

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

Анужингийн Сайнзолбоо

**АЖИЛ ОЛГОГЧДЫН ӨГӨГДЛИЙН АНАЛИЗ  
СИСТЕМ ДЭЭР СУУРИЛСАН ЧАТ БОТ  
(Chat bot based on sytem analysis of employers' data)**

Мэдээллийн технологи (D061303)  
Бакалаврын судалгааны ажил

Улаанбаатар

2022 оны 03 сар

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

**АЖИЛ ОЛГОГЧДЫН ӨГӨГДЛИЙН АНАЛИЗ СИСТЕМ**  
**ДЭЭР СУУРИЛСАН ЧАТ БОТ**

**(Chat bot based on sytem analysis of employers' data)**

Мэдээллийн технологи (D061303)  
Бакалаврын судалгааны ажил

Удирдагч: \_\_\_\_\_ Др. Ч.Алтангэрэл

Хамтран удирдагч: \_\_\_\_\_ Др. Г.Амарсанаа

Гүйцэтгэсэн: \_\_\_\_\_ А.Сайнзолбоо ('18B1NUM1762')

Улаанбаатар

2022 оны 03 сар

# Зохиогчийн баталгаа

Миний бие Анужингийн Сайнзолбоо ”АЖИЛ ОЛГОГЧДЫН ӨГӨГДЛИЙН АНАЛИЗ СИСТЕМ ДЭЭР СУУРИЛСАН ЧАТ БОТ” 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.

## ГАРЧИГ

УДИРТГАЛ .....	1
БҮЛГҮҮД .....	2
1. СЭДВИЙН ТАНИЛЦУУЛГА .....	2
1.1 Оршил .....	2
1.2 Зорилго .....	2
1.3 Зорилт .....	2
1.4 Алсын хараа .....	3
2. СИСТЕМИЙН СУДАЛГАА .....	4
2.1 Системийн судалгаа .....	4
2.2 Ижил төстэй системүүд .....	4
2.3 Технологийн судалгаа .....	5
3. СИСТЕМИЙН ШИНЖИЛГЭЭ .....	6
3.1 Бизнесийн үйл ажиллагааны шинжилгээ .....	6
3.2 Хэрэглэгч .....	6
3.3 Функционал шаардлага .....	6
3.4 Функционал бус шаардлага .....	6
3.5 Use case диаграм .....	6
4. СИСТЕМИЙН ЗОХИОМЖ .....	7
4.1 Өгөгдлийн сангийн диаграм .....	7
4.2 Өгөгдлийн элемент .....	7
5. ХЭРЭГЖҮҮЛЭЛТ, ҮР ДҮН .....	8
5.1 Өгөгдлийн сангийн диаграм .....	8
5.2 Өгөгдлийн элемент .....	8

НОМ ЗҮЙ .....	9
ХАВСРАЛТ .....	10
.1    Үечилсэн төлөвлөгөө .....	10
А. КОДЫН ХЭРЭГЖҮҮЛЭЛТ .....	11
А.1  Өгөгдөл цугуулалт .....	11

## ЗУРГИЙН ЖАГСААЛТ

1	Бакалаврын судалгааны ажлын үечилсэн төлөвлөгөө .....	10
---	---	----

## ХҮСНЭГТИЙН ЖАГСААЛТ

# Кодын жагсаалт

A.1	xd . . . . .	11
-----	--------------	----



## **УДИРТГАЛ**

# 1. СЭДВИЙН ТАНИЛЦУУЛГА

## 1.1 Оршил

Энэхүү бакалаврын судалгааны ажлын хүрээнд "Ажил олгогчдын өгөгдлийн анализ систем дээр суурилсан чатбот" сэдвийн дагуу ажлын байрны мэдээллээр хангах Чатбот системийг хөгжүүлнэ. Ажлын байрны мэдээллийг Data Scraping аргын тусламжтайгаар, системд шаардлагатай мэдээллийг өгөгдлийн сангийн хэлбэрт оруулан бүтэцтэйгээр нэгтгэн авах бөгөөд үүнээс ажил горьлогчдын дунд байдаг түгээмэл асуултуудын хариултыг өгнө. Мөн энэ системд машин сургалтын арга болох Language Understanding-ийг ашиглан хэрэглэгчийн асуултыг таамаглаж оновчтой хариулт өгөх боломжийг олгох юм.

## 1.2 Зорилго

Ажлын горьлогчдын хэрэгцээт асуултад хариулж, ажлын байрны хүртээмжийг нэмэгдүүлэхэд энэхүү системийн гол зорилго оршино.

## 1.3 Зорилт

Дээрх зорилгод хүрэхийн тулд дараах зорилтуудыг тавьсан. Үүнд:

- Ашиглагдах технологиудыг сонгох, судлах
- Ижил төстэй системийн судалгаа хийх
- Системийн шинжилгээ хийх
- Системийг зохиомжлох
- Системийг хөгжүүлэх, сайжруулалт хийх

## **1.4 Алсын хараа**

Ажлын байрны дэлгэрэнгүй мэдээллийг цуглуулснаар цаашид тэдгээрт шинжилгээ хийж эрэлттэй ажлын байр, өндөр цалинтай ажлын байр гэх зэрэг мэдээллүүдийг систем хэрэглэгчдэд хүргэх боломжтой юм.

## 2. СИСТЕМИЙН СУДАЛГАА

Технологийн хөгжил болон Ковид-19 цар тахлаас үүдэн дэлхийн хэмжээнд онлайн худалдааны сайтуудын борлуулалт, худалдан авагч болон онлайн дэлгүүрүүдийн тоо өндөр өсөлттэй байна. Үүнийгээ дагаад маш олон тооны интернетээр бараа бүтээгдэхүүн борлуулдаг (e-commerce) төрлийн сайтууд үүссэн ба эдгээрээс манай бүтээгдэхүүнтэй ижил төстэй 2 системийг авч танилцууллаа.

### 2.1 Системийн судалгаа

Ашигласан технологийн хувьд яг ижил ба үүнд React, GraphQL, Django технологиудыг ашиглан хөгжүүлжээ. Мөн Merchant вэбийнхээ интерфэйс дээр Material-ui хэрэглэсэн байна.

Хэрэглэгч буюу дэлгүүрийн админ өөрийн хүссэн үедээ шинээр дэлгүүр нээх боломжтой. Ингэснээр зөвхөн танай дэлгүүрт зориулагдсан e-commerce вэб, дэлгүүрээ удирдах мерчант вэбийг бэлдэн гаргаж өгөх болно. Манай Oneline төсөл дэлгүүр хэсгээ зөвхөн нэг гар утасны апп дээр шийдсэн бол уг систем вэб болон PWA ашиглан дэлгүүр бүрт шинэ бүтээгдэхүүн өгдөг байдлаараа ялгаатай байна.

### 2.2 Ижил төстэй системүүд

Ашигласан технологийн хувьд Saleor болон манай Oneline-тай мөн ижил. Энэ системийн онцлог нь зөвхөн хүнсний ногоо хэрэглэгчдэд борлуулдаг бөгөөд өөр дээрээ хүргэлтийн үйлчилгээтэй. Ингэснээр өөрийн дэлгүүрийг нээсэн хүн заавал ямар нэг барилга дотор бодит дэлгүүр барьж, түрээс, тог цахилгаан гэх мэт урсгал зардлын мөнгө төлөх шаардлагагүй болж өндөр үнийн дүнг хэмнэж чаддаг. Тийм учир зарагдаж буй хүнсний ногоо зах зээлийн үнээс хангалттай доогуур, эрүүл, аюулгүй байж чаддаг байна. Та вэб болон гар утасны аппыг нь зэрэг ашиглах боломжтой

## **2.3 Технологийн судалгаа**

## **3. СИСТЕМИЙН ШИНЖИЛГЭЭ**

**3.1 Бизнесийн үйл ажиллагааны шинжилгээ**

**3.2 Хэрэглэгч**

**3.3 Функционал шаардлага**

**3.4 Функционал бус шаардлага**

**3.5 Use case диаграм**

## 4. СИСТЕМИЙН ЗОХИОМЖ

Технологийн хөгжил болон Ковид-19 цар тахлаас үүдэн дэлхийн хэмжээнд онлайн худалдааны сайтуудын борлуулалт, худалдан авагч болон онлайн дэлгүүрүүдийн тоо өндөр өсөлттэй байна. Үүнийгээ дагаад маш олон тооны интернетээр бараа бүтээгдэхүүн борлуулдаг (e-commerce) төрлийн сайтууд үүссэн ба эдгээрээс манай бүтээгдэхүүнтэй ижил төстэй 2 системийг авч танилцууллаа.

### 4.1 Өгөгдлийн сангийн диаграм

Ашигласан технологийн хувьд яг ижил ба үүнд React, GraphQL, Django технологиудыг ашиглан хөгжүүлжээ. Мөн Merchant вэбийнхээ интерфэйс дээр Material-ui хэрэглэсэн байна.

Хэрэглэгч буюу дэлгүүрийн админ өөрийн хүссэн үедээ шинээр дэлгүүр нээх боломжтой. Ингэснээр зөвхөн танай дэлгүүрт зориулагдсан e-commerce вэб, дэлгүүрээ удирдах мерчант вэбийг бэлдэн гаргаж өгөх болно. Манай Oneline төсөл дэлгүүр хэсгээ зөвхөн нэг гар утасны апп дээр шийдсэн бол уг систем вэб болон PWA ашиглан дэлгүүр бүрт шинэ бүтээгдэхүүн өгдөг байдлаараа ялгаатай байна.

### 4.2 Өгөгдлийн элемент

Ашигласан технологийн хувьд Saleor болон манай Oneline-тай мөн ижил. Энэ системийн онцлог нь зөвхөн хүнсний ногоо хэрэглэгчдэд борлуулдаг бөгөөд өөр дээрээ хүргэлтийн үйлчилгээтэй. Ингэснээр өөрийн дэлгүүрийг нээсэн хүн заавал ямар нэг барилга дотор бодит дэлгүүр барьж, түрээс, тог цахилгаан гэх мэт урсгал зардлын мөнгө төлөх шаардлагагүй болж өндөр үнийн дүнг хэмнэж чаддаг. Тийм учир зарагдаж буй хүнсний ногоо зах зээлийн үнээс хангалттай доогуур, эрүүл, аюулгүй байж чаддаг байна. Та вэб болон гар утасны аппыг нь зэрэг ашиглах боломжтой

## 5. ХЭРЭГЖҮҮЛЭЛТ, ҮР ДҮН

Технологийн хөгжил болон Ковид-19 цар тахлаас үүдэн дэлхийн хэмжээнд онлайн худалдааны сайтуудын борлуулалт, худалдан авагч болон онлайн дэлгүүрүүдийн тоо өндөр өсөлттэй байна. Үүнийгээ дагаад маш олон тооны интернетээр бараа бүтээгдэхүүн борлуулдаг (e-commerce) төрлийн сайтууд үүссэн ба эдгээрээс манай бүтээгдэхүүнтэй ижил төстэй 2 системийг авч танилцууллаа.

### 5.1 Өгөгдлийн сангийн диаграм

Ашигласан технологийн хувьд яг ижил ба үүнд React, GraphQL, Django технологиудыг ашиглан хөгжүүлжээ. Мөн Merchant вэбийнхээ интерфэйс дээр Material-ui хэрэглэсэн байна.

Хэрэглэгч буюу дэлгүүрийн админ өөрийн хүссэн үедээ шинээр дэлгүүр нээх боломжтой. Ингэснээр зөвхөн танай дэлгүүрт зориулагдсан e-commerce вэб, дэлгүүрээ удирдах мерчант вэбийг бэлдэн гаргаж өгөх болно. Манай Oneline төсөл дэлгүүр хэсгээ зөвхөн нэг гар утасны апп дээр шийдсэн бол уг систем вэб болон PWA ашиглан дэлгүүр бүрт шинэ бүтээгдэхүүн өгдөг байдлаараа ялгаатай байна.

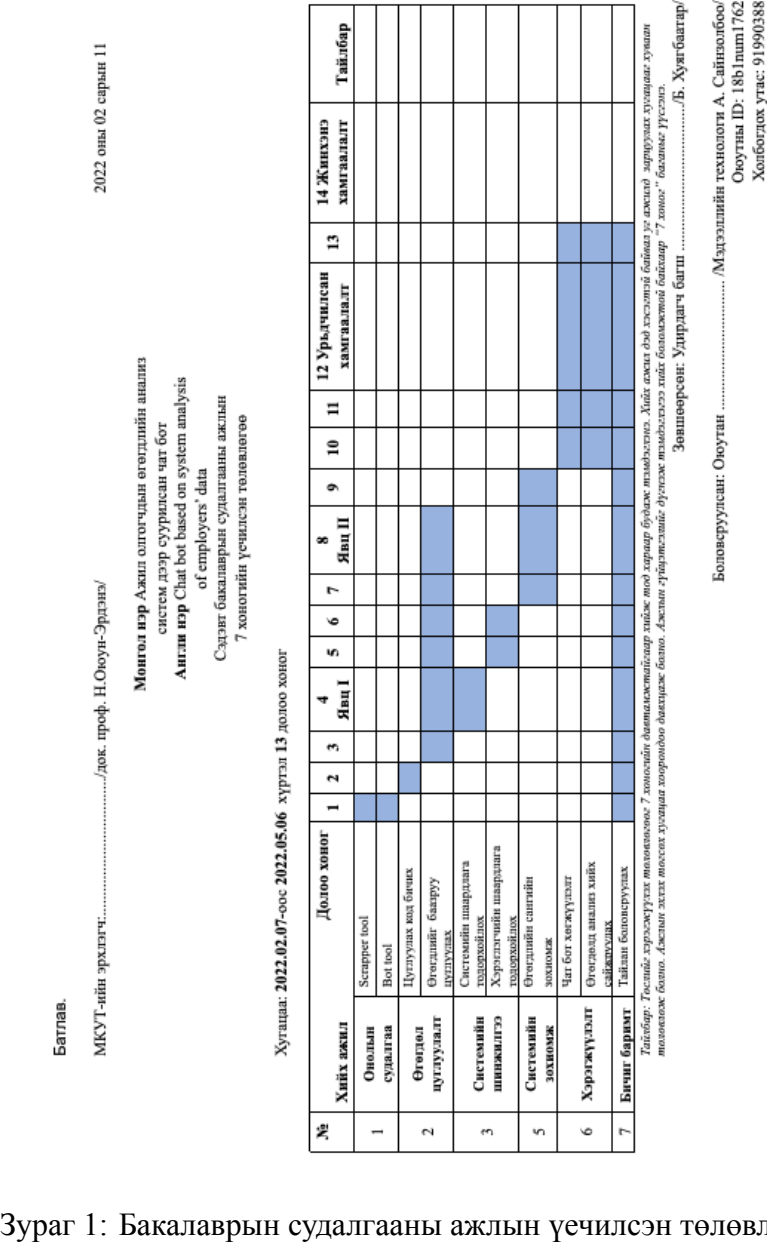
### 5.2 Өгөгдлийн элемент

Ашигласан технологийн хувьд Saleor болон манай Oneline-тай мөн ижил. Энэ системийн онцлог нь зөвхөн хүнсний ногоо хэрэглэгчдэд борлуулдаг бөгөөд өөр дээрээ хүргэлтийн үйлчилгээтэй. Ингэснээр өөрийн дэлгүүрийг нээсэн хүн заавал ямар нэг барилга дотор бодит дэлгүүр барьж, түрээс, тог цахилгаан гэх мэт урсгал зардлын мөнгө төлөх шаардлагагүй болж өндөр үнийн дүнг хэмнэж чаддаг. Тийм учир зарагдаж буй хүнсний ногоо зах зээлийн үнээс хангалттай доогуур, эрүүл, аюулгүй байж чаддаг байна. Та вэб болон гар утасны аппыг нь зэрэг ашиглах боломжтой



# **Bibliography**

.1 Үечилсэн төлөвлөгөө



Зураг 1: Бакалаврын судалгааны ажлын үечилсэн төлөвлөгөө

# A. КОДЫН ХЭРЭГЖҮҮЛЭЛТ

## A.1 Өгөгдөл цуглуулалт

Beautiful Soup ашиглан үндсэн өгөгдлийг цуглуулах програм нь дараах бүтэцтэй байна.

- Өгөгдөл цуглуулах үндсэн код (dataScrapping.py)
- 

### A.1.1 *Beautiful Soup ашиглан үндсэн өгөгдлийг цуглуулах эх код*

```
1 from datetime import date
2 import time
3 from assets.classTypes import Category
4 from assets.scrape import UseBeautifulSoup as useScrape
5 from assets.adScrape import advertisementScrape as useAdScrape
6 from assets.pagination import createLinkList as createLinkList
7
8 start_time = time.time()
9 initialUrl = 'https://www.zangia.mn/'
10 today = str(date.today())
11 # all categories set
12 categorySet = set()
13 # all advertisement's link set
14 adUrlDict = {}
15 # all ads object set
16 adsSet = set()
17
18 # scrape initial links
19 soup = useScrape(initialUrl)
20 navigatorList = soup.find_all('div', class_='filter')
21 for navigator in navigatorList:
22     if navigator.find('h3').text.strip() != '    ,    ':
23         continue
24     # ALL CATEGORY LINKS
25     categoryList = navigator.find_all('div')
26
27 for categoryItem in categoryList:
28     categories = categoryItem.find('a')
29     url = initialUrl + categories['href']
30     tempCategory = Category(url, categories.text, '')
31     print('CATEGORY LINK SCRAPED! ', url)
32     soup = useScrape(url)
33     subCategory = soup.find('div', class_='pros')
34     # ALL SUBCATEGORY LINKS
35     subCategoryList = subCategory.find_all('a')
36     for subCategoryItem in subCategoryList:
37         subCategoryUrl = initialUrl + subCategoryItem['href']
38         tempSubCategory = Category(
```

```

39         subCategoryId, subCategoryId.text, tempCategory.name)
40         categorySet.add(tempSubCategory)
41
42     for categoryItem in categorySet:
43         if categoryItem.parentId == '':
44             continue
45         soup = useScrape(categoryItem.url)
46         hasPagination = soup.find('div', class_='page-link')
47         pagesUrl = []
48         if hasPagination != None:
49             pagesUrl = createLinkList(hasPagination, categoryItem.url)
50         else:
51             pagesUrl.append(categoryItem.url)
52         for pageUrl in pagesUrl:
53             soup = useScrape(pageUrl)
54             ads = soup.find_all('div', class_='ad')
55             # CREATE UNIQUE AD DICTIONARY
56             for ad in ads:
57                 adUrl = initialUrl+ad.find('a', class_=None)['href']
58                 adUrlDict[adUrl] = categoryItem
59     print(pagesUrl)
60     pagesUrl.clear()
61
62     file = open(today+'adScrape.csv', 'w', encoding='utf-8')
63     file.write('Parent Category Name' + '\t' +
64               'Category Name' + '\t' +
65               'Link' + '\t' +
66               'Employee Company' + '\t' +
67               'Title' + '\t' +
68               'Roles' + '\t' +
69               'Requirements' + '\t' +
70               'Additional Info' + '\t' +
71               'City/Province' + '\t' +
72               'District' + '\t' +
73               'Level' + '\t' +
74               'Type' + '\t' +
75               'Min Salary' + '\t' +
76               'Max Salary' + '\t' +
77               'Is Dealable' + '\t' +
78               'Address' + '\t' +
79               'Phone' + '\t' +
80               'Fax' + '\t' +
81               'Ad Added Date' + '\n')
82     print(adUrlDict)
83     for adUrl in adUrlDict:
84         print(adUrl)
85         try:
86             tempAdItem = useAdScrape(adUrl)
87             tempAdItem.setCategory(adUrlDict[adUrl])
88             file.write(
89                 tempAdItem.category.parentId+'\t' +
90                 tempAdItem.category.name+'\t' +

```

```
91         tempAdItem.url+'\t' +
92         tempAdItem.company+'\t' +
93         tempAdItem.title+'\t' +
94         tempAdItem.roles+'\t' +
95         tempAdItem.requirements+'\t' +
96         tempAdItem.additionalInfo+'\t' +
97         tempAdItem.city+'\t' +
98         tempAdItem.district+'\t' +
99         tempAdItem.level+'\t' +
100        tempAdItem.type+'\t' +
101        tempAdItem.minSalary+'\t' +
102        tempAdItem.maxSalary+'\t' +
103        tempAdItem.isDealable+'\t' +
104        tempAdItem.address+'\t' +
105        tempAdItem.phoneNumber+'\t' +
106        tempAdItem.fax+'\t' +
107        tempAdItem.adAddedDate+'\n')
108    del tempAdItem
109 except:
110     print('Ad writing error')
111 file.close()
112 print("--- %s seconds ---" % (time.time() - start_time))
```

Код A.1: xd