрүүлэх, үйлчилгээ үзүүлэх зорилгоор үйл ажиллагаагаа эхэлж байсан бөгөөд Мэдээллийн Технологийн чиглэлээр төрөлжин үйл ажиллагаа явуулдаг 6 компанитайгаар 26 дахь жилдээ ажиллаж байгаа. Нийт 380 гаруй ажилтан, тэдгээрийн 200 гаруй ажилтан нь инженерүүд байдаг. Мэдээллийн технологийн дэд бүтэц, тоног төхөөрөмжийн худалдаа, үйлчилгээ, Мэдээллийн болон биет аюулгүй байдал, Программ хангамж үйлдвэрлэл, Цахим засаг, Клауд болон Менежед үйлчилгээ, Дата болон AI, салбаруудын мэдээллийн технологийн шийдэл чиглэлээр үйл ажиллагаа явуулдаг.

2.1 Компаний тухай

”MOGUL” групп нь 2023 оны өвлийн улиралд 6 салбар компанитай байсан бөгөөд эдгээрт:

- ITZone ХХК
- Новелсофт ХХК
- Могул Сервис энд Саппорт ХХК
- Дижитал Воркс ХХК
- Дижитал Повер ХХК
- Могул Экспресс ХХК

компаниуд орно. ”Новелсофт” ХХК-ийн бизнесийн үйл ажиллагаа өргөжсөнөөр ”Nomadic Software Solution” ХХК компанийг 2023 оны хаврын улирал үүсгэн байгуулсан.

2.2 Үйл ажиллагаа

”Новелсофт” ХХК нь захиалгат програм хангамж хөгжүүлэлт, дата аналитик, дата менежмент, мэргэжлийн үйлчилгээ (outsourcing) зэрэг үйл ажиллагаа явуулдаг байсан ба үүнээс програм хангамж болон дата аналитик гэх үйл ажиллагааны хүрээнд 2 хуваагдан ”Nomadic SS” ХХК бий болсон.

”Nomadic SS” ХХК нь одоогоор програм хангамж хөгжүүлэлт, мэргэжлийн үйлчилгээ (outsourcing) зэрэг үйл ажиллагааг явуулдаг.

2.3 Технологиуд болон системүүд

Компаний хувьд хөгжүүлэгдэж буй системүүд нь цогц системүүд байдаг ба дийлэнх хөгжүүлэгчдийн туршлага, системийн зохиомжоос хамааран

- Front-end
 - Mobile: Flutter - BLoC (Business Logic Components)
 - Web: Angular - Primeng
 - Desktop: .NET
- Server
 - Java - JHipster Spring boot
 - OAuth - Keycloak
 - Cassandra
 - Kafka
 - Redis
 - Prometheus

2.3. ТЕХНОЛОГИУД БОЛОН СИСТЕМҮҮД БҮЛЭГ 2. БАЙГУУЛЛАГЫН ТУХАЙ

технологиудыг best practice болгон мөрдөж ашигладаг. Хэрэглэгчийн шаардлагаас үүдэн өөр технологиудыг ашигласан тохиолдлууд ч байдаг.

3. НООМЕ ТӨСӨЛ

Үйлдвэрлэлийн дадлагын хүрээнд одоо идэвхитэй явагдаж буй "Нооме" төсөл дээр ажилласан бөгөөд энэхүү төсөл нь Нооме мобайл аппликейшн, Нооме сөх веб, Нооме контор веб зэрэг системүүдээс бүрдэх цогц систем юм. Одоогоор бүртгэлтэй 24000 хэрэглэгч байгаа ба тэдгээрийн 13000 нь идэвхитэй хэрэглэгч.

Бизнесийн үйл ажиллагаа нь голчлон хэрэглэгчдийн өдөр тутмын амьдрал дээр тулгуурласан бөгөөд СӨХ-өөс гадна машины зогсоолын хэсгийг нэвтрүүлээд байгаа билээ. Зогсоолын хэсэг нь бие даасан систем бөгөөд CCP(Cloud Car Parking) гэх төслийн хүрээнд идэвхитэй хэрэгжиж байгаа болно. Уг системийг "Нооме" мобайл аппликейшнд feature байдлаар оруулж өгсөн байгаа.

"Novelsoft" ХХК-ийн бүтээгдэхүүн болох "Homebook" СӨХ-ийн системийг сайжруулж сошиал коммунити аппликейшн болгож 2022 онд улмаар "Нооме" гэх шинэ төслийг эхлүүлсэн.

3.1 Хотхон

"Нооме" платформын хотхоны систем нь СӨХ-ийн менежмент, төлбөр, оршин суугчдын бүх төрлийн харилцааг удирдах тусгай систем болон түүнийг иргэдэд хүргэх "Нооме" сошиал аппын цогц бөгөөд

- СӨХ-ийн төлбөр бодолт
- Хэрэглэгчийн төлбөр төлөлт
- Автомат иБаримт гаргах, илгээх
- Тайлан гаргах
- Хотхоны бүлгэм үүсгэх, сошиал пост, чат

- Оршин суугчдын жагсаалт, мэдээлэл, автомат бүртгэл
- Зогсоол, агуулах удирдлага

зэрэг функцуудээс бүрдэнэ.

3.2 Контор

”Нооме” платформын хотхоны систем нь конторын төлбөр бодолт, үйлчилгээний захиалга авах гэх мэт бүх үйл ажиллагааг удирдах, системтэй.

- Бүх хэрэглэгчийн жагсаалт
- Конторын төлбөр бодолт
- Хэрэглэгчийн төлбөр төлөлт
- Тайлан гаргах
- Тоолуурын заалт
- Тариф удирдлага тохиргоо
- Автомат иБаримт гаргах, илгээх

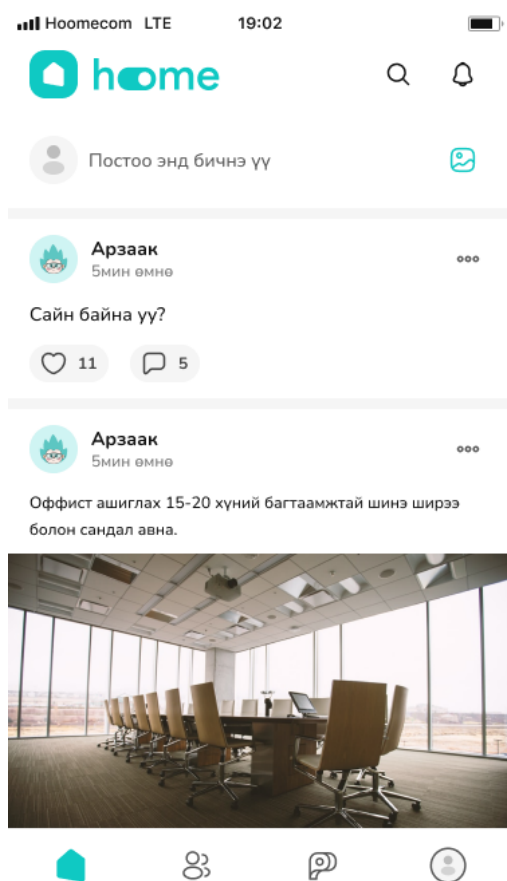
зэрэг функцуудээс бүрдэнэ.

3.3 Мобайл аппликейшн

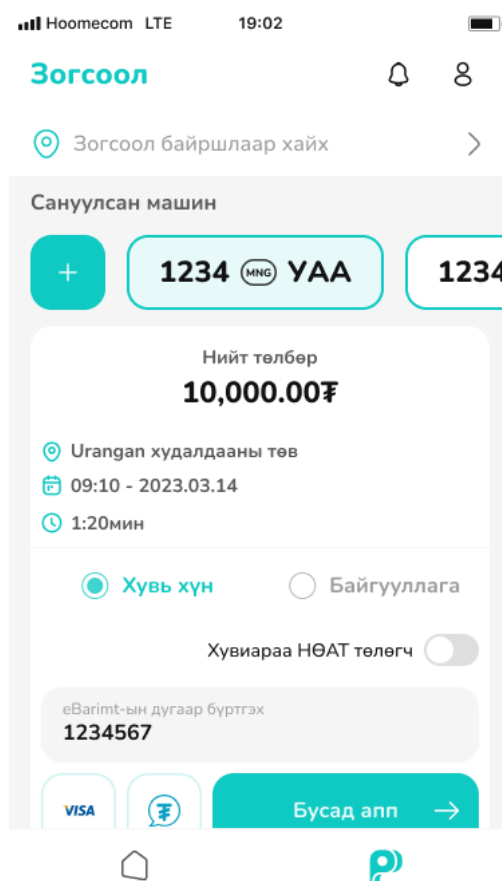
”Нооме” платформын мобайл аппликейшн нь СӨХ-ийн төлбөр төлөгч болон Нооме-д бүртгэлтэй зогсоолд машинаа тавьсан хэрэглэгчдэд зориулагдсан. Хэрэглэгч өөрийн хотхоны СӨХ-д элссэнээр сар бүрийн төлөх зардлууд болон тооцоог нэг дороос харах боломжтой болно. Зогсоолын төлбөр буюу ”Cloud Car Parking” систем нь Нооме-д бүртгэлтэй зогсоолууд дээр машин тавьсан хэрэглэгчийг төлбөрийг орсон хугацаанаас нь бодож тооцдог бөгөөд давхар

orgware буюу хаалгач хүнийг ажиллуулдаг билээ. Тухайн зогсоолын төлбөрийг зөвхөн "Нооме" биш, зарим тохиолдолд "Токи" аппликейшнийн хэрэглэгчид аппликейшнээрээ дамжуулан төлбөрөө төлдөг бөгөөд энэ нь Нооме-ССР-ийн зэрэгцээ орших ижил төстэй систем болно.

Энэхүү асуудлыг шийдэхээр шийдэл дэвшүүлж дадлагын хугацаанд "Нооме" платформын front-end болон back-end хөгжүүлэлтүүдийг хийж гүйцэтгэсэн билээ.



Зураг 3.1: Нооме - Коммунити нүүр.



Зураг 3.2: Нооме - Cloud Car Parking нүүр.

4. ТЕХНОЛОГИЙН СУДАЛГАА

4.1 BLoC

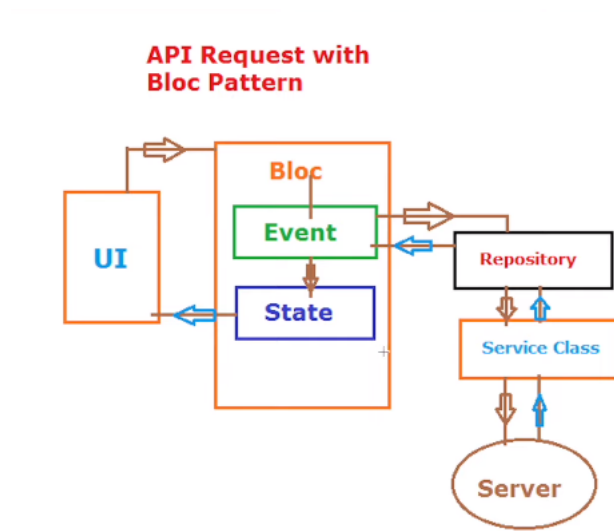
”Hoome” платформын мобайл аппликейшн нь програмчлалын Dart хэл буюу Flutter технологийг ашиглан бичигдсэн бөгөөд түүн дотроо төлвийн менежмент сан болох BLoC(Business Logic Components)-ийг ашигладаг.

BLoC нь хэрэглэгчийн интерфэйсийг бизнес логигоос тусгаарлаж өгөх зорилготой сан бөгөөд event-driven архитектур дээр суурилсан байдаг.

BLoC нь таны Flutter програмын төлвийг удирдах, хэрэглэгчийн харилцан үйлчлэлийг бүтэцтэй байдлаар зохицуулах design pattern юм. Энэ нь апп доторх data stream болон удирдахын тулд event, stream-ийн тухай ойлголтыг ашигладаг.

Flutter BLoC технологи нь:

1. **Events:** Event нь хэрэглэгчийн аппликейшнд үзүүлэх ямарваа нэгэн хариу үйлдэл бөгөөд тухайн event-ийг BLoC компонент хүлээн авч, логик үйлдлүүдийг хийж боловсруулснаар одоо байгаа төлвүүдийг шинэчилж шинэ төлвийг үүсгэдэг.
2. **BLoC Component:** Event-үүдийг сонсож, утга хүлээн авахад үргэлж бэлэн байдаг бөгөөд event-үүд нь өөрийн утгатай байх боломжтой. Хүлээн авсан эвентүүдийг хэрэглэгчийн интерфэйст ашиглагдаж буй төлөв, төлвийн өгөгдлийг шинэчлэхэд ашигладаг.
3. **Streams:** Event-үүд нь ихэвчлэн stream байдлаар хадгалагдаж, орсон дарааллаараа ачаалладаг бөгөөд BLoC компонентүүд нь тус бүр өөрийн зааж өгсөн урсгалыг сонсож байдаг.
4. **States:** Аппликейшн дээр ашиглагдаж буй бүх төрлийн датаг state буюу төлөв гэж нэрэлж байгаа бөгөөд тухайн төлөв нь BLoC компонентоор дамжуулагдан шинэчлэгдэж, улмаар хэрэглэгчийн интерфэйст өөрчлөлт ороход хүргэдэг.



Зураг 4.1: BLoC

Эдгээр 4 үндсэн элементүүдээс бүрдэх бөгөөд хэрэглэгчийн интерфейст шинэчлэл хийх, өөрчлөлт оруулах үед интерфесийн логигоос хамааран онцгой тохиолдол(exception), алдаа(runtime error)-наас сэргийлэх боломжтой. Аппликейшнийн usability болон scalability-г илүү амар хялбар байдлаар хангаж өгдөг сан юм.

5. УРАМШУУЛЫН СИСТЕМ

6. ИШЛЭЛ, ЗҮҮЛТ

6.1 Ишлэл

Ашигласан материал эсвэл номзүйг бичвэр тодор ишлэхдээ `cite` командаар заалтыг нь оруулна. Үүний тулд энэ хуудасны хамгийн доор байгаа *Ашигласан материал, ном зүй хэсэгт* `bibitem` командыг нэмнэ.

Жишээ нь: `bibitem{image1}` Гарчиг, Зохиогчдын нэр, хэвлэсэн он, хэвлэсэн газар

Дээрх жишээнд `image1` гэдэг нь ишлэх нэр. Доод талын мөрөнд нь байгаа дарааллын дагуу ашигласан материалыг бичнэ.

Ишлэхдээ `cite` командад ишлэх нэрийг дамжуулж өгнө. Жишээ нь `cite{image1}`.

6.2 Зүүлт

Зүүлтийг `footnote` командаар оруулна ¹.

¹Энэ холбоосоос зүүлтийн талаар дэлгэрэнгүй унш: <https://www.sharelatex.com/learn/Footnotes>

7. ЗУРАГ

Зураг оруулахдаа `includegraphics` командыг ашиглана. Доорх жишээнд `figure01.png` гэдэг нь зургийн файлын нэр бөгөөд өргөтгөлийг заавал бичих шаардлагагүй. Зургийн файл нь `main.tex` файлтай нэг фолдерт байх шаардлагатайг анхаарна уу! Дэлгэрэнгүйг [1]-с үз.

ShareLaTeX



Learn

7.1 Зургийн хэмжээ өөрчлөх

Хэмжээг томруулахдаа 0-1 хооронд утга ашиглана. Хэрэв 2 гэвэл 2 дахин томроно.

```
includegraphics[scale=0.5]{figure01}
```

ShareLaTeX



Learn

Өндөр өргөнийг шууд зааж өгч болох бөгөөд дөрвөлжин хаалтан дотор доорх байдлаар бичнэ.

```
includegraphics[width=3cm, height=4cm]{figure01}
```

ShareLaTeX



Learn

7.2 Зураг эргүүлэх

Зургийн эргүүлэхдээ `angle` параметрт эргүүлэх өнцгийн хэмжээг өгнө.

```
includegraphics[width=3cm, height=4cm, angle=45]{figure01}
```



7.3 Зургийн нэр

Зургын нэрийг `begin{figure}` хооронд `includegraphics` командтай хамт оруулна Зураг 7.1-ыг хар.

Энд зургийн нэрээс гадна `label`-ийг давхар бичиж өгөх шаардлагатай ба энэ нь зургийн дугаараар заалт хийхэд ашиглана. Жишээ нь: Зураг 7.2



Зураг 7.1: Зураг голлуулах

7.4 Зураг голлуулах

Зургийг голлуулахдаа `includegraphics` командын өмнө `centering` командыг бичээд `reflectbox` командыг `includegraphics` болон `caption` командуудад үйлчлэхээр оруулна.



Зураг 7.2: Зургийн нэрийг энд бичнэ

7.5 Зургийн чанар

LaTeX-т зургийг вектор форматаар (`svg`, `eps`) оруулбал хэвлэх болон томруулж харахад зургийн чанар алдагдахгүй. Тиймээс аль болох вектор зураг оруулж өгвөл зүгээр.

8. ХҮСНЭГТ ОРУУЛАХ

Хүснэгт оруулахад `tabular` командыг ашигладаг [5].

Table 8.1: Хүснэгтийн нэр. Хүснэгтийн нэр хүснэгтийн дээд талд байрлана.

Багана1	Багана2	Багана3	Багана4	Багана5
өгөгдөл	<i>өгөгдөл1</i>			

8.1 Хүснэгт зурах хэрэгсэл

Цэвэр LaTeX кодоор Хүснэгт үүсгэхэд харьцангуй төвөгтэй байдаг учир хялбар хэрэгслийг ашиглаж болно.

Тухайлбал <https://www.tablesgenerator.com/> холбоосруу орж хүснэгтийг визуал орчинд зураад үүсгэж өгсөн LaTeX кодыг энд хуулж оруулна.

9. КОД БА АЛГОРИТМ ОРУУЛАХ

Код оруулахдаа `begin{lstlisting}` ... `end{lstlisting}` командын хооронд бичнэ.

```
1 #include <stdio.h>
2 #define N 10
3 /* Block
4  * comment */
5 int main()
6 {
7     int i;
8     // Line comment.
9     puts("Hello_world!");
10    for (i = 0; i < N; i++)
11    {
12        puts("LaTeX_is_also_great_for_programmers!");
13    }
14    return 0;
15 }
```

Код 9.1: C хэлний кодын жишээ

Мөн кодын эх файлыг шууд оруулж ирж болох бөгөөд доорх командыг бичнэ.

```
1     puts("Hello_world!");
2
3     for (i = 0; i < N; i++)
4     {
5         puts("LaTeX_is_also_great_for_programmers!");
6     }
```

Код 9.2: Кодын файлаас хэсэгчилж оруулах

Мэдээллийн технологи, програм хангамжийн ажлын тайланд алгоритмыг хийсвэр кодын бичиглэлээр оруулах шаардлага гардаг. Дараах жишээгээр (Алгоритм 1) хийсвэр кодоор хэрхэн бичиж болохыг харуулав. Мөн бичвэр дотроо алгоритмд ашиглаж байгаа *parentId* хувьсагчийг дурдаж бичиж болдог.

Алгоритм 1 Даалгавар үүсгэх алгоритм

```

1: function TRAVERSE(parentId)                                ▷ parentId–эцэг ойлголтын дугаар
2:   children ← GETCHILDCONCEPTIDS(parentId)
3:   childCount ← children.count
4:   if childCount == 0 then
5:     return
6:   end if
7:   for i = 0 to childCount do
8:     GENERATETASK(childreni)                                ▷ Орчуулгын даалгавар үүсгэх
9:   end for
10:  for i = 0 to childCount do
11:    TRAVERSE(childreni)
12:  end for
13: end function

```

Дүгнэлт

Дүгнэлтийг энд бич

Bibliography

- [1] Inserting Images, Share LaTeX, https://www.sharelatex.com/learn/Inserting_Images
- [2] Paragraphs and new lines, Share LaTeX, https://www.sharelatex.com/learn/Paragraphs_and_new_lines
- [3] Bold, italics and underlining, Share LaTeX, https://www.sharelatex.com/learn/Bold,_italics_and_underlining
- [4] Lists, Share LaTeX, <https://www.sharelatex.com/learn/Lists>
- [5] Tables, Share LaTeX, <https://www.sharelatex.com/learn/Tables>

А. ШИНЖИЛГЭЭ ЗОХИОМЖ

Хавсралтын агуулга

В. КОДЫН ХЭРЭГЖҮҮЛЭЛТ

```
1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     #compute the bitwise xor matrix
10    M1 = bitxormatrix(genl1)
11    M2 = np.triu(bitxormatrix(genl2),1)
12
13    for i in range(m-1):
14        for j in range(i+1, m):
15            [r,c] = np.where(M2 == M1[i,j])
16            for k in range(len(r)):
17                VT[(i)*n + r[k]] = 1;
18                VT[(i)*n + c[k]] = 1;
19                VT[(j)*n + r[k]] = 1;
20                VT[(j)*n + c[k]] = 1;
21
22            if M is None:
23                M = np.copy(VT)
24            else:
25                M = np.concatenate((M, VT), 1)
26
27            VT = np.zeros((n*m,1), int)
28
29    return M
```