

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

Пүрэвийн Баасанхүү

**ПРОГРАМ ХАНГАМЖИЙН ИНЖЕНЕР
ХӨТӨЛБӨРИЙН ҮЙЛДВЭРЛЭЛИЙН
ДАДЛАГЫН ТАЙЛАН
(Веб хөгжүүлэгч)**

Программ хангамж (D061302)
Бакалаврын судалгааны ажил

Улаанбаатар

2023 оны 01 сар

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

ПРОГРАМ ХАНГАМЖИЙН ИНЖЕНЕР ХӨТӨЛБӨРИЙН
ҮЙЛДВЭРЛЭЛИЙН ДАДЛАГЫН ТАЙЛАН
(Веб хөгжүүлэгч)

Программ хангамж (D061302)
Бакалаврын судалгааны ажил

Удирдагч:	_____	Г. Мөнхзаяа
Хамтран удирдагч:	_____	И.Бямбажав
Гүйцэтгэсэн:	_____	П. Баасанхүү (19B1NUM1461)

Улаанбаатар
2023 оны 01 сар

Зохиогчийн баталгаа

Миний бие Пүрэвийн Баасанхүү ”ПРОГРАМ ХАНГАМЖИЙН ИНЖЕНЕР ХӨТӨЛБӨРИЙН ҮЙЛДВЭРЛЭЛИЙН ДАДЛАГЫН ТАЙЛАН” сэдэвтэй судалгааны ажлыг гүйцэтгэсэн болохыг зарлаж дараах зүйлсийг баталж байна:

- Ажил нь бүхэлдээ эсвэл ихэнхдээ Монгол Улсын Их Сургуулийн зэрэг горилохоор дэвшүүлсэн болно.
- Энэ ажлын аль нэг хэсгийг эсвэл бүхлээр нь ямар нэг их, дээд сургуулийн зэрэг горилохоор оруулж байгаагүй.
- Бусдын хийсэн ажлаас хуулбарлаагүй, ашигласан бол ишлэл, зүүлт хийсэн.
- 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. БАЙГУУЛЛАГЫН ТАНИЛЦУУЛГА	3
1.1 Товч танилцуулга	3
1.2 Ямар үйлчилгээ үзүүлдэг вэ?	3
1.3 Миний ажилласан систем?	4
2. СИСТЕМИЙН ШИНЖИЛГЭЭ.....	5
2.1 Хэрэглэгчийн шаардлага	5
2.2 Use Case Diagram.....	6
3. АШИГЛАХ ТЕХНОЛОГИ	7
3.1 Git.....	7
3.2 CSharp.....	8
3.3 React (JavaScript library)	10
3.4 MS SQL Server	11
4. ХЭРЭГЖҮҮЛЭЛТ	12
4.1 Capex.mn-ий мэдээний демо төсөл	12
5. ДҮГНЭЛТ	40
5.1 Үр дүн capex.mn-ий мэдээний демо төсөл	40
6. ҮР ДҮНГИЙН ТАЙЛАН	44
НОМ ЗҮЙ	45
ХАВСРАЛТ	46
А. УДИРДАГЧИЙН ҮНЭЛГЭЭ	46
В. КОДНЫ ХЭРЭГЖҮҮЛЭЛТ	47

ЗУРГИЙН ЖАГСААЛТ

2.1	Capex system Use Case	6
3.1	Git-н ажиллах процесс	7
4.1	CSharp Арын хөгжүүлэлт	13
4.2	ReactJS Нүүрэн талын хөгжүүлэлт	28
5.1	Мэдээний мессэжүүд	40
5.2	Мэдээ рүү хэрэглэгч нэвтрэх	40
5.3	Мэдээний Header хэсэг	41
5.4	Мэдээ оруулах	41
5.5	Мэдээ нэмэх	41
5.6	Мэдээ хадгалагдсан байдал	42
5.7	Мэдээнүүд	42
5.8	Мэдээний Footer хэсэг	42
5.9	Postman ашиглаж API-тэй харьцах	43
A.1	Удирдагчийн үнэлгээ	46

ХҮСНЭГТИЙН ЖАГСААЛТ

1	Дадлагын төлөвлөгөө	2
---	----------------------------------	---

Кодын жагсаалт

3.1	CSharp хэлний жишээ	9
3.2	JSX ашиглаж "Comp" нэртэй функц -н html элемент буцаах компонент	10
3.3	JSX ашиглаж "Comp" нэртэй класс -н html элемент буцаах компонент	11
4.1	CustomErrorMessages.cs Буцаах мессэжүүд	13
4.2	Json.cs Хүсэлт илгээхэд буцаах датаг хэлбэржүүлнэ	14
4.3	Database.cs өгөгдлийн сантай харьцах код	14
4.4	NewsController.cs -р ирсэн хүсэлтийг (GET, POST) хүлээж аваад боловсруулалт хийх	15
4.5	NewsManager.cs өгөгдлийн санаас авч боловсруулж буцаах хэсгийг энэд агуулна	21
4.6	Хэрэглэгч нэвтэрсэн эсхээс хамаарч ажиллах компонент	27
4.7	Хэрэглэгчийн форм бөглөх компонент	28
4.8	Мэдээний Header хэсэг	32
4.9	Мэдээний Контент хэсэг	33
4.10	Мэдээ нэмэх хэсэг	34
4.11	Мэдээ засах хэсэг	37

УДИРТГАЛ

Миний бие П. Баасанхүү нь үйлдвэрлэлийн дадлага хийх байгууллагаар "Даталанд технологи ХХК"-ийг сонгосон бөгөөд

Уг компанид дадлага хийх хугацаанд React, C Sharp, MS MSSQL SERVER, POSTMAN гэсэн технологиуд дээр голчлон ажилласан. Дадлагын хугацаанд байгууллагын хэрэгжүүлж буй "https://www.sarex.mn дижитал хөрөнгийн бирж"-ын программын ажлын даалгавар боловсруулах төсөлд "back-end" хөгжүүлэлтэд оролцож ажилласан бөгөөд C Sharp, Reactjs ашиглан алдааг засах сайжруулах таскуудыг хийж гүйцэтгэсэн. Мөн байгууллагын мөрдөж ажилладаг арга барилууд болох Agile хөгжүүлэлт болон түүний фрэймворк болох Scrum аргачлал гэх мэт олон зүйлсийг судалж практикт хэрэгжүүлэхийг зорьсон юм.

Table 1: Дадлагын төлөвлөгөө

№	Гүйцэтгэх ажил	Хугацаа 1 сарын	Биелэлт	Дадлагын удирдагчийн үнэлгээ
1	Компанитэй танилцах, хөгжүүлэлтийн орчноо бэлдэх (frontend, backend) ажиллуулах, судлах	03–05		
2	Capex.mn Сайтын хэрэглэгчийн хэтэвч хэсэгт биржүүдийн Balance нь 0 дата-г дамжуулж байгааг шийдэх	06		
3	Хэтэвчинд орлогын хэсэгт алдаа гарсан судалж шийдэх	09-11		
4	Мэдээний хэсэгт Login эрхтэй болгох. Энэ нь Capex.mn -ий эрхээр нэвтэрнэ	11-13		
5	Мэдээ рүү нэвтрэх үед Token mail рүү илгээдэг болгох	16		
6	React Toast ашиглаж бүх үфзэгдэл хийх бүрт хэрэглэгчдээ мессэж өгч сурах(Мэдээ хадгаллаа, Алдаа гарлаа ..)	16-17		
7	Токен хадгалж Reload хийхэд гардаггүй болгох	17		
8	Нэвтрэх, Мэдээний хэсгийн хэлбэржүүлэлт сайжруулал	18-19		
9	Бүх CSS- module css-тэй болгосон давуу	20		
10	Мэдээг Header, Footer, Navigation-тэй болгох	23		
11	Кодоо бүтэцлэх, Logout-тэй болгох	23		
12	Сервэрээс хамаарч мессэжээ зөв өгөх	24		
13	Test хийх (Capex.mn сайтын Favority хэсэгт алдаа гарсан!) GET,POST Судлах	25		
14	DB server-тэй холбогдох, базтай харьцаж сурах, Mobile- д тохируулж Translation API бичих	26		
15	Өөр өгөгдлийн сангаас SocialLinks - γ_2 CoinName-р шүүж авах мөн Тестлэлийн өгөгдлийн санг өгөгдөлтэй болгох	27		
16	Арилжаануудын ханшийг MNT рүү шилжүүлэх үед	27		

1. БАЙГУУЛЛАГЫН ТАНИЛЦУУЛГА

1.1 Товч танилцуулга

”ДАТАЛАНД ТЕХНОЛОГИ ХХК”-ий хамт олон нь өөрсдийн мэдээллийн технологийн салбарт хуримтлуулсан туршлага, ур чадварууд дээрээ суурилан 2019 онд үүсгэн байгуулагдаж, программ хангамж, системийн интеграцлал болон техникийн шийдлийн чиглэлээр Монгол улсын мэдээллийн технологийн зах зээлд идэвхтэй үйл ажиллагаа явуулж буй компани юм.

ДАТАЛАНД ТЕХНОЛОГИ ХХК нь Виртуал хөрөнгийн үйлчилгээ үзүүлэгч дижитал бирж болон төсвийн нэгдсэн систем зэрэг платформуудыг олон улсын стандартад нийцүүлэн хөгжүүлж байгаа компани. Уг компани нь үйл ажиллагаандаа шударга ёс, хууль дүрмийг эрхэмлэн үйлчлүүлэгч, хэрэглэгч бүрийн хэрэгцээ шаардлагыг хангасан чанартай бүтээгдэхүүн үйлчилгээ үзүүлэхийн тулд ажилдаа чин сэтгэлээсээ хандаж өөрийгөө хөгжүүлэн хурдан суралцаж идэвх санаачилгатай ажиллах ажил хэрэгч, халуун дулаан уур амьсгалтай баг хамт олныг бүрдүүлэн ажилладаг. Үүсгэн байгуулагдсанаасаа хойш богино хугацаанд уг компаний хамт олон нь иргэдэд зориулсан цаг хугацаа, орон зайнаас үл хамааран хууль эрх зүйн зөвлөгөө, мэдээллийг хуулийн технологид суурилсан ухаалаг платформоор дамжуулан авах боломжтой “iLaw-таны цахим хуульч” үйлчилгээний хөгжүүлэлт, шийдлийг амжилттай гүйцэтгэж олон нийтэд хүргэж буй компани.

1.2 Ямар үйлчилгээ үзүүлдэг вэ?

ДАТАЛАНД ТЕХНОЛОГИ ХХК нь үүсгэн байгуулагдсанаасаа хойш богино хугацаанд уг компаний хамт олон нь иргэдэд зориулсан цаг хугацаа, орон зайнаас үл хамааран хууль эрх зүйн зөвлөгөө, мэдээллийг хуулийн технологид суурилсан ухаалаг платформоор дамжуулан авах боломжтой “iLaw-таны цахим хуульч” үйлчилгээний хөгжүүлэлт, шийдлийг амжилттай

гүйцэтгэж олон нийтэд хүргэж буй компани.

1.3 Миний ажилласан систем?

Би Сарех системийн арын хэсэг болох хэтэвч болон арилжаа, launchpad модулиуддээр ажилласан бөгөөд энэхүү систем нь хэрэглэгчийн хувийн мэдээллийг цуглуулах, ашиглах, дамжуулах, хамгаалах зэргийг багтаасан байна. Үйлчилгээ үзүүлэгч нь дижитал хөрөнгийн биржээр үйлчилгээ үзүүлэхэд бүртгүүлсэн бүх хэрэглэгчдийн хувийн нууцлалыг энэхүү баримт бичигт зааснаар хадгалж, хамгаалж дамжуулдаг.

Жишээлбэл: Бүртгүүлэх, орлого, зарлагын гүйлгээ хийх, арилжаа хийх, санал асуулгад оролцох, олон нийтийн хуудаснуудад мэдээ нийтлэх эсхүл бусад хэлбэрээр харилцах үед таны мэдээллийг цуглуулдаг.

2. СИСТЕМИЙН ШИНЖИЛГЭЭ

2.1 Хэрэглэгчийн шаардлага

2.1.1 Хэрэглэгчид

- Админ, үйлчилгээ авах хэрэглэгч

2.1.2 Функциональ хэрэглэгчийн шаардлагууд

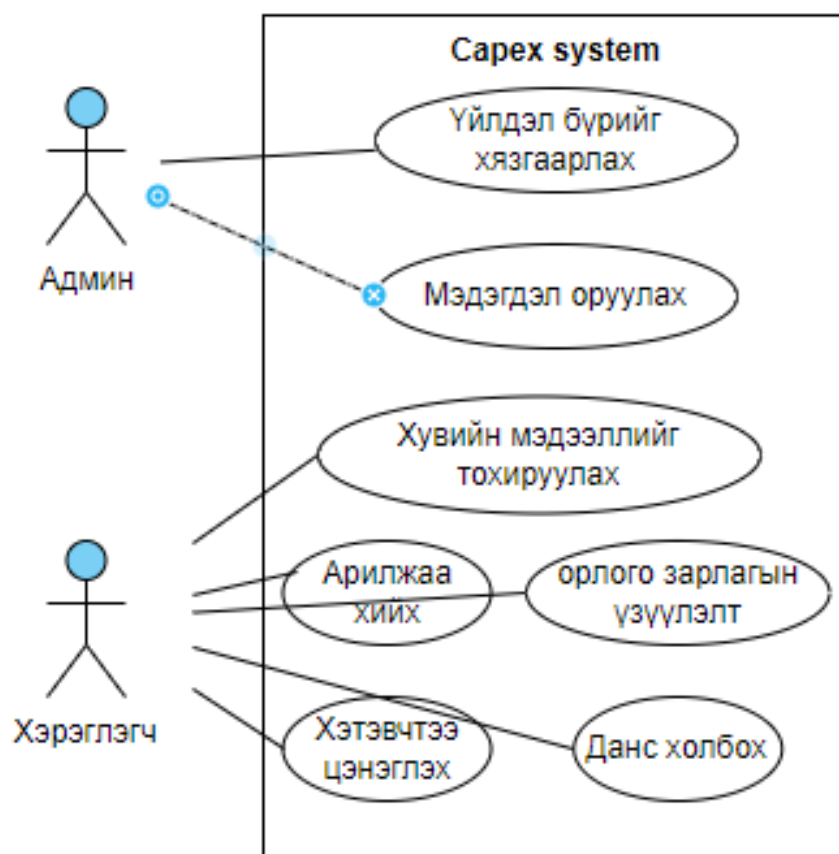
- /ФШ01/- Бүртгүүлсэн хэрэглэгчид удирдлагын аппыг ашиглахад хялбар байх.
- /ФШ02/-Хэрэглэгч нь өөрийн бүртгэл болон цахим хэтэвчээр дамжуулан дижитал хөрөнгийн бирж дээрх бүх төрлийн зөвшөөрөгдсөн виртуал хөрөнгө арилжиж, хадгалж, бусдад шилжүүлэх боломжтой
- /ФШ03/-Бүртгүүлэгч бүр өөрийн данстай хэтэвчтэй байх
- /ФШ04/-Хэрэглэгч нэвтрээгүй байсан ч Chart буюу арилжааны ханшийг харуулдаг байх
- /ФШ05/-Мөн хэрэглэгчид Favorite(Од)-г хүссэн биржээрээ дарж хадгалах боломжтой
- /ФШ06/-Үйлчилгээний нөхцөлийг бүх хуудсанд харагдуулдаг байх
- /ФШ07/-Бирж дээрх хөрөнгөө бусад биржүүдэд хөрвүүлэх

2.1.3 Функционал бус хэрэглэгчийн шаардлагууд

- /ФБШ01/-Цэс responsive байх
- /ФБШ02/-Бүх хуудасны арын суурь өнгө ижил байх
- /ФБШ03/-Арилжаа хийхэд хялбар байх

- /ФБШ04/-Системийн аюулгүй болон найдвартай байдлыг хангадаг байх
- /ФБШ05/-Систем баталгаажуулах мэдээлээ үнэн зөв оруулахад баталгаажуулах нэг удаагийн код авдаг байх
- /ӨФБШ06/-Бүх хуудас тусламж авах хэсэгтэй байх

2.2 Use Case Diagramm

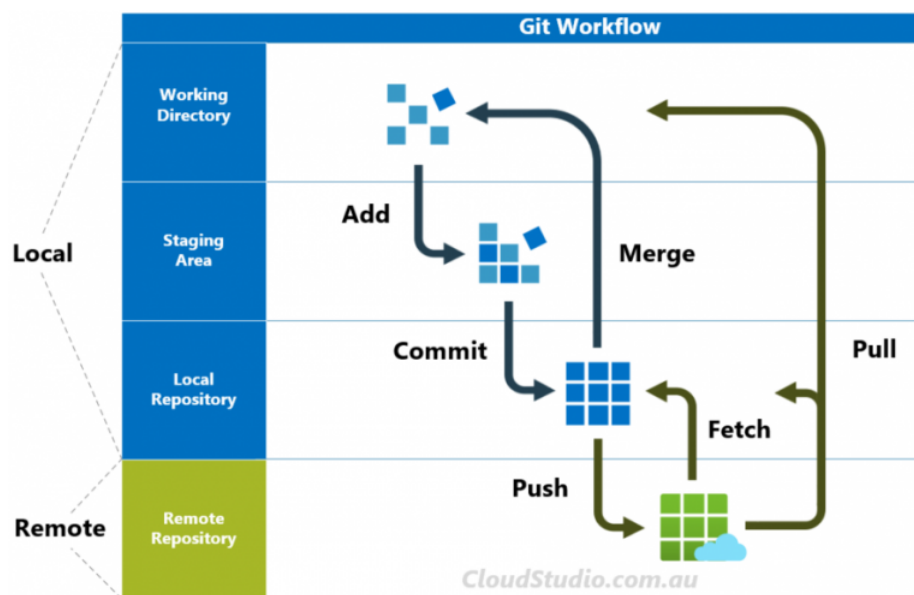


Зураг 2.1: Capex system Use Case

3. АШИГЛАХ ТЕХНОЛОГИ

3.1 Git

Энэ технологийг уг компани багаар ажиллах болон бичсэн кодын өөрчлөлтийг хянах зорилгоор ашигладаг. Git-ийг 2004 онд Linus Torvalds зохиосон бөгөөд бид түүнийг алдарт Linux үйлдлийн системийг бүтээсэн гэдгээр нь мэднэ. Тухайн үед Linux-ийн хөгжүүлэлтийн шаардлагыг бүрэн хангах систем олдохгүй байснаас болж git-ийг бүтээсэн гэдэг. **GIT гэж юу вэ?** Git гэдэг бол программын сорс код болон ямар нэг файлд орсон өөрчлөлтийн түүхийг бичиж үлдээх зорилготой тархсан хувилбар удирдах систем юм. Linux Kernel-н сорс кодыг удирдах зорилгоор Linux Kernel-г хөгжүүлсэн. Түүнээс хойш маш олон том том прожекторууд GIT хувилбар удирдах системийг ашиглан хөгжиж байна.



Зураг 3.1: Git-н ажиллах процесс

Git ашиглаж буй хэрэглэгч бүр өөрийн ажлын явцад бүх түүхийг агуулсан байх ба үндсэн Repository-тай яг адилхан Repository-г өөр дээрээ үүсгэнэ. Тийм учраас сүлжээний асуудал

зэргээс хамаарч үндсэн Repository-д хандаж чадахгүй орчин байлаа ч түүхийг харах, өөрчлөлт оруулах зэрэг ихэнх ажлаа хийх боломжтой байдаг. **GIT-н давуу тал** нь компьютер дээрээ засвар болон түүхийг агуулсан үндсэн хуудсыг хуулаад авчих боломжтой юм. Энэ нь өөрөөр хэлбэл үндсэн фолдерийг өөр дээрээ хуулаад чөлөөтэй засвар хийж болно гэсэн үг юм. Remote-Repository-г clone хийж аваад Local-Repository үүсгээд үүн дээрээ засвар өөрчлөлтүүдээ хийгээд явна гэсэн үг юм.

3.2 CSharp

CSharp нь ISO стандартаар батлагдсан Microsoft-ийн хөгжүүлсэн хамгийн алдартай програмчлалын хэлнүүдийн нэг юм. CSharp хэл нь ерөнхий зориулалттай, өндөр түвшний, бүрэн Объект хандалттай, мөн бүрэлдэхүүн хэсэг рүү чиглэсэн програмчлалын хэл байхаар бүтээгдсэн. Энэхүү өргөн хэрэглэгддэг програмчлалын хэлийг бүтээгч нь ноён Андерс Хейлсберг юм. Энэ хэл нь C++ болон Java дээр суурилсан бөгөөд өөр өөр Объект хандлагат болон бүрэлдэхүүн хэсгүүдэд чиглэсэн програмчлалын хэл.

CSharp-н давуу тал

- **Объект хандалтат хэл**

CSharp нь цэвэр объект хандалтат хэл бөгөөд энэ нь танд модульчлагдсан засвар үйлчилгээ хийх боломжтой програмууд болон дахин ашиглах кодуудыг үүсгэх боломжийг олгодог. Энэ бол CSharp хэлний C++ хэлээс хамгийн том давуу талуудын нэг юм.

- **Санах ой цоорхой**

CSharp нь системд эмх замбараагүй байдал үүсгэдэггүй.

- **Хөгжүүлэхэд хялбар**

Баялаг сантай бөгөөд олон функцийг хэрэгжүүлэхэд хялбар . CSharp нь дэлхийн ихэнх программистуудад нөлөөлдөг бөгөөд програмчлалын ертөнцөд өөрийн гэсэн түүхтэй.

- **Илүү сайн интеграцчилал**

.NET дээр бичигдсэн програмууд нь бусад NET Technologies-тэй илүү сайн уялдаж, харилцан ажиллах чадвартай

- **Илүү гаргацтай кодчилол**

Get-set методтой ингэснээр кодыг илүү ойлгомжтой болгодог

```
1 class Object
2 {
3     private string name;
4     public string Name
5     {
6         get { return name; }
7         set { name = value; }
8     }
9 }
10
11 class Main
12 {
13     static void Main(string[] args)
14     {
15         Object obj = new Object();
16         obj.Name = "Liam";
17         Console.WriteLine(obj.Name);
18     }
19 }
```

Код 3.1: CSharp хэлний жишээ

3.3 React (JavaScript library)

React бол веб болон мобайл апп хөгжүүлэхэд ашигладаг алдартай JavaScript сан юм. Үүнээс ReactJs нь веб хөгжүүлэх React Native нь апп хөгжүүлэх зориулалттай технологи юм. Өнөөгийн олон томоохон компаниуд (Netflix, Instagram, Facebook гэх мэт) мөн ReactJS ашигладаг. Энэ хүрээг ашиглах нь бусад фреймворкуудаас олон давуу талтай бөгөөд энэ нь сүүлийн хэдэн жилийн турш янз бүрийн хэлний зэрэглэлийн индексээр шилдэг 10 програмчлалын хэлний жагсаалтад багтаж байна. ReactJs нь **Model View Controller (MVC)** архитектурыг дагаж мөрддөг бөгөөд гар утас болон веб программуудыг удирдах үүрэгтэй. Энэхүү технологийг ашиглахын тулд үндсэн хэдэн ойлголтууд авах хэрэгтэй. Үүнд: **component, jsx, state, hooks, props** мөн хамгийн чухал зүйл болох **Virtual Dom** нар багтана. Компонент нь UI-ын хэсгүүдийн (**sidebar, dialog box, slider** гэх мэт) салган дахин хэрэглэгдэж болохуйц бие даасан хэсгүүд юм. JavaScript функц параметртэй байдаг бол компонент нь “props” гэж нэрийдсэн өгөгдөл/оролт(input) авч болдог, эргүүлээд React элемент буцаана. Компонентэд 5 API байдаг. State (өөрийнх нь хувьсагч, props-ыг дамжуулна), context (global хувьсагч шиг ямар нэг өгөгдлийг бүх компонент хэрэглэх шаардлагатай үед ашигладаг), props (JavaScript функцийн параметр), render (DOM-руу элементийг зурна, react элемент буцаана), lifecycle events (компонент зурагдлаа, шинэчлэгдлээ, DOM-с устгагдлаа гээд амьдралынх нь чухал үйл явдлууд)-ыг тус тус гүйцэтгэдэг юм. Контэйнерийн үүрэг нь өгөгдлийг API-с дуудан авчрах мөн харгалзах компонентыг рендер хийдэг. Presentational компонентууд нь ихэвчлэн props-г түлхүү ашигладаг, props-р функц(callback) өгөгдөл гэх мэт UI-д шаардлагатай бүхий л зүйлийг дамжуулах боломжтой.

```
1 import React from "react";
2 const Comp = (props) => {
3   return (<div>{props.value}</div>)
4 }
5 export default Comp
```

Код 3.2: JSX ашиглаж "Comp" нэртэй **функц**-н html элемент буцаах компонент

```

1 import React, { Component } from "react";
2 class Comp extends Component{
3     constructor(props) {
4         super(props);
5     }
6     render() {
7         return <h2>{this.props.value}</h2>;
8     }
9 }
10 export default Comp

```

Код 3.3: JSX ашиглаж "Comp" нэртэй **класс**-н html элемент буцаах компонент

JSX нь React дээр HTML бичих боломжийг олгодог.

3.4 MS SQL Server

MS SQL Server нь Microsoft-ын бүтээсэн өгөгдлийн сангийн удирдлагын систем (**RDBMS - Relational Database Management System**) юм. Энэ бүтээгдэхүүн нь бусад программын шаардлагын дагуу өгөгдлийг сэргээх үндсэн функцэд зориулагдсан бөгөөд үүнийг нэг компьютер дээр эсвэл сүлжээгээр өөр компьютер дээр ажиллуулж болно. Энэ нь өгөгдөл үүсгэх, сэргээх, нэвтрэх болон нөөцлөлт үүсгэх, зөвшөөрөл олгох гэх мэт SQL серверийн үндсэн болон дэвшилтэт ойлголтуудыг агуулсан байна.

4. ХЭРЭГЖҮҮЛЭЛТ

4.1 Сарех.mn-ий мэдээний демо төсөл

Дадлагын хугацаанд CSharp, MS SQL болон React судалж түүнийг бататгах сарех.mn сайтын мэдээ мэдээллийг оруулах засах гэсэн үйлдэлтэй демо ажлыг ахлахын тусламж зааврын дор хийж гүйцэтгэсэн юм.

Сурах ур чадвар:

- Ашиглаж буй технологио зөв хэрэглэж сурах
- Тайлбар бичиж сурах
- Кодоо цэвэрхэн бичих
- Хавтаслаж бичих (**Pages, Components, assets**)
- Хувьсагчийг тохиромжтой өгөх, Бусад функц, component хувьсагчдийн нэршлийг тохиромжтой өгөх (**Жишээ нь мэдээ: NewsSave(), NewsUpdate()**)
- Git ашиглаж сурах **Бичсэн кодоо push,pull хийх өөр laptop дээр ажиллахад clone хийж авах** гэх мэт
- Style бичихэд суралцах inline css-с зайлс хийж mobile css ашиглаж сурах
- Броузерийн Developer Tools-тэй харьцаж сурах. **Үүнд: network(сүлжээгээр ирж буй data-г шалгах), application (localstorage), console (тестлэл хийхэд тохиромжтой)** гэх мэт

Системийн шаардлага :

- Сарех-н эрхээр хэрэглэгч нэвтэрдэг байх

- Нэвтрэхдээ нэг удаагийн код авдаг байх
- Мэдээ нэмдэг байх
- Header, Content, Footer гэсэн бүтэцтэй байх
- Мэдээ засдаг байх
- Ямар нэгэн үйлдэл хийхэд мессежийг харуулдаг байх (Toast)

Мэдээний демо төслийн зохион байгуулалт. Уг код нь компаний нууцлалтай учир бичсэн хэсгээ шүүж авсан болно

```

> C# CustomErrorMessages.cs
> C# Database.cs
> C# Json.cs
> C# NewsController.cs
> C# NewsManager.cs

```

Зураг 4.1: CSharp Арын хөгжүүлэлт

Csharp backend хэрэгжүүлэлт => Компаний нууцлалтай холбоотой Кодыг бүрэн харуулаагүй болон

```

1 namespace Models
2 {
3     //
4     public class CustomErrorMessages
5     {
6         public string Exception_General = "Our servers are experiencing
7             some glitch, please try again later.";
8         public string Exception_BadRequest = "Invalid Payload";
9         public string Success_General = "Success!";
10    }
11 }

```

```

9      }
10 }

```

Код 4.1: **CustomErrorMessages.cs** Буцаах мессэжүүд

```

1 namespace Models
2 {
3     public class Json
4     {
5         public string status { get; set; }
6         public string message { get; set; }
7         public dynamic data { get; set; }
8         public Json(string _status, string _errorMessage, dynamic
           _data)
9         {
10             //JSON-r data butsaah
11             status = _status; //success OR error
12             message = _errorMessage; //messega -g ened avna
13             data = _data; //butsaasan data-g avna
14         }
15     }
16 }

```

Код 4.2: **Json.cs** Хүсэлт илгээхэд буцаах датаг хэлбэржүүлнэ

```

1 //Tuhain tbl_News - Ogogdliin Sand bah talbaruud
2 public partial class tbl_News
3 {
4     [Key]
5     [DatabaseGenerated(DatabaseGeneratedOption.Identity)]
6     public long ID { get; set; }

```

```

7      public string Author { get; set; }
8      public string? Title { get; set; }
9      public string? Sub_Title { get; set; }
10     public string? NewsData { get; set; }
11     public DateTime Create_Date { get; set; }
12     public bool? Status { get; set; }
13     public string? Banner_Img { get; set; }
14     public string? TypeList { get; set; }
15     public string? Actual { get; set; }
16     public string? ForeCast { get; set; }
17     public string? Previous { get; set; }
18 }

```

Код 4.3: **Database.cs** өгөгдлийн сантай харьцах код

```

1 namespace Controllers
2 {
3     //iim nertei route oghod dotrokh oghoghdluud GET PSOT-s khamaarch
4     ajillakh
5     [Route("api/news")]
6     [ApiController]
7     public class NewsController : ControllerBase
8     {
9         private readonly CustomErrorMessages customErrorMessages;
10        private readonly NewsManager newsManager;
11        public NewsController(CustomErrorMessages customErrorMessages,
12                               NewsManager newsManager)
13        {
14            this.customErrorMessages = customErrorMessages;
15        }
16    }
17 }

```

```

13         this.newsManager = newsManager;
14     }
15     //medeenii baganuudiig aguulsan Class uuniig obiekt bolgoj
        ashighlana
16     public class NewsEntry
17     {
18         public long ID;
19         public string Author;
20         public string Title;
21         public string Sub_Title;
22         public string NewsData;
23         public bool Status;
24         public string Banner_Img;
25         public string TypeList;
26         public string Actual;
27         public string ForeCast;
28         public string Previous;
29     }
30
31     // GET - eer api/news/getNewsList huselt duudahad buh medeegeh
        avah
32     [HttpGet]
33     [Route("/getNewsList")]
34     public async Task<Object> getNewsList()
35     {
36         try
37         {
38             /**

```

```

39         * GetNewsList() =>newsManager gesen file-s GetNewsList
           () damjuuln
40         * GetNewsList() => ene funct ni database-tei ajillaj
           amjiltta avbal medeenuudiig amjiltgui bol aldaa
           ogoh hariu medeegee butsaana
41         * Json(status, _errorMessage, data) butetstei
           obiektiig butsaana
42         */
43         return new Json("Success", nameof(customErrorMessages.
           Success_General), await newsManager.GetNewsList());
44
45     }
46     catch (Exception ex)
47     {
48         //aldaag ene hesegt barij avna
49         Console.WriteLine(ex.Message + " " + ex.StackTrace);
50         // Json(status, _errorMessage, data) buteststei obiektiig
           butsaana
51         return new Json("Error", nameof(customErrorMessages.
           Exception_General), null);
52     }
53 }
54
55 /**
56  * GET - eer api/news/getNewsListById huselt duudahad
           getNewsListById([FromQuery] long id) ene funct ajillana
57  * route/:id ingj damjuulj medeenuudees id-r shuuj avah
58  */

```



```

59     [HttpGet]
60     [Route("getNewsListById")]
61     public async Task<Object> getNewsListById([FromQuery] long id)
62     {
63         try
64         {
65             /**
66              * GetNewsByID() =>newsManager gesen file-s GetNewsByID
67              *              () damjuuln
68              * GetNewsByID() => ene funct ni database-tei ajillaj
69              *              amjiltta avbal medeenuudees id-r shuuj amjiltgui
70              *              bol aldaa ogoh hariu medeegee butsaana
71              * Json(status, _errorMessage, data) butetstei
72              *              obiektiig butsaana
73              */
74             return new Json("Success", nameof(customErrorMessages.
75                 Success_General), await newsManager.GetNewsByID(id))
76                 ;
77         }
78         catch (Exception ex)
79         {
80             //aldaag ene hesegt barij avna
81             Console.WriteLine(ex.Message + " " + ex.StackTrace);
82             // Json(status, _errorMessage, data) butestei obiektiig
83             // butsaana
84             return new Json("Error", nameof(customErrorMessages.
85                 Exception_General), null);
86         }
87     }

```

```

79     }
80
81     /**
82      * POST - oor api/news/saveNews  huselt duudahad saveNews(
83      NewsEntry request) ene funct ajillana
84
85      * request { NewsEntry buyu class-r uussen zagvariin daguu
86      data orj irne } uuniig databased shalgaj nemeh
87
88      */
89
90     [HttpPost]
91     [Route("saveNews")]
92     public async Task<Object> saveNews(NewsEntry request)
93     {
94         try
95         {
96             /**
97              * saveNews(request.Author, .... ) =>newsManager gesn
98              fileas saveNews-n parametereer utguudiig avah
99
100             * saveNews(request.Author,...) => ene funct ni
101             databasetei ajillaj medeeg amjiltta avbal nemeed
102             amjiltgui bol aldaa ogod hariu butsaana
103
104             * Json(status, _errorMessage, data) butetstei
105             obiektiig butsaana
106
107             */
108
109             return new Json("Success", nameof(customErrorMessages.
110                 Success_General), newsManager.saveNews(request.ID,
111                 request.Author, request.Title, request.Sub_Title,
112                 request.NewsData, request.Status, request.Banner_Img
113                 , request.TypeList, request.Actual, request.ForeCast
114                 , request.Previous));

```

```

96         }
97         catch (Exception ex)
98         {
99             //aldaag ene hesegt barij avna
100             Console.WriteLine(ex.Message + " " + ex.StackTrace);
101             // Json(status, _errorMessage, data) butestei obiektiig
102             butsaana
103             return new Json("Error", nameof(customErrorMessages.
104                 Exception_General), null);
105         }
106     }
107
108     /**
109     * POST - oor api/news/updateNews huselt duudahad updateNews(
110     NewsEntry request) ene funct ajillana
111     * request { NewsEntry buyu class-r uussen zagvariin daguu
112     data orj irne } uuniig databased shalgaj nemeh
113     */
114     [HttpPost]
115     [Route("updateNews")]
116     public async Task<Object> updateNews(NewsEntry request)
117     {
118         try
119         {
120             /**
121             * updateNews(request.Author, .... ) =>newsManager gesn
122             fileas updateNews-n parametereer utguudiig avah
123             * updateNews(request.Author,...) => ene funct ni

```

```

        databasetei ajillaj medeeg amjiltta avbal oorcloud
        amjiltgui bol aldaa ogod hariu butsaana
119     * Json(status, _errorMessage, data) butetstei
        obiektiig butsaana
120     */
121     return new Json("Success", nameof(customErrorMessages.
        Success_General), newsManager.updateNews( request.ID
        , request.Author, request.Title, request.Sub_Title,
        request.NewsData, request.Status, request.Banner_Img
        , request.TypeList, request.Actual, request.ForeCast
        , request.Previous));
122 }
123 catch (Exception ex)
124 { //aldaag ene hesegt barij avna
125     Console.WriteLine(ex.Message + " " + ex.StackTrace);
126     // Json(status, _errorMessage, data) butestei obiektiig
        butsaana
127     return new Json("Error", nameof(customErrorMessages.
        Exception_General), null);
128 }
129 }
130 }
131 }

```

Код 4.4: **NewsController.cs**-р ирсэн хүсэлтийг (**GET**, **POST**) хүлээж аваад боловсруулалт хийх

```

1 using Models;
2 namespace Services

```

```

3 {
4     public class NewsManager
5     {
6         //Scope interface
7         private readonly IServiceScopeFactory serviceScopeFactory;
8         // Ene ni custom aldaa butsaah zoriulalttai
9         private readonly CustomErrorMessages customErrorMessages;
10
11        public NewsManager(IServiceScopeFactory serviceScopeFactory,
12                           CustomErrorMessages customErrorMessages)
13        {
14            this.serviceScopeFactory = serviceScopeFactory;
15            this.customErrorMessages = customErrorMessages;
16        }
17
18        //Buh messegiig database-s shuuju avah funct
19        public async Task<Object> GetNewsList()
20        {
21            // Olon medee avahiin tuld List ashiglah heregtei
22            List<tbl_News> newsList;
23
24            // instance-g uusgeh uuregtei interface bogood ene ni
25            // ergeed IServiceProvider-n ashiglaltiin hugatsaag udirдах
26            // uuregtei
27            using (var scope = serviceScopeFactory.CreateScope())
28
29            // Database-daa holdogдох
30            using (var db = scope.ServiceProvider.GetRequiredService<

```

```

28         Database>())
29
30         //medeenuudiigh ogoghdliin sanghaas avahdaa status-n
31         idevhitei boghood hughatsaagh buurah erembeer shuuji
32         avah code
33
34         newList = db.tbl_News.Where(x => x.Status == true).
35         OrderByDescending(x => x.Create_Date).ToList();
36
37         //List ee butsaah
38
39         return new { news = newList };
40
41     }
42
43     //Ene funct ni medeenuudees ID-r shuuji avah funct
44     public async Task<Object> GetNewsByID(long id)
45     {
46
47         //neg medee avakh uchir zhaghsaalt uusghekh shaardlagagui
48         medeenii toroltei huvsagch zarlaad utga onooj butsaana
49         tbl_News newList;
50
51         //IServiceProvider-iin ashighlaltiin hughatsaagh udirdah
52         uureghtei
53
54         using (var scope = serviceScopeFactory.CreateScope())
55
56         // Database-daa holdogdoh
57
58         using (var db = scope.ServiceProvider.GetRequiredService<
59             Database>())
60
61         //medeeg ogoghdliin sanghaas avakhdaa id-r shuuji avah
62         newList = db.tbl_News.Where(x => x.ID == id).

```

```

FirstOrDefault();

49
50     if (newsList == null)//Hervee medee hooson bol {ogogdol
        baihgui baina} gesen utga butsaan
51     {
52         return new { news = "data unavaible" };
53     }
54     else // medee gee butsaah
55         return new { news = newsList };
56 }
57 //
58 public async Task<Object> saveNews(long ID, string Author,
    string Title, string Sub_Title, string NewsData, bool Status
    , string Banner_Img, string TypeList, string Actual, string
    ForeCast, string Previous)
59 {
60     try
61     {
62         using (var scope = serviceScopeFactory.CreateScope())
63         using (var db = scope.ServiceProvider.
            GetRequiredService<Database>())
64         {
65             tbl_News newNews;//tbl_news toroltoi obiekt uusgh
66             newNews.ID = ID;// Medeenii ID
67             newNews.Author = Author;//Medee niitlegch
68             newNews.Title = Title;//Medeenii garchig
69             newNews.Sub_Title = Sub_Title;//Ehnii garchig
70             newNews.Create_Date = DateTime.UtcNow;//Medee

```

```

oruulsan ognoo - yg odoonii hugatsaag avah
71 newNews.NewsData = NewsData; // Medee
72 newNews.Status = Status; ; //Medee niitleh eseh
boolean
73 newNews.Banner_Img = Banner_Img; //Zuragnii URL
74 newNews.TypeList = TypeList; //MEdeenii torol (
calendar, marget..)
75 newNews.Actual = Actual;
76 newNews.ForeCast = ForeCast;
77 newNews.Previous = Previous;
78 db.tbl_News.Add(newNews); //irsen medeeg based nemeh
79 db.SaveChanges();
80
81 return newNews;
82 }
83 }
84 catch (Exception ex)
85 {
86     return new Json("Error", nameof(customErrorMessages.
Exception_General), ex.Message); //Aldaa uush uyd
catch-ruu irne
87 }
88 }
89
90 //Medeeg shinjcleh
91 public async Task<Object> updateNews(long ID, string Author,
string Title, string Sub_Title, string NewsData, bool Status
, string Banner_Img, string TypeList, string Actual, string

```



```

ForeCast, string Previous)
92     {
93         try
94         {
95             using (var scope = serviceScopeFactory.CreateScope())
96             using (var db = scope.ServiceProvider.
                GetRequiredService<Database>())
97             {
98                 tbl_News newNews;//tbl_news toroltoi obiekt uusgh
99                 newNews = db.tbl_News.Where(x => x.ID == ID).
                    FirstOrDefault();//Shinjleh gej bui medeeni ID-r
                        shuuju avah
100                 newNews.Author = Author;//Medee niitlegch
101                 newNews.Title = Title;//Medeenii garchig
102                 newNews.Sub_Title = Sub_Title;//Ehnii garchig
103                 newNews.Create_Date = DateTime.UtcNow;//Medee
                        oruulsan ognoo - yg odoonii hugatsaag avah
104                 newNews.NewsData = NewsData;// Medee
105                 newNews.Status = Status; ;//Medee niitleh eseh
                        boolean
106                 newNews.Banner_Img = Banner_Img; //Zuragnii URL
107                 newNews.TypeList = TypeList;//MEdeenii torol (
                        calendar, marget..)
108                 newNews.Actual = Actual;
109                 newNews.ForeCast = ForeCast;
110                 newNews.Previous = Previous;
111                 db.tbl_News.Update(newNews);//irsen medeeg based
                        shincleh

```

```

112         db.SaveChanges();
113
114         return newNews;
115     }
116 }
117 catch (Exception ex)
118 {
119     return new Json("Error", nameof(customErrorMessages.
        Exception_General), ex.Message); //Aldaa uush uyd
        catch-ruu irne
120 }
121 }
122 }
123 }

```

Код 4.5: **NewsManager.cs** өгөгдлийн санаас авч боловсруулж буцаах хэсгийг энэд агуулна

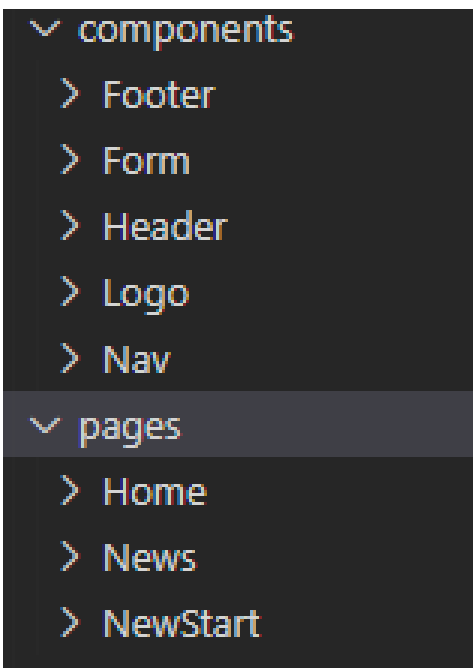
ReactJS frontend хэрэгжүүлэлт

App компонент нь хэрэглэгч нэвтэрсэн бол **NewsStart** компонент render-лэгдэж хэрэглэгчийн эрхүүд нээгдэнэ **localStorage** - хэсэгт хэрэглэгчийн токенийг хадгална ингэснээр хэрэглэгч нэвтэрсэн байна гэж үзнэ. Нэвтрээгүй бол **Form** компонент render-лэгдэнэ.

```

1 function App() {
2     if(!localStorage.getItem("token")) {
3         return (
4             <Form/>
5         );
6     }
7     return <NewsStart/>
8 }

```



Зураг 4.2: ReactJS Нүүрэн талын хөгжүүлэлт

```
9 export default App;
```

Код 4.6: Хэрэглэгч нэвтэрсэн эсхээс хамаарч ажиллах компонент

Form компонент нь хэрэглэгч эхлээд capex.mn сайтад бүртгэлтэй эрхээрээ нэвтэрч нэг удаагийн код авч нэвтэрнэ. **OTPReq** энэ суман функц нь хэрэглэгчийн мэдээллийг шалгаж үнэн зөв байвал нэг удаагийн кодыг утас руу нь илгээнэ ээ. **https://api/AuthenticateUser** энэ api нь мэдээллийг шалгаад үнэн байвал нэг удаагийн кодыг буцааж өгнө. Баталгаажуулах нэг удаагийн дугаар ирэхэд түүнийг хийж **Нэвтрэх** товчийг дарахад **LoginReq** суман функцыг дуудна. Энэ функц нь нэг удаагийн кодыг мөн Тухайн Browser-н DevIceID гээр шүүж хэрэглэгчид **https://token** энэ api-с токен буцаана. Үүнийг localStorage дээ хадгалснаар хэрэглэгч системээс гарсан хэдий ч дахин нэвтрэхэд хэрэглэгчийг шууд таних юм

```
1  
2 const Form = (props) => {  
3   const [email, setEmail] = useState();
```

```

4     const [password, setPassword] = useState();
5     const [otp, setOtp] = useState();
6     const [disable, setDisable] = useState(false);
7
8     const [tempAuthToken, setTempAuthToken] = useState();
9     let OTPReq = async (e) => {
10         e.preventDefault();
11         var data = {
12             email: email,
13             password: password
14         }
15         try {
16             let res = await fetch("https://../api/AuthenticateUser", {
17                 method: 'POST',
18                 headers: {
19                     'Content-Type': 'application/json'
20                 },
21                 body: JSON.stringify(data)
22             });
23             let resJson = await res.json();
24             if (res.status === 200) {
25                 if(resJson.status === "Success"){
26                     setTempAuthToken(resJson.data.tempAuthToken);
27                     toast(
28                         "OTP Submitted",
29                         { type: 'success' }
30                     );
31                     setDisable(true);

```

```

32         }
33         else{
34             toast(
35                 "warning. ",
36                 { type: 'warning' }
37             );
38         }
39     } else {
40         toast(
41             "Error. try again",
42             { type: 'error' }
43         );
44     }
45     } catch (err) {
46         toast(
47             "Error server",
48             { type: 'error' }
49         );
50     }
51 }
52 let LoginReq = async (e) => {
53     e.preventDefault();
54     const data = {
55         grant_type: 'password',
56         username: tempAuthToken,
57         password: otp,
58         dvc_otp: "",
59     };

```

```

60     if (localStorage.getItem('deviceId')) {
61         data.dvc_id = localStorage.getItem('deviceId')
62     }
63     try {
64         let res = await fetch(
65             "https://../token",
66             {
67                 method: 'POST',
68                 headers: { Accept: 'application/json', 'Content-
69                     Type': 'application/x-www-form-urlencoded', },
70                 body: qs.stringify(data)
71             }
72         );
73
74         let OTPRes = await res.json();
75         if (res.status === 200) {
76             localStorage.setItem('token', OTPRes.access_token)
77             toast(
78                 "Success Login",
79                 { type: 'success' }
80             );
81             window.location.reload(true)
82             setOtp();
83             setEmail();
84             setPassword();
85         } else {
86             toast(

```

```

87         { type: 'warning' }
88     );
89 }
90 } catch (err) {
91     toast(
92         "Error try again",
93         { type: 'error' }
94     );
95 }
96 }
97 return(
98     ...
99 )
100 }
101 export default Form;

```

Код 4.7: Хэрэглэгчийн форм бөглөх компонент

Nav компонент нь Лого болон гарах товчтой байна. Хэрэглэгч гарах гэсэн точийг дарах(**onClick=Logout**) д функц дуудагдах бөгөөд уг функц нь хэрэглэгчийн токенийг localStorage хэсгээс цэвэрлэж дахин ачааллуулснаар App.js-н Form компонент дуудагдан

```

1 const Nav = () => {
2     const Logout = () =>{
3         localStorage.setItem("token","");
4         window.location.reload(true)
5     }
6     return(
7         <div className={css.navs}>
8             <Logo />

```

```

9      <div className={css.NavContent}>
10          <button className={css.logout} onClick={Logout}>
11
12          </button>
13      </div>
14  </div>
15  )
16  }
17  export default Nav;

```

Код 4.8: Мэдээний Header хэсэг

Home компонент нь мэдээний жагсаалтууд, Мэдээ засах, нэмэх гэсэн зүйлсээс бүрдэнэ. Программ ачааллах үед `UseEffect` дуудагдаж түүн дотор байсан `getData()` гэсэн функц дуудагдаж ажиллана. Энэ функц нь мэдээний жагсаалтыг `api`-с массив болгож авна. Тэдгээр датаг `map` ашиглаж `render` хийнэ.

```

1  export default function Home() {
2      const [listData, setListData] = useState([]);
3      useEffect(() => {
4          getData();
5      }, []);
6
7      const getData = async () => {
8          const result = await axios.get('https://.../news/getAdminNewsList
9              ');
10         setListData(result.data.data.news);
11     }
12     return (
13         ...

```



```
13     }}
```

Код 4.9: Мэдээний Контент хэсэг

AddNews энэ хуудас нь мэдээ нэмэх үйлдэл хийнэ. Түүнчлэн утгуудаа `useState` ашиглаж утгаа авах бөгөөд `Form` дотроос элементүүдийн `OnChange` үзэгдлийг ашиглаж утгуудыг хувьсагчид хадгална. Бүх талбарыг бөглөж илгээхэд `../news/saveNew` `api` хүлээж аваад өгөгдлийн санд нэмнэ.

```
1 export default function AddNews() {
2   const editorRef = useRef();
3   const [author, setAuthor] = useState();
4   const [title, setTitle] = useState();
5   const [bannerImg, setBannerImg] = useState();
6   const [subTitle, setSubTitle] = useState();
7   const [isLive, setIsLive] = useState(true);
8   const [newsType, setNewsType] = useState(1);
9   const [showCalendar, setShowCalendar] = useState(false);
10  const [actual, setActual] = useState(null);
11  const [foreCast, setForeCast] = useState(null);
12  const [previous, setPrevious] = useState(null);
13
14  const onClickHandler = async () => {
15    const formData = {
16      Author: author,
17      Title: title,
18      Sub_Title: subTitle,
19      NewsData: editorRef.current.getContent(),
20      Banner_Img: bannerImg,
21      Status: isLive,
```

```

22     TypeList: newsType,
23     Actual: actual,
24     ForeCast: foreCast,
25     Previous: previous
26 };
27 var url = "https://.../news/saveNews";
28 const response = await axios.post(url, {
29     formData
30 }, {
31     headers: {
32         'Authorization': `bearer ${localStorage.getItem('token')}`
33     }
34 })
35 if (response.data.status === "Success") {
36     toast("Success", { type: 'success' });
37     setAuthor();
38     setTitle();
39     setSubTitle();
40     editorRef.current = 'A';
41 } else {
42     toast("Error server", { type: 'error' });
43 }
44 }
45
46 const handleChangeAuthor = (event) => {
47     setAuthor(event.target.value);
48 }
49 const handleChangeNewsType = (event) => {

```

```

50     setNewsType(event.target.value);
51     if(event.target.value === "2"){
52         setShowCalendar(true); setActual(0); setForeCast(0); setPrevious
           (0)
53     }else {
54         setShowCalendar(false); setActual(null); setForeCast(null);
           setPrevious(null)
55     }
56 }
57
58 const handleChangeTitle = (event) => { setTitle(event.target.value);
    }
59 const handleChangeBannerImg = (event) => { setBannerImg(event.target.
    value); }
60 const handleChangeSubTitle = (event) => { setSubTitle(event.target.
    value); }
61 const onChangeNews = () => {
62     setIsLive(!isLive);
63 }
64 return (
65     <div>
66         ....
67     </div>
68 )
69 }

```

Код 4.10: Мэдээ нэмэх хэсэг

NewsDetail энэ хуудас нь мэдээ засах үйлдэл хийнэ. Түүнчлэн <https://../news/getNewsList->

ById?id энэ api-г ашиглаж тухайн мэдээний ID-тэй өгөгдүүдийг шшүж авсаны дараа On-Change үзэгдлийг ашиглаж утгуулыг хувьсагчид хадгална. Бүх талбарыг бөглөж илгээхэд **https://.../news/updateNews** api хүлээж аваад өгөгдлийн санд шинэчилний дараа мэдээлэл өөрчлөгдөх юм.

```
1 export default function NewsDetail() {
2
3   const editorRef = useRef();
4   const [author, setAuthor] = useState();
5   const [title, setTitle] = useState();
6   const [bannerImg, setBannerImg] = useState();
7   const [firstTitle, setFirstTitle] = useState();
8   const [isLive, setIsLive] = useState(true);
9
10  useEffect(() => {
11    const location = window.location.href;
12    const ticker = location.substring(location.lastIndexOf('/') + 1);
13    getOfferData(ticker);
14  }, [window.location.href]);
15
16  const getOfferData = async (ticker) => {
17    const result = await axios.get(`https://.../news/getNewsListById?id=${ticker}`);
18    console.log(result.data.data.news.newsData);
19    if (result.data.status === 'Success') {
20      setAuthor(result.data.data.news.author);
21      setTitle(result.data.data.news.title);
22      setBannerImg(result.data.data.news.banner_img);
23      setFirstTitle(result.data.data.news.sub_title);
```

```

24     setIsLive(result.data.data.news.status);
25     editorRef.current = result.data.data.news.newsData;
26 }
27 }
28
29 const onClickHandler = async () => {
30     const formData = {
31         Author: author,
32         Title: title,
33         Sub_Title: firstTitle,
34         NewsData: editorRef.current.getContent(),
35         Banner_Img: bannerImg,
36         Status: isLive
37     };
38     var url = "https://.../news/updateNews";
39     const response = await axios.post(url, {
40         formData
41     }, {
42         headers: {
43             'Authorization': `bearer ${localStorage.getItem('token')}`
44         }
45     })
46     if (response.data.status === "Success") {
47         toast("Success save", { type: 'success' });
48         setAuthor();
49         setTitle();
50         setFirstTitle();
51         editorRef.current = 'A';

```

```

52     } else {
53         toast("Error Server", { type: 'error' });
54     }
55 }
56 const handleChangeAuthor = (event) => { setAuthor(event.target.value)
    ; }
57 const handleChangeTitle = (event) => { setTitle(event.target.value);
    }
58 const handleChangeBannerImg = (event) => { setBannerImg(event.target.
    value); }
59 const handleChangeFirstTitle = (event) => { setFirstTitle(event.
    target.value); }
60 const onChangeNews = () => { setIsLive(!isLive); }
61
62 return (
63     <div>...</div>
64 )
65 }

```

Код 4.11: Мэдээ засах хэсэг

5. ДҮГНЭЛТ

5.1 Үр дүн sarex.mn-ий мэдээний демо төсөл

Хэрэглэгч нэвтрэх үед алдаа гарах

! Алдаа гарлаа

×

Хэрэглэгч нэр эсвэл нууц үгээ буруу оруулах үед

! Та мэдээллээ үнэн зөв оруулна уу

×

Нэг удаагийн илгээх үед

✓ Нэг удаагийн код илгээгдлээ

×

Нэг удаагийн кодыг буруу оруулах үед

! Нэг удаагийн код буруу байна

×

Нэг удаагийн кодыг зөв оруулах үед

✓ Амжилттай нэвтэрлээ

×

Мэдээгээ оруулах үед

✓ Амжилттай хадгаллаа

×

Зураг 5.1: Мэдээний мессэжүүд

The screenshot shows a login form titled "Нэвтрэх" (Login) on a purple background. The form has two input fields: "Имэйл хаягаа оруулна уу" (Enter your email) with the value "purevbaasankhuu@gmail.com" and "Нууц үгээ оруулна уу" (Enter your password) with masked characters "*****". Below the password field is a button "Нэг удаагийн код авах" (Get one-time code). Underneath is the text "Нэг удаагийн код оруулна уу." (Enter one-time code). At the bottom is a button "Нэвтрэх" (Login). A green success message overlay at the top right reads "✓ Нэг удаагийн код илгээгдлээ" (One-time code received) with a close button "x".

Зураг 5.2: Мэдээ рүү хэрэглэгч нэвтрэх



Зураг 5.3: Мэдээний Header хэсэг

Мэдээний төрөл

Зөвхөн танд

Нийтлэгч

Baasankhuu

Гарчиг

Dadlaga

Banner Зураг

https://res.cloudinary.com/daily-now/image/upload/f_auto,q_auto/v1/posts/2a51b32c7b006a0a22dbcacd5101af6e

Эхний гарчиг

Dadlagiin taiian

Мэдээ

File Edit View Insert Format

↶ ↷ Paragraph B I ☰ ☷ ☹ ☹ ☹ ☹ ☹ ☹

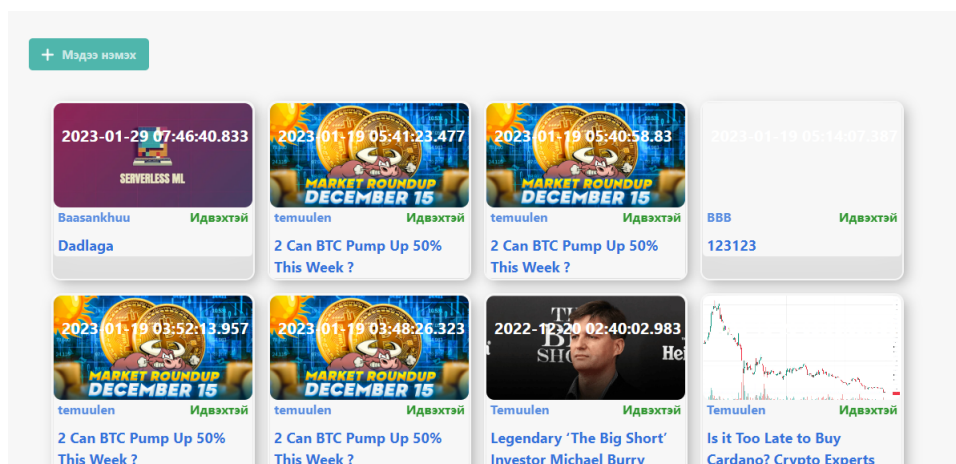
☒ Мэдээг нийтлэх

Мэдээ нэмэх

Зураг 5.5: Мэдээ нэмэх

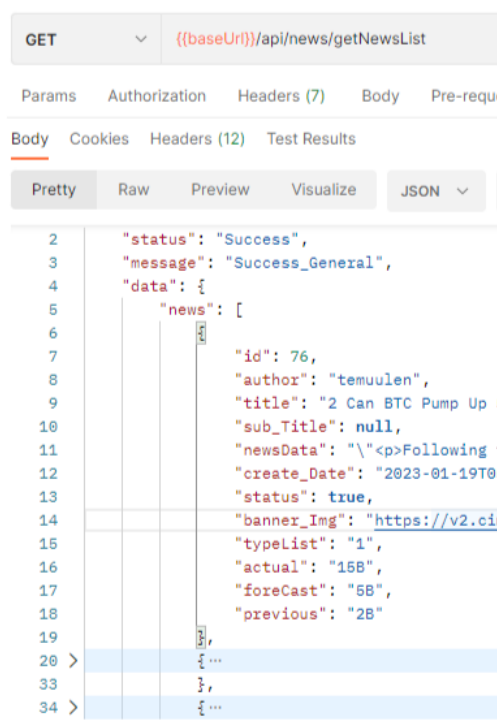
<pre>SELECT typeList, Author, Title,Banner_Img, Sub_Title,Create_Date NewsData FROM tbl_News</pre>						
Results 1 ×						
<pre>SELECT typeList, Author, Title,Banner_Img, Sub_Title,Create_Date NewsData FROM tbl_News</pre>						
	ABC typeList	ABC Author	ABC Title	ABC Banner_Img	ABC Sub_Title	NewsData
1	1	Baasankhuu	Dadlaga		Dadlagiin tailan	2023-01-29 07:46:40.833
2	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	2023-01-29 07:39:44.530
3	1	temuulen	2 Can BTC Pump Up		[NULL]	2023-01-19 05:41:23.477
4	1	temuulen	2 Can BTC Pump Up		[NULL]	2023-01-19 05:40:58.830
5	a	BBB	123123	123asdas.jpg	sub..	2023-01-19 05:14:07.387
6	1	temuulen	2 Can BTC Pump Up		[NULL]	2023-01-19 03:52:13.957
7	1	temuulen	2 Can BTC Pump Up		[NULL]	2023-01-19 03:48:26.323
8	3	Temuulen	Legendary 'The Big Short'		Binance Crypto Exchange	2022-12-20 02:40:02.983
9	1	Temuulen	Is it Too Late to Buy Cardano?		Crypto Experts Give T	2022-12-20 02:38:02.553

Зураг 5.6: Мэдээ хадгалагдсан байдал

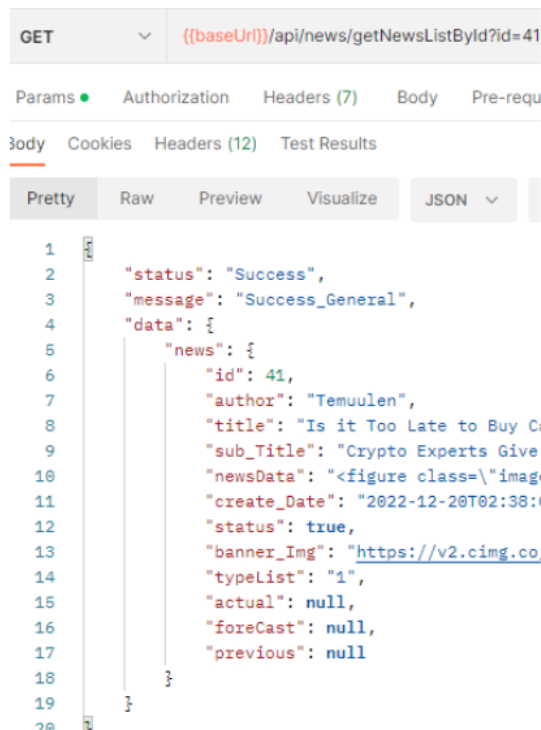


Зураг 5.7: Мэдээнүүд

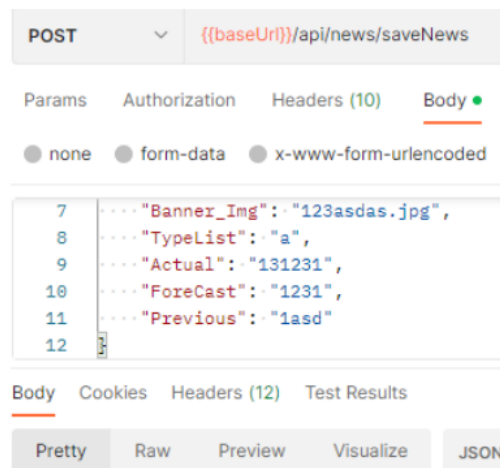
Зураг 5.8: Мэдээний Footer хэсэг



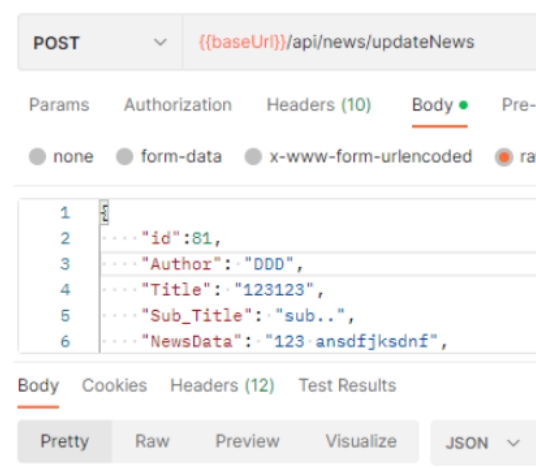
Бүх мэдээг харуулах



ID-р шүүж мэдээг харуулах



Мэдээ нэмэх



Мэдээг шинэчлэх

Зураг 5.9: Postman ашиглаж API-тэй харьцах

6. ҮР ДҮНГИЙН ТАЙЛАН

Миний бие Даталанд Технологй ХХК-д 21 хоногийн хугацаатай мэргэжлийн дадлагыг амжилттай гүйцэтгэж дуусгалаа.Энэ хугацаанд хичээлийн хүрээнд үзсэн онолын ойлголтуудыг практик дээр туршиж, хэрэгжүүлсэн бөгөөд үндсэн судлах технологиудаа судалсны дараа үндсэн хөгжүүлэлт рүү оруулж ажиллуулсан нь надад маш олон зүйл сурахад дэмжлэг болсон. Мөн хөгжүүлэлтэй явцад асуудал тулгарсан асуудал бүрээ шийдэхэд мэдлэгээ бататгах боломж болж байсан. Даталанд Технологи ХХК нь маш найрсаг хамт олонтой ба бизнесийн үйл ажиллагаандаа teams платформ ашиглах, баг хооронд зохицон ажиллах чадвар,хөгжүүлэлтийн шинэ арга барилуудыг амжилттай эзэмшсэн гэж дүгнэж байна. Миний хувьд дадлага хийхээс өмнө postman, git сайн ашиглаж мэддэггүй байсан ба энэ хугацаанд git, postman-тай маш сайн ажиллаж сурсан, Мөн CSharp, reactjs-дээр хөгжүүлэлтээ хийсэн бөгөөд энэ нь миний сонирхлыг маш их татсан учраас цаашдаа веб хөгжүүлэлтээрээ суралцаж мөн ажиллаж явах төлөвлөгөөтэй болсон. Дадлагын явцад кодтойгоо танилцаж таскууд хийх явцдаа жижиг төсөл хэрэгжүүлсэн нь надад уншсан зүйлээ бататгах сайхан боломж олдсон. Дадлагын удирдагчийн хувьд маш нээлттэй,харилцааны ур чадвартай хүн байсан учир харилцахад маш амар асуудал гарвал шууд ханддаг байсан нь их сайхан байсан.Эцэст нь дүгнээд хэлэхэд: Энэхүү дадлагын хугацаанд өмнө нь ашиглаж байгаагүй технологиудтай танилцаж түүн дээрээ хөгжүүлэлт хийж,байгууллагын соёлоос суралцаж чадсан.

Bibliography

- [1] Git, <https://sainaatserenbyamba.medium.com/git-%D0%BD-%D1%82%D0%B0%D0%BB%D0%B0%D0%B0%D1%80-%D0%BE%D0%B9%D0%BB%D0%B3%D0%BE%D0%BB%D1%82%D1%82%D0%BE%D0%B9-%D0%B1%D0%BE%D0%BB%D0%BE%D1%85-%D0%BD%D1%8C-d8871690fe65>
- [2] React Library, <https://mindmajix.com/introduction-to-react-js>
- [3] C Sharp, <https://www.w3schools.in/csharp/history-of-csharp> <http://proprogrammershub.blogspot.com/2016/04/top-10-advantages-of-c.html>

А. УДИРДАГЧИЙН ҮНЭЛГЭЭ

МУИС, ХШУИС-ИЙН МЭДЭЭЛЭЛ, КОМПЬЮТЕРИЙН УХААНЫ
ТЭНХИМ

“ПРОГРАММ ХАНГАМЖ” АНГИЙН
ОЮУТАН БААСАНХҮҮ-ЫН
ДАДЛАГЫН АЖЛЫН УДИРДАГЧИЙН ТОДОРХОЙЛОЛТ

2023 оны 01 сарын 31

“Программ хангамж” ангийн 19B1NUM1461 кодтой оюутан Баасанхүү нь манай байгууллагад 2023 оны 01 сарын 03-ны өдрөөс 01 сарын 27-ны өдөр хүртэл мэргэшүүлэх дадлагыг батлагдсан удирдамж, ажлын төлөвлөгөөний дагуу гүйцэтгэлээ. Дадлагын удирдагч Түвшинтөр-ын удирдамжийн дагуу тухайн оюутны дадлагын ажлыг гүйцэтгэсэн байдал:

1. Системийн ерөнхий бүтэц, ажиллагаатай танилцах - Сайн
2. Орчин тохиргоо хийж, системийг хөгжүүлэлтийн орчинд асааж, холбох - Сайн
3. Backend дээр өгсөн ажлууд - Сайн

Өгсөн ажил хийж байгаа байдал, мэдэхгүй зүйлээ асууж, сурах тал дээр хичээл зүтгэл сайн гаргаж байгаа нь харагдаж байсан. Сурах хүлээн авах тал дээр сайн, түргэн ойлгож сурж байсан, кодчлол чадвар сайн гэж дүгнэж байна.

Үнэлгээний санал:

Дадлагын удирдагч Д.Түвшинтөр / /

Дадлагын удирдагчийн талаарх мэдээлэл:

1. Удирдагчийн овог, нэр, регистрийн дугаар - АЮ94121515
2. Алба хэлтэс, албан тушаал - МТГЗ
3. Утас, н-мэйл хаяг - 99961922
4. Төлбөрийн дансны дугаар банк

Зураг А.1: Удирдагчийн үнэлгээ

В. КОДНЫ ХЭРЭГЖҮҮЛЭЛТ

B.0.1 Form Компонентийн Module css-үүд

```
1 .Container {
2     height: 100%;
3     margin: 0 auto;
4     background: linear-gradient(45deg,#140f28,rgba(77,36,113,.92) 30%,
5         rgba(80,35,119,.92) 31%,rgba(138,16,231,.92) 50%,rgba
6         (79,35,117,.92) 76%,rgba(77,36,113,.92) 77%,#0d122f);
7 }
8 .Form{
9     display: flex;
10    flex-direction: column;
11    position: absolute;
12    top: 50%;
13    left: 50%;
14    transform: translate(-50%, -50%);
15    background: #fff;
16    width: 35%;
17    border-radius: 3%;
18    box-shadow: 0 19px 38px rgba(0, 0, 0, 0.3);
19 }
20 .Title{
21     font-weight: bold;
22     font-size: 32px;
23     text-align: center;
24     margin-top: 5%;
25     margin-bottom: 5%;
26     color: rgba(77,36,113,.92);
27 }
28 .InputName{
29     border: none;
30     outline: none;
31     border-bottom: 1px solid;
32     background: none;
33     margin: 0.9rem 12%;
34     width: 76%;
35     font-size: 1rem;
36 }
37 .InputLbl{
38     margin: 0 12%;
39     font-weight: bolder;
40     opacity: 0.5;
41 }
42 .Btn{
43     text-align: center;
44     margin: 1rem 25%;
```

```

45     padding: 0.5rem;
46     cursor: pointer;
47     border-radius: 0.5rem;
48     border: none;
49     font-size: 1rem;
50     color: rgba(255, 255, 255, 0.8);
51     background: linear-gradient(45deg, #140f28, rgba(77, 36, 113, .92) 30%,
        rgba(80, 35, 119, .92) 31%, rgba(138, 16, 231, .92) 50%, rgba
        (79, 35, 117, .92) 76%, rgba(77, 36, 113, .92) 77%, #0d122f);
52 }
53 .Btn:hover{
54     color: rgba(255, 255, 255, 1);
55     box-shadow: rgba(77, 36, 113, .92) 1px 1px 8px;
56 }
57 button:disabled, input:disabled{
58     cursor: not-allowed;
59     pointer-events: all !important;
60 }
61 @media screen and (min-width: 1778px) {
62     .Form{
63         width: 30%;
64     }
65 }
66 @media screen and (max-width: 768px) {
67     .Form{
68         width: 70%
69     }
70     .InputName{
71         margin: 0.9rem 9%;
72         width: 82%;
73     }
74     .InputLbl{
75         margin: 0 9%;
76     }
77 }

```

B.0.2 Header Компонентийн Module css-үүд

```

1 .NewsTitle{
2     background: linear-gradient(45deg, #140f28, rgba(77, 36, 113, .92) 30%,
        rgba(80, 35, 119, .92) 31%, rgba(138, 16, 231, .92) 50%, rgba
        (79, 35, 117, .92) 76%, rgba(77, 36, 113, .92) 77%, #0d122f);
3 }
4 .Header{
5     display: flex;
6     flex-direction: row;
7     padding: 60px 50px;
8 }
9 .Paragraph{
10     color: white;
11     text-transform: uppercase;

```

```

12     font-size: 47px;
13     line-height: normal;
14     font-weight: bolder;
15 }

```

B.0.3 Нөмө хуудасны Module css-үүд

```

1  .News{
2      background-color: rgb(247, 247, 247);
3      padding-top: 40px;
4      min-height: 65vh;
5  }
6
7  .AddNewsButton{
8      display: flex;
9      flex-direction: row;
10     width: 150px;
11     margin-left: 30px;
12     margin-top: auto;
13     margin-bottom: auto;
14     height: 40px;
15     justify-content: center;
16     align-items: center;
17     color: rgba(255, 255, 255, 0.7);
18     font-weight: bold;
19     font-size: 15px;
20     border-radius: 5px;
21     background-color: rgba(38, 166, 154, 0.8);
22 }
23 .AddNewsButton:hover{
24     color: rgb(255, 255, 255);
25     background-color: rgb(38, 166, 154) ;
26     box-shadow: #26a69a 1px 1px 8px;
27 }
28
29 .addBtn{
30     width: 15px;
31     height: 15px;
32     margin-right: 10px;
33 }
34 .Cards{
35     flex-flow: wrap;
36     display: flex;
37     padding: 30px 50px;
38 }
39 .NewsContainer2, .NewsContainer1{
40     background-color: #F7F7F7;
41 }
42 .NewsContainer1{
43     width: 250px;
44     padding: 1px;

```



```

45     margin: 10px 10px;
46     border-radius: 10px;
47     box-shadow: inset 0 -3em 3em rgba(0, 0, 0, 0.1), 0 0 0 2px #F7F7F7,
48         0.3em 0.3em 1em rgba(0, 0, 0, 0.3);
49 }
50 .NewsContainer2{
51     width: 100%;
52     border-radius: 5px;
53     display: flex;
54     flex-direction: column;
55     justify-content: center;
56     cursor: pointer;
57 }
58 .HomeListBanner{
59     width: 100%;
60     height: 130px;
61     background-size: cover;
62     background-position: center center;
63     display: flex;
64     border-radius: 10px;
65 }
66 .NewsTimeText{
67     color: #FFF;
68     font-weight: bold;
69     font-size: 20px;
70     margin-top: 25px;
71     margin-left: 10px;
72 }
73 .StatusUser{
74     display: flex;
75     flex-direction: row;
76     width: 100%;
77 }
78
79 .UserAuth{
80     width: 50%;
81     text-align: start;
82     margin-left: 5px;
83     font-weight: bold;
84     opacity: 0.8;
85 }
86 .Status{
87     width: 50%;
88     text-align: end;
89     margin-right: 5px;
90     font-weight: bold;
91     opacity: 0.8;
92 }
93 .TitleParagraph{
94     margin-top: 10px;
95     margin-left: 5px;
96     margin-right: 5px;

```

```

97     font-size: 18px;
98     font-weight: bold;
99 }

```

B.0.4 AddNews болон NewsDetail хуудасны Module css-үүд

```

1  .AddNewsContainer{
2      display: flex;
3      flex-direction: column;
4      margin-bottom: 30px;
5      justify-content: flex-start;
6      align-items: flex-start;
7      width: 100%;
8  }
9  .newsBody{
10     margin: 50px;
11 }
12 .TitleTextStyle{
13     margin-left: 50px;
14     margin-top: 50px;
15     font-size: 30px;
16     cursor: pointer;
17 }
18
19 .TitleTextStyle:hover{
20     opacity: 0.8;
21 }
22 .backIcon{
23     width: auto;
24     height: auto;
25     margin-top: 30px;
26 }
27 .AddNewsLabel{
28     margin-bottom: 10px;
29     font-size: 20px;
30     font-weight: bold;
31 }
32 .InputStyle{
33     border-radius: 5px;
34     border-width: thin;
35     height: 40px;
36     width: 100%;
37 }
38 .News{
39     width: 100%;
40     height: 400px;
41 }
42 .NewsCommitCont{
43     display: flex;
44     flex-direction: row;
45     margin-bottom: 30px;

```

```

46 }
47 .NewsCommitted{
48     margin-top: 8px;
49     font-weight: bold;
50     font-size: 18px;
51 }
52 .ButtonStyle{
53     cursor: pointer;
54     padding: 0.5rem 1rem;
55     font-size: 1.25rem;
56     line-height: 1.5;
57     border-radius: 0.3rem;
58     color: #fff;
59     background-color: #007bff;
60     border-color: #007bff;
61     display: inline-block;
62     font-weight: 400;
63     text-align: center;
64     white-space: nowrap;
65     vertical-align: middle;
66     transition: color .15s ease-in-out,background-color .15s ease-in-out,
67               border-color .15s ease-in-out,box-shadow .15s ease-in-out;
68     border: 1px solid transparent;
69 }
70 .ButtonStyle:hover{
71     color: #fff;
72     background-color: #0069d9;
73     border-color: #0062cc;
74 }

```