

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Сайлаубек Алтай

«Несие алушының төлем қабілеттілігін бағалаушы жүйені жобалау»

ДИПЛОМДЫҚ ЖҰМЫС

5B070300 – «Ақпараттық жүйелер» мамандығы

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ

КАӨЖС кафедрасы меңгерушісі
техн. ғыл. канд, ассистент-
профессор

_____ Н.А. Сейлова
“_____” _____ 20__ ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: «Несие алушының төлем қабілеттілігін бағалаушы жүйені жобалау»

5B070300 – «Ақпараттық жүйелер» мамандығы

Орындаған

Сайлаубек Алтай

Пікір беруші

Лектор

_____ Бауыржан М.Б.
«_____» мамыр 2020 ж.

Ғылыми жетекші

Лектор

_____ Бауыржан М.Б.
«_____» мамыр 2020 ж.

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

5B070300 – «Ақпараттық жүйелер» мамандығы

БЕКІТЕМІН

КАӨЖС кафедрасы меңгерушісі
техн. ғыл. канд, ассистент-
профессор

_____ Н.А. Сейлова
“_____” _____ 20__ ж.

**Дипломдық жұмыс орындауға
ТАПСЫРМА**

Білім алушы Сайлаубек Алтай

Тақырыбы: Несие алушының төлем қабілеттілігін бағалаушы жүйені жобалау

Университет Ректорының «27» қаңтар 2020 жылғы №762 -б бұйрығымен
бекітілген

Аяқталған жұмысты тапсыру мерзімі «27» мамыр 2020 ж.

Дипломдық жұмыстың бастапқы берілістері: Қазіргі күнгі несие бюроларының
жұмысын талдау және теориялық түрде скорингтік жүйелерді талдау

Дипломдық жұмыста қарастырылатын мәселелер тізімі

а) қазіргі банктердің клиенттер туралы жинайтын ақпараттарын талдау

ә) скорингтік жүйелерді талдау

б) төлем қабілеттілігін анықтайтын параметрлерді анықтау

Сызба материалдарының _____ 17 _____ слайдта көрсетілген

Ұсынылатын негізгі әдебиет _____ 20 _____ атаудан тұрады

Дипломдық жұмысты дайындау
КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекші мен кеңесшілерге көрсету мерзімдері	Ескерту
Қазіргі банктердің клиенттер туралы жинайтын ақпараттарын талдау	13.01.2020-10.02.2020	
Скорингтік жүйелерді талдау	11.02.2020-02.03.2020	
Төлем қабілеттілігін анықтайтын параметрлерді анықтау	03.03.2020-27.04.2020	

норма бақылаушының аяқталған жұмысқа (жобаға) қойған
қолтаңбалары

Бөлімдер атауы	Кеңесшілер, аты, әкесінің аты, тегі (ғылыми дәрежесі, атағы)	Қол қойылған күні	Қолы
Бағдарламалық қамтама бақылаушы	Бауыржан М.Б. лектор		
Норма бақылау	А.Н. Дуйсенбаева, техн.ғыл.магистрі, лектор		

Ғылыми жетекші _____ Бауыржан М.Б.

Тапсырманы орындауға алған білім алушы _____ Сайлаубек А.

Күні _____ «27» қаңтар 2020 ж.

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
СӨТБАЕВ университеті

Ғылыми жетекшінің сын пікірі

Дипломдық жобаға

Сайлаубек Алтайға

5B070300 – Ақпараттық жүйелер

Тақырыбы: Несие алушының төлем қабілеттілігін бағалаушы жүйені жобалау

Дипломдық жұмыс түсіндірме хат кіріспеден, қорытындыдан, пайдаланылған әдебиеттер тізімінен тұрады.

Бұл дипломдық жұмыс несие алушының төлем қабілеттілігін бағалаушы жүйе құруға арналған. Несие алушылардың несиелік қабілеттілігін талдау және несие беру туралы шешім қабылдау несие схемасының кезеңдерінің ең маңыздысы болып табылады.

Дипломдық жұмыста өтінім қабылдаушы веб-парақшаны құру үшін Python – web программалау тілі және мәліметтер қорын құру сонымен қатар несие қабілеттілігін бағалаушы процедура құру үшін Django қолданылады.

Сайлаубек Алтай дипломдық жұмыс орындау барысында өз беттілік көрсетіп, жаңа бағдарламалық камтамалармен танысып және әдебиеттермен жұмыс істеген. Дипломдық жұмыс “Ақпараттық жүйелер” мамандығының бітіру жұмыстарына қойылатын талаптарға сәйкес.

Қорытындылай келе, Сайлаубек Алтайдың дипломдық жобасын қорғауға жіберуге болады.

Ғылыми жетекші:

Техника ғылымдарының магистрі, тьютор

_____ М. Б. Бауыржан

«__» _____ 2020 ж

Протокол анализа Отчета подобия

заведующего кафедрой / начальника структурного подразделения

Заведующий кафедрой / начальник структурного подразделения заявляет, что ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Сайлаубек Алтай

Название: Несие алушының төлем қабілеттілігін бағалаушы жүйені жобалау

Координатор: Мадина Бауыржан

Коэффициент подобия 1:8,9

Коэффициент подобия 2:1,7

Замена букв:0

Интервалы:0

Микропробелы:0

Белые знаки:0

После анализа отчета подобия заведующий кафедрой / начальник структурного подразделения констатирует следующее:

- ☐ обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, работа признается самостоятельной и допускается к защите;
- ☐ обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- ☐ обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, работа не допускается к защите.

Обоснование:

.....

.....

.....

.....

.....

.....

.....

.....
Дата

.....
*Подпись заведующего кафедрой /
начальника структурного подразделения*

Окончательное решение в отношении допуска к защите, включая обоснование:

.....

.....

.....

.....

.....

.....

.....

.....
Дата

.....
*Подпись заведующего кафедрой /
начальника структурного подразделения*

Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Сайлаубек Алтай

Название: Несие алушының төлем қабілеттілігін бағалаушы жүйені жобалау

Координатор: Мадина Бауыржан

Коэффициент подобия 1:8,9

Коэффициент подобия 2:1,7

Замена букв:0

Интервалы:0

Микропробелы:0

Белые знаки: 0

После анализа Отчета подобия констатирую следующее:

☐ обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;

☐ обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;

☐ обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование:

.....
.....

.....
Дата

.....
Подпись Научного руководителя

ДАТА ОТЧЕТА: 2020-05-25 10:23:10

НАЗВАНИЕ:

Несие алушының төлем қабілеттілігін бағалаушы жүйе құру

АВТОР:

Сайлаубек Алтай

НАУЧНЫЙ РУКОВОДИТЕЛЬ:

Мадина Бауыржан

ПОДРАЗДЕЛЕНИЕ:

ИКиИТ

ДАТА ЗАГРУЗКИ ДОКУМЕНТА:

2020-05-25 10:21:55

ЧИСЛО ПРОВЕРОК ДОКУМЕНТА:

1

ПРОПУЩЕННЫЕ ВЕБ-СТРАНИЦЫ:

Уровень заимствований

Обратите внимание! Высокие значения коэффициентов не означают плагиат. Отчет должен быть проанализирован экспертом.



Предупреждение и сигналы тревоги

В этом разделе вы найдете информацию, касающуюся манипуляций в тексте, с целью изменить результаты проверки. Для того, кто оценивает работу на бумажном носителе или в электронном формате, манипуляции могут быть невидимы (может быть также целенаправленное вписывание ошибок). Следует оценить, являются ли изменения преднамеренными или нет.

Замена букв	0	показать в тексте
Использование символов из другого алфавита - может указывать на способ обойти систему, поэтому следует установить их использование.		
Интервалы	0	показать в тексте
Количество увеличенного расстояния между буквами (просим определить является ли расстояние имитацией пробела, так как исходно слова могут быть написаны слитно).		
Микропробелы	0	показать в тексте
Количество пробелов с нулевым размером - необходимо проверить влияют ли они на неправильное разделение слов в тексте.		
Белые знаки	0	показать в тексте
Количество символов, выделенных белым цветом, пожалуйста, проверьте не используются ли белые символы вместо пробела, соединяя слова (в отчете подобиия система изменяет автоматически цвет букв в черный, чтобы их сделать видимыми).		

Заимствования по списку источников

Просмотрите список и проанализируйте, в особенности, те фрагменты, которые превышают КП №2 (выделенные жирным шрифтом). Используйте ссылку «Обозначить фрагмент» и посмотрите, являются ли выделенные фрагменты повторяющимися короткими фразами, разбросанными в документе (совпадающие сходства), многочисленными короткими фразами расположенные рядом друг с другом (парафразирование) или обширными фрагментами без указания источника ("криптоцитаты").

10 самых длинных фраз (3,77 %)

Десять самых длинных фрагментов найденных во всех доступных ресурсах.

ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ И АДРЕС ИСТОЧНИКА URL (НАЗВАНИЕ БАЗЫ)	АВТОР	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ	
1	https://stud.kz/referat/show/6527		47	0,71 %
2	https://stud.kz/referat/show/6527		41	0,62 %
3	http://ansya.ru/health/lekciya-45-saat-sj-45-saat-obsj-45-saat-barli-saat-sani--135-s/pg-7.html		25	0,38 %
4	https://studopedia.org/11-7721.html		22	0,33 %
5	http://adilet.zan.kz/kaz/docs/Z1300000094		22	0,33 %
6	https://zharar.kz/id/1905		21	0,32 %
7	https://studopedia.org/11-7721.html		20	0,30 %
8	http://kzdocs.docdat.com/docs/index-53385.html?page=2		20	0,30 %

9	Документ из базы НЭУ 51d11a14-211c-4606-8c64-071fc0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	17	0,26 %
10	https://stud.kz/referat/show/6527		16	0,24 %

из базы данных RefBooks (0,00 %)

Все фрагменты найдены в базе данных RefBooks, которая содержит более 3 миллионов текстов от редакторов и авторов.

ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	АВТОР	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (КОЛИЧЕСТВО ФРАГМЕНТОВ)
---------------------	----------	-------	---

ЗАИМСТВОВАНИЙ НЕ НАЙДЕНО

из домашней базы данных (0,00 %)

Все фрагменты найдены в базе данных вашего университета.

ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	АВТОР	ДАТА ИНДЕКСАЦИИ	ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
---------------------	----------	-------	--------------------	---------------------------------

ЗАИМСТВОВАНИЙ НЕ НАЙДЕНО

из программы обмена базами данных (1,17 %)

Все фрагменты найдены в базе данных других университетов.

ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ НАЗВАНИЕ БАЗЫ ДАННЫХ	АВТОР	ДАТА ИНДЕКСАЦИИ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (КОЛИЧЕСТВО ФРАГМЕНТОВ)
1	Документ из базы НЭУ 51d11a14-211c-4606-8c64-071fc0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	2015-11-05	17 (1) 0,26 %
2	Документ из базы НЭУ 51dbc368-f458-4831-b9f9-3927c0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	2015-11-05	14 (1) 0,21 %
3	ҚАЗАҚСТАННЫҢ ЕКІНШІ ДЕҢГЕЙЛІ БАНКТЕРІНІҢ ЭКОНОМИКАНЫҢ НАҚТЫ СЕКТОРЫН НЕСИЕЛЕНДІРУІ БАНКТИҢ МЫСАЛЫНДА NARXOZ (NEU) (Кафедра Финансы)	Артығалиева С. Ф.	2016-05-24	14 (2) 0,21 %
4	Документ из базы НЭУ 51d6b0b4-4f10-4e72-81a1-071fc0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	2015-11-05	12 (1) 0,18 %
5	Документ из базы НЭУ 524e91f3-c050-41ce-9f3e-0707c0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	2015-11-05	11 (1) 0,17 %
6	Документ из базы НЭУ 51dbc84d-d978-4532-98c5-3925c0a80d1d.docx NARXOZ (NEU) (Information Technology Center)	na	2015-11-05	10 (1) 0,15 %

из интернета (7,71 %)

Все фрагменты найдены в глобальных интернет-ресурсах открытого доступа.

ПОРЯДКОВЫЙ НОМЕР	ИСТОЧНИК URL	ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
1	https://stud.kz/referat/show/6527	172 (10) 2,58 %
2	https://studopedia.org/11-7721.html	82 (5) 1,23 %
3	http://dereksiz.org/zejnetai-toleu-jonindegi-memlekettik-ortalifi.html	46 (5) 0,69 %
4	https://zharar.kz/id/1905	43 (4) 0,65 %
5	http://ansya.ru/health/lekcija-45-saat-sj-45-saat-obsj-45-saat-barli-saat-sani--135-s/pg-7.html	36 (2) 0,54 %
6	http://kzdocs.docdat.com/docs/index-53385.html?page=2	32 (2) 0,48 %
7	https://massaget.kz/laws/kaz/docs/Z990000365	29 (3) 0,44 %
8	http://adilet.zan.kz/kaz/docs/Z1300000094	22 (1) 0,33 %
9	http://adilet.zan.kz/kaz/docs/Z1800000204	17 (3) 0,26 %
10	https://23.astana-bilim.kz/enpf?lang=kz	15 (3) 0,23 %
11	https://online.zakon.kz/m/Document/?doc_id=31396173	12 (1) 0,18 %
12	https://www.kazatk.kz/material/nauka/practica/294-299.pdf	8 (1) 0,12 %

АНДАТПА

Бұл дипломдық жұмыс несие алушының төлем қабілеттілігін бағалаушы жүйе құруға арналған. Несие алушылардың несиелік қабілеттілігін талдау және несие беру туралы шешім қабылдау несиелеу схемасының кезеңдерінің ең маңыздысы болып табылады. Қарыз алушының міндеттемелерін орындау қабілеттілігін бағалау үшін индикаторларды таңдау проблемасы банктік дамудың барлық кезеңінде өзекті болды және несиелік қабілеттілікті анықтау проблемасы болып табылады.

Дипломдық жұмыстың мақсаты: Жасалған анализ негізінде қазіргі нарықтағы несие алушының төлем қабілеттілігін бағалайтын жүйе құру.

Қойылатын тапсырмалар:

- қазіргі банктердің клиенттер туралы жинайтын ақпараттарын талдау;
- скорингтік жүйелерді талдау;
- төлем қабілеттілігін анықтайтын параметрлерді анықтау.

Дипломдық жұмыс келесі логикалық бөліктерден тұрады:

- қазіргі таңда елімізде несие беріп отырған екінші деңгейлі банктер несиені рәсімдеу барысында қажет ететін ақпараттарды анықтау және талдау;
- мен құрастырған жүйе – оның ерекшеліктері – яғни, өзге банктердің жүйелерінен артықшылықтары.

Дипломдық жұмыста өтінім қабылдаушы веб-парақшаны құру үшін Python – web программалау тілі және мәліметтер қорын құру сонымен қатар несие қабілеттілігін бағалаушы процедура құру үшін Django қолданылады.

АННОТАЦИЯ

Данная дипломная работа предназначена для создания системы оценки платежеспособности заемщика. Анализ кредитоспособности заемщиков и принятие решений по кредитованию является одним из важнейших этапов схемы кредитования. Проблема выбора показателей для оценки способности заемщика выполнять свои обязательства актуальна на всех этапах развития банковской деятельности и является проблемой определения кредитоспособности.

Цель дипломной работы: Создать систему оценки платежеспособности заемщика на текущем рынке на основе анализа.

Задания:

- анализ информации о клиентах, собранной современными банками;
- анализ скоринговых систем;
- определение параметров, определяющих платежеспособность.

Дипломная работа состоит из следующих логических частей:

- выявить и проанализировать информацию, необходимую банкам второго уровня, которые в настоящее время предоставляют кредиты в стране, в процессе подачи заявки на кредит;
- система, которую я разработал, - ее особенности, то есть преимущества перед системами других банков.

На дипломной работе используется Python - язык веб-программирования и база данных для создания веб-страницы заявителя, а также Django для создания процедуры оценки кредитоспособности.

ABSTRACT

This diploma project is intended to create a system for assessing the solvency of the borrower. Analysis of the creditworthiness of borrowers and decision-making on lending is one of the most important stages of the lending scheme. The problem of selecting indicators to assess the borrower's ability to fulfill its obligations is relevant at all stages of banking development and is a problem of determining creditworthiness.

The purpose of the diploma project: To create a system for assessing the solvency of the borrower in the current market based on analysis.

Tasks:

- analysis of customer information collected by modern banks;
- analysis of scoring systems;
- determination of the parameters determining solvency.

The diploma project consists of the following logical parts:

- identify and analyze the information needed by second-tier banks that currently provide loans in the country in the process of applying for a loan;
- the system that I developed is its features, that is, advantages over the systems of other banks.

The diploma project uses Python - a web programming language and a database for creating the applicant's web page, as well as Django for creating a credit assessment process.

МАЗМҰНЫ

КІРІСПЕ	15
1 ҚАРЫЗ АЛУШЫНЫҢ НЕСИЕЛІК ҚАБІЛЕТІН БАҒАЛАУДЫҢ ТЕОРИЯЛЫҚ НЕГІЗІ	17
1.1 Қарыз алушының несиелік қабілетін бағалаудың маңызы мен қажеттілігі. Клиент туралы ақпарат көздерін алу.	17
1.2 Несие алушының қаржылық жағдайын талдау	19
1.3 Қарыз алушының несиелік қабілетін бағалаудың әдістері	21
1.4 «Жеке тұлғалар» мемлекеттік деректер базасы	22
2 НЕСИЕЛІК БЮРОЛАР. НЕСИЕЛІК АҚПАРАТ БЮРОЛАРЫНЫҢ ТҮРЛЕРІ	23
2.1 Мемлекеттік несие бюросы	25
2.2 Бірінші несие бюросы	26
2.3 Зейнетақы төлеу жөніндегі мемлекеттік орталық	30
2.4 Несиелік тарихты қалыптастыру және ақпаратты пайдалану принциптері	31
2.5 Фотосуреттер мен жеке деректерді өңдеуге келісім беру	32
3 СКОРИНГТІК ЖҮЙЕ	34
3.1 Бағдарламалық қосымша	35
3.2 Базалық құрылым	41
3.4 Python және Django web программалары	45
ҚОРЫТЫНДЫ	47
ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ	48
Қосымша А	49
Қосымша Б	51
Қосымша В	53
Қосымша Г	55
Қосымша Д	56
Қосымша Ж	57
Қосымша Л	63

КІРІСПЕ

Несие алушылардың несие төлеу қабілетін зерттеу және бағалау қазіргі банктер мен микрокредиттік ұйымдар үшін маңызды шарттардың бірі болып табылады. Осылайша, несиелік талдаудың мақсаттары мен міндеттері- несие алушы азамат немесе азаматшаның несиені уақтылы және толық көлемде қайтару қабілетін анықтау болып табылады. Несие алушыға берілуі мүмкін несие мөлшері мен берудің шарттары банк қабылдауға дайын болатын тәуекел дәрежесіне байланысты болады. Бұл банктен клиенттің белгілі бір күнге төлем қабілеттілігін ғана емес, сонымен бірге оның қаржылық тұрақтылығының болашағын да бағалауды талап етеді. Қарыз алушының қаржылық тұрақтылығын объективті бағалау және несиелік операцияларға байланысты мүмкін болатын тәуекелдерді ескере отырып, банкке несие ресурстарын объективті басқаруға және пайда табуға мүмкіндік береді.

Қазіргі уақытта коммерциялық банктер несиелендіруді, нарықтық қатынастарды ынталандыруға, әлеуметтік өндірістің тиімділігін арттыруға, Қазақстан Республикасының егеменді мемлекетінің экономикасы мен қаржысын нығайтуға, ақша ұсынысының негізсіз өсуін шектеуге, инфляцияны болдырмауға және ұлттық валютаны (теңге) нығайтуға міндетті.

Несие - бұл белгілі бір мерзімге пайыздарды өтеу, қарызды өтеу мерзімін ұзарту және қарыз алушының несиелік міндеттемелерін өтеу түрінде көрсетіледі.

Банк заемдарды өтеу, мерзімі, тиімділігі, мақсатты пайдалану қағидаларына қатаң сәйкестікте және заңмен қарастырылған және өзара келісілген коммерциялық негізде несиелер береді. Кез келген банктің кірістілігі банк берген несиелердің қайтарымдылық дәрежесіне, яғни оның табыстылығына байланысты болады. Несиені мерзімінен кешіктіріп қайтару, дәлірек айтқанда уақытында қайтарылмауы банкке және оның қызметіне шығын әкеледі. Сондықтан банктер несиелік тәуекелді уақтылы басқаруға және де несиенің мерзімінде қайтарылып отыруын өте мұқият қадағалауға міндетті.

Несиелік тәуекел - бұл қарыз алушының несиені немесе ол бойынша пайыздарды уақтылы төлей алмауынан банктің зиян шегу көлемін көрсетеді. Сондықтан банк қарыз алушыға несие бермес бұрын, оның несиелік қабілетін (несие төлеу қабілетін) анықтауы қажет. Қарыз алушылардың несиелік қабілетін анықтау әр банктің және оның клиентінің несиелік саясатына байланысты.

Қарыз алушының несие қабілеттілігі- қарыз алушының несиені уақтылы және толық көлемде төлеу қабілеттілігін бағалауымен сипатталады. Клиенттің несиені уақтылы төлей алмауы көптеген факторлардан туындауы мүмкін, сондықтан банк несие туралы шешім қабылдағанға дейін клиенттің несиелік қабілетін талдайды. Бұл көрсеткіш банктің өтімділігіне әсер етеді.

Қарыз алушының несиелік қабілетін талдау кезінде келесі факторлар ескеріледі:

1.Несиелік қабілеттілік. Қарыз алушы несие берушінің атынан әрекет ететін тұлғаның жарғысы мен нұсқаулықтарын оқып шығуы керек, олар несие берушінің құқықтық қабілетін анықтайды.

2.Қарыз алушының іскерлік беделі. Несиелік мәміленің беделі қарыз алушының қарызды өтеуге дайындығын ғана емес, сонымен қатар келісімшарт бойынша міндеттемелердің орындалуын да білдіреді.

3.Табыс алу мүмкіндігі. Банк қарыз алушының несиені өтеу үшін жеткілікті қаражат таба алу қабілетін бағалауы керек.

Сонымен бірге, осы тақырыптың өзектілігі банктердің негізгі функцияларының бірі несиелеу болып табылатындығында және несиелік ресурстарға қазіргі сұраныс артып келе жатқандықтан, банктер несиелік тәуекелдерден сақтану және төмендету мақсатында қарыз алушының несиелік қабілеттінің әдістемелерін үнемі жаңартып отыруы керек.

Бұл тақырыпты зерттеудің мақсаты - қарыз алушының несиелік қабілетін талдаудың теориялық және практикалық негіздерін зерттеу.

Қойылған мақсатқа байланысты мәселелер келесі қадамдарда шешілуі керек:

1.Несие алушы азамат немесе азаматшаның несиелік қабілетін бағалау үшін талап етілетін шарттарды қарастыру.

2.Несие алушының финанстық жағдайын талдаудың әдістерін зерттеу.

3.Қазақстан Республикасында және шетелдердегі банктердің тәжірибесіндегі пайдаланылатын қарыз алушының несиелік қабілетін сараптау.

4.Несие алушының несиелік қабілетін бағалау бойынша жобаларды қарау.

Теориялық негіз - несие беруді ұйымдастыру бойынша шетелдік және отандық ғалымдардың ғылыми жұмыстары.

Бірінші және тарауда кәсіпорынның несиелік қабілеттілігін талдаудың теориялық және әдіснамалық негіздері, атап айтқанда, тәуекелдерді азайту жүйесіндегі несиелік қабілеттіліктің түсінігі, өлшемдері мен көрсеткіштері, оны бағалаудың заманауи әдістері және несие қабілеттілігі туралы ақпарат көздері көрсетілген.

Үшінші тарау арнайы бағдарламалық қамтамасыздандыруды талдау негізінде ағымдағы нарықтағы қарыз алушының төлем қабілеттілігін бағалау жүйесін құруға арналған.

1 ҚАРЫЗ АЛУШЫНЫҢ НЕСИЕЛІК ҚАБІЛЕТІН БАҒАЛАУ

1.1 Несие алушының төлем қабілетін бағалаудың маңызы мен қажеттілігі. Клиент туралы ақпарат көздерін алу

Несиенің қазіргі жүйесі- қарыз алушының несие төлеу қабілетін тұжырымдау әдістеріне негізделген. Қарыз алушының несие қабілеттілігі- коммерциялық банктің балансының өтімділігін сақтаудың бір тәсілі.

Несие беру процесі операциялардың уақтылы өтелмеуіне әкелетін көптеген қауіп факторларының әсерімен байланысты. Сондықтан, банктік несие беру оған әсер ететін факторларды алдын-ала зерттей отырып, несие қабілеттілігін анықтауды жеңілдетеді. Қарыз алушының несиелік қабілеттілігі – несие алушы азамат немесе азаматшаның қарызды толық көлемде, сонымен қатар, уақтылы төлеу мүмкіндігі.

Қарыз алушының несие қабілеттілігі мен оның несиелік қабілеттілігінің арасындағы айырмашылықтардың бірі- ол алдыңғы кезеңнің немесе кез-келген күннің төленбеуіне назар аудармайды, яғни ол болашақтағы несиеге қабілеттілікті болжайтындығын білдіреді. Клиенттің несие қабілеттілігін бағалау үшін қолданылатын ең ресми көрсеткіштердің бірі – өткен өміріндегі қарыз алушының төлем қабілетсіздігінің деңгейі. Егер несие алушының берешегі бар болса, ал қалдық өтімді болса және меншікті капиталдың мөлшері жеткілікті болса, банкке төлемдердің бір реттік кешіктірілуі клиенттің төлем қабілетсіздігі туралы қорытынды жасауға негіз болмайды. Несиеге қабілетті клиенттер банкке, жеткізушілерге, бюджетке ұзақ мерзімді төлемсіздікке жол бермейді.

Клиенттің несие деңгейі банктің нақты қарыз алушыға беретін нақты сомасына байланысты қарапайым (жеке) тәуекелдердің деңгейін көрсетеді. Қарыз алушының қаржылық тұрақтылығын объективті бағалау және несиелік операцияларға байланысты мүмкін болатын тәуекелдерді есепке алу банкке несиелік ресурстарды тиімді басқаруға және пайда табуға мүмкіндік береді.

Әлемдік және ұлттық банктік тәжірибе қарыз алушылардың төлем қабілеттілігінің келесі өлшемдерін ажыратады:

Қарыз алушының сипаттамасы, қаражатты тарту мүмкіндігі; ағымдағы қызмет барысында (қаржылық қабілеттілік), капиталда, несие кепілдігінде, несиені пайдалану шарттарында, бақылауда (қарыз алушы қызметінің заңдылығы, банктік стандарттар мен банктік талаптарға сәйкес) несиені өтеу үшін қаражат тарту мүмкіндігі бақылау органдарына берілген несиенің сипаты).

Несиені қайтару жауапкершілігі, азамат немесе азаматшаның мінездемесі, оның заңды тұлға ретіндегі беделі мен менеджерлердің беделі және несие объектісінің банктің несиелік саясатымен сәйкестігі болып табылады.

Несие алушының заңды тұлға ретіндегі беделіне оның көпжылдық тәжірибесі, экономикалық көрсеткіштердің салалық орташа көрсеткіштерге сәйкестігі, несиелік тарихы, іскери әлемдегі серіктестерінің (жеткізушілер, сатып алушылар, несие берушілер) беделі кіреді. Менеджерлердің беделі

олардың кәсіби құзыреттілігіне (оқыту, кәсіби тәжірибе), адамгершілік сапасына, жеке қаржылық және отбасылық жағдайына, оларды басқару құрылымына және банк арасындағы қатынастардың нәтижелеріне негізделеді.

Қарыз алу мүмкіндігі қарыз алушының несие алуға өтініш беруге, несие келісіміне қол қоюға немесе келіссөздер жүргізуге, яғни бизнес немесе бизнес өкілдерінің белгілі бір өкілеттіктерге ие екендігін білдіреді.

Қарыз алушының несие қабілеттілігінің негізгі критерийлерінің бірі - олардың несиені өтеу үшін күнделікті қызметі барысында қаражат жинау мүмкіндігі.

Қарыз алушының капиталы оның несие қабілеттілігінің маңызды критерийін бағалауда келесі екі шарт маңызды:

- оның жеткіліктілігі-орталық банктің акционерлік қордың (жарғылық капиталдың) ең аз мөлшері мен левераж коэффициентіне қойылатын талаптары негізінде талданады;

- несие беруге меншікті капиталдың қаншалықты жұмсалғандығы, яғни бұл банк пен қарыз алушы арасында тәуекелдің бөлінгендігін көрсетеді;

Капиталға салынған инвестиция неғұрлым жоғары болса, қарыз алушының несиелік тәуекеліне әсер ететін факторларды бақылау мүмкіндігі соғұрлым жақсы болады.

Несиелік кепілдік дегеніміз - қарыз алушының активтерінің құны және несиелік келісім-шартта көрсетілген қарызды өтеу үшін екінші ретті көздердің болуы (кепілдік, кепілдеме, кепілдеме, сақтандыру). Бұл қосымша көздер қарыз алушының қаржылық қиындықтар туындаған жағдайда банк алдындағы міндеттемелерін уақтылы орындауын қамтамасыз етеді. Кепілдеменің сапасы, кепілгердің, кепіл берушінің және сақтандырушының тұрақтылығы қарыз алушыдан қаражат жеткіліксіз болған жағдайда өте маңызды.

Несиелік операцияларды жүргізу жағдайларына елдегі, аймақтағы, саладағы және саяси факторлардағы ағымдағы немесе болжамды экономикалық жағдай кіреді. Бұл жағдай банктің сыртқы тәуекел дәрежесін анықтайды.

Соңғы критерий - бұл бақылау, ол келесі сұрақтарды ескереді:

1. Қарыз алушының қызметі мен несиелік шараларды жүзеге асырудың нормативтік-құқықтық базасы бар ма?

2. Жоспарланған заңнамалық өзгерістер (мысалы, салық заңнамасы) қарыз алушының жұмысына қалай әсер етеді?

3. Қарыз алушы мен несиелік өтінімде берілген несие туралы ақпарат банктің несие саясаты туралы құжатта көрсетілген банктің стандарттарына, сондай-ақ несие сапасын бақылайтын банктік реттеушілердің стандарттарына қаншалықты сәйкес келеді?

Банк клиентінің төлем қабілеттілігін бағалаудың бұл өлшемдері төлем қабілеттілігін бағалау әдістерінің мазмұнын анықтайды. Бұл әдістер мыналарды қамтиды:

- коммерциялық тәуекелдерді бағалау;
- қаржылық коэффициенттерді бағалау;
- басқаруды бағалау;

- несие алушының қаржылық тұрақтылығын бағалау;
- ақша ағындарын талдау;
- несие алушы туралы ақпарат жинау;
- жұмыс орнына бару арқылы, қарыз алушының жұмысын бақылау.

1.2 Несие алушының қаржылық жағдайын талдау

Мемлекеттік банкте несие алушының қаржылық жағдайы қарыз алушының төлем қабілеттілігін бағалаудың негізгі факторы болып табылады. Ол инвестициялардың құрылымымен және меншікті қаражаттар мен қарыз қаражаттарын пайдаланумен, сондай-ақ пайданы қарыз алушының тиімділігінің көрсеткіші ретінде пайданы алу, бөлу және тиімді пайдалануымен сипатталады.

Қарыз алушының төлем қабілеттілігін бағалау- оның қаржылық жағдайын бағалауды сипаттайды. Клиенттің қаржылық есептілігі мен құжаттары жағдайды бағалау үшін қолданылады.

Қарыз алушының қаржылық жағдайы оның несиелік қабілетіне байланысты:

- экономикалық талаптарға сәйкес төлем талаптарын уақтылы орындау;
- несиені өтеу;
- жұмысшылар мен жұмысшыларға жалақы төлеу;
- бюджеттік төлемдер және салық төлемдері.

Егер қарыз алушының қаржылық жағдайы жақсы болса, ол барлық міндеттемелерді қайтара алады. Қарыз алушылардың төлем қабілеттілігіне банктік талдау шектеулі және толық түрде жасалынады. Қарыз алушының несиелік қабілеттілігін талдау және оның тиімділігін, несиелік қабілеттілігін, экономикалық және қаржылық нәтижелерін жан-жақты бағалауға мүмкіндік беретін көрсеткіштер жүйесін қамтамасыз ету үшін көптеген әдістер қолданылады.

ҚР-да Қазақстан Республикасы Ұлттық банкіндегі перспективалық зертеулер орталығы әзірлеген, ҚР Ұлттық банк Директоратының 1997 жылғы 27 қыркүйек мәжілісінде №26 қаулы бойынша бекітілген «Қарыз алушының несиелік қабілетін банктердің талдауына байланысты әдістемелік нұсқауы», сонымен қатар АҚШ сияқты шет мемлекетте қолданылатын әдістемелерде өте танымал.

Бұл көрсеткіштердің ішіндегі ең маңыздысы баланстық өтімділікке байланысты көрсеткіштер.

Өтімділік - бұл компанияның төлемдердің барлық түрлері бойынша өз міндеттемелерін орындау қабілеттілігі. Өтімділік жалпы қарыз мен қолма-қол ақшаға байланысты (қолма-қол ақша, қарыз, тауарлы-материалдық қорлар). Баланстық өтімділік- активтердің ақша қаражаттарына айналу ықтималдығына байланысты жинақталған қаражаттарды міндеттемелерге көбейту арқылы анықталады.

Несие алушының несиелік қабілетін бағалауда бірнеше негізгі көрсеткіштер қолданылады:

1 Кесте – Несие алушының несиелік қабілетін бағалау көрсеткіштері

	Көрсеткіштер атауы	Есептеу жолдары
1.	Ағымдағы өтімділік коэффициенті	Ағымдағы активтер/ міндеттер
2.	Мерзімді өтімділік коэффициенті	Ағымдағы активтер- запастар/ ағымдағы міндеттеме
3.	Активтер рентабельдігі	Пайда /кәсіпорын активтері
4.	Рентабельдік сауда	өнімді сатудан түскен пайда /өнімді сатудан түскен түсім.
5.	Негізгі құралдар мен басқа айналымнан тыс активтер рентабельдігі	Пайда/ негізгі құралдар– айналымнан тыс активтер
6.	Меншікті капитал рентабельдігі	Пайда/ меншікті капитал
7.	Ұзақ мерзімді активтер рентабельдігі	Пайда/ұзақ мерзімді несие және зайымдар
8.	Қаржы тұтқасы коэффициенті	Заемдық қаражаттар/ меншікті қаржаттар
9.	Тіркеме активтерді өтеу коэффициенті	Тіркемелі активтер (ғимараттар және құрылыстар – амортизация)/ қамтамасыз етілген несиелер
10.	Қарызға қызмет ету коэффициенті	Пайда/ қарыз бойынша шығындар
11.	Қор қайтарымы	Жалпы пайда – түсім /активтер

Баланстық активтер өтімділік деңгейіне қарай үш топқа бөлінеді:

- ақшалай қаражаттар (тексеру және басқа банктік шоттар, қолма-қол ақша, басқа қолма-қол ақша, арнайы қорлардың күрделі жұмсалыуына пайдаланылмаған қаражат қалдығы);
- оңай орындалатын талаптар (жеңілдіктер, тауарлар, жөнелтілген тауарлар, оларға қаржылық көмек ретінде берілген қаражаттың қарызы, банктік қарыздар және компанияның арнайы қорларындағы қарыздарға шектеулер);
- орындауға оңай талаптар кешіктірілген жөнелту немесе жеткізу жұмыстарын қамтымайды, үш айға дейінгі қарыз қарызды өтеуге кірмейді;
- тез іске асатын тауарлық-материалдық қорлар (арзан тауарлық-материалдық қорлар, жұмыс істейтін жұмыс, болашақ шығындар, дайын тауарлар және басқа айналым қаражаттары).

Алайда, алашақ қарыздың сипатын ескеру қажет. Алашақтар туралы есеп қарыз алушының жағдайына әсерін немесе несие саясатыңыздың тиімділігін бағалау үшін қарыз туралы есепті пайдаланылады. Қазіргі уақытта жеке тұлғаларға арналған екінші деңгейдегі банктік несиелер тұтынушылық несиелер мен ипотекалық несиелерді қамтиды.

Тұтынушылық несиенің келесі түрлері қолданылады:

- автонесие;
- ұзақ мерзімде пайдалануға болатын тауарларды сатып алуға несиелер;

- тұрғын үйге берілетін несиелер;
- шұғыл қажеттіліктерге арналған несиелер (оқу, емдеу, демалу және т.б.).

1.3 Қарыз алушының несиелік қабілетін бағалаудың әдістері

Қарыз алушының несие қабілеттілігі- қарыз алушының несиені толық және уақтылы төлеу қабілеттілігін бағалауымен сипатталады. Банктер несие туралы шешім қабылдағанға дейін клиенттердің несие қабілеттілігін талдайды, өйткені несие алушының несиені уақытында төлей алмау қаупі көптеген факторларға байланысты болуы мүмкін. Бұл көрсеткіш банктердің сұраныс деңгейіне әсер етеді.

Несие алушының несиелік қабілеттілігін талдау кезінде келесі факторлар ескеріледі:

- Ақша алу мүмкіндігі. Қарыз алушыға соманы аудару кезінде ол несие берушінің және заемшының атынан әрекет ететін адамның заңды қабілетін анықтайтын нұсқаулықтарды білуі керек.

- Қарыз алушының іскерлік беделі. Несиелік мәміленің беделі қарыз алушының несиені қайтаруға дайындығын ғана емес, сонымен қатар келісімшарт бойынша барлық міндеттемелердің орындалуын білдіреді.

- Табыс табу мүмкіндігі. Банктер қарыз алушының қайтаруы керек қажетті сумманы төлеу үшін жеткілікті қаражат табу қабілетін бағалауы керек.

Қарыз алушының несиелік рейтингін жақсарту үшін:

- Қаржылық коэффициенттердің кеңейтілген жиынтығын пайдаланыңыз, өйткені шектеулі санды қолдану сіздің талдауыңыздың сапасын төмендетеді.

- Соңғы есепті кезең үшін емес, бірнеше есепті кезеңдердегі қарыз алушының қаржылық жағдайындағы өзгерістер динамикасын талдау.

- Қаржылық коэффициенттерді талдаумен қатар, несие қабілеттілігін талдау үшін пайдаланылатын клиенттердің ақша ағындарын талдау.

- Төлемдердің уақтылығы мен толық төленуі көп жағдайда несиені қайтару қабілеттілігіне тәуелді, сондықтан қарыз алушының қаржылық жағдайын дәстүрлі талдаумен қатар, несиені өтеу үшін ақша табу мүмкіндігі мен клиенттің моральдық қасиеттері бағаланды.

- Қарыз алушының басқа банктермен және қаржы институттарымен қарым-қатынас жасау тәжірибесін нақтылау үшін оның несиелік тарихын мұқият зерттеу керек.

Несиелік сапаны талдауда қосымша көрсеткіш ретінде келесі көрсеткіштер қолданылады: бизнес-тәуекел деңгейі, дебиторлық және кредиторлық берешектің жағдайы және олардың байланысы, басқаруды бағалау және басқа да әр түрлі коммерциялық банктердегі мерзімі өткен несие мерзімі мен мөлшері, т.б.

Несиелік тәуекелді басқару бойынша ұсыныстарға банк өкілдеріне бару, кепілзаттың жай-күйін және қамтамасыз ету сапасын тексеру (тексеру уақытын өзгерту), клиент тарапынан алаяқтықты анықтау, соның ішінде жеткілікті және

сапалы кепілзат алу және өзгерістерді анықтау, қарыз алушы өнімдері / қызметтері бағаларының динамикасын бақылау кіреді.

Индикаторлар жүйесіне тек қаржылық көрсеткіштер ғана емес, сонымен бірге басқару деңгейі, несие алушының іскерлік беделі, өнімдер мен қызметтердің сапасы сияқты қаржылық емес көрсеткіштер де кіруі керек. Шектеулі индикаторлар және т.б. Қарыз алушының несиелік рейтингісін есептеу әдістемесі сандық және қаржылық көрсеткіштермен толықтырылған және тәжірибенің көптеген қажеттіліктерін қанағаттандыра алады.

1.4 «Жеке тұлғалар» мемлекеттік деректер базасы

«Жеке тұлғалар» мемлекеттік дерекқоры - бұл Қазақстан Республикасындағы жеке тұлғалар туралы ақпаратты тіркеу және сақтаудың біртұтас жүйесі.

Жобаның мақсаты: Қазақстан Республикасының мемлекеттік органдарының ақпараттық жүйелерінде жеке сәйкестендіру нөмірлері бар адамдар туралы ақпарат беру, атап айтқанда, жеке тұлғаларды тіркеу жүйесінің барлық жүйелерін азаматтық хал актілерін тіркеу органдарына бірыңғай логикамен біріктіру, әлеуметтік сақтандыру, денсаулық сақтау, зейнетақы, салықтар және басқа қызметтер.

Мақсаты: ведомстволық жүйелердің ақпараттық ресурстарын біріктіру, олар өз өкілеттіктері шегінде және Қазақстан Республикасының қолданыстағы заңнамасына сәйкес барлық ведомстволық жүйелерді одан әрі дамыту үшін жеке тұлғаларды есепке алады және шығындарды үнемдеуді қамтамасыз етеді.

Міндеттері:

- жеке сәйкестендіру нөмірін (ЖСН) енгізу негізінде жеке тұлғаларды тіркеудің бірыңғай жүйесіне көшу;

- қолданыстағы және дамып жатқан ведомстволық есепке алу жүйелерінің сәйкес еместігі туралы мәселені шешу, Қазақстан Республикасының ұлттық тізілімдерін қалыптастыруға алғышарттар жасау;

- мемлекеттік органдардың ақпараттық базасын жаңарту және ақпараттың қайталануын болдырмау.

2 НЕСИЕЛІК БЮРОЛАР. НЕСИЕЛІК АҚПАРАТ БЮРОЛАРЫНЫҢ ТҮРЛЕРІ

Несиелік бюролар - бұл Қазақстан Республикасы халқының несие тарихы жөнінде барлық ақпаратты сақтайтын ұйымдар. Дәлірек айтқанда, әрбір қазақстандық банк, микроқаржыландыру және басқа несиелік агенттіктердің барлығы- несие бюроларына барлық қарыз алушылар мен мерзімі өткен несиелер туралы ақпарат береді. Мұндай ақпаратты кредиттік бюролар әр 15 жұмыс күн сайын немесе оданда жиі жаңартылып отырады.

Алайда, несиелік агенттіктер несиелік бюроларға ақпаратты жіберіп / жаңартып қана қоймай, сонымен қатар керісінше несиелік бюролардан ақпарат ала алады. Кез келген азамат немесе азаматша несие алуға жүгінген кезде, банктер мен басқа несие берушілер несие бюросынан несие туралы есепті сұрайды, бұл несие алуға өтініш берушінің бұрын алған несиесінің уақытында қайтарғандығы туралы ақпарат біледі.

Несие тарихы туралы ақпарат 10 жыл бойы сақталады және соңғы несиелік келісім жасалған күннен бастап жаңартылады.

Қолданыстағы заңға сәйкес, банк келісімшартқа қол қоюдан бұрын қарыз алушы мен кепілгерден екі келісімшартты талап етеді. Бұл ақпаратты несиелік бюроларға жібереді және бюродан мағұлмат алады.

Несиелік бюроларды дамыту үшін несиелік бюроларға ақпарат беретін ұйымдар:

- банктер, банк операцияларының жекелеген түрлерін жүзеге асыратын ұйымдар, коллекторлық агенттіктері мен микроқаржы ұйымдары;
- тауарларды немесе қызметтерді бірнеше айға бөліп, несие түрінде беретін немесе бөліп төлейтін жеке кәсіпкерлер немесе корпорациялар;
- жылжымайтын мүлікке құқықтарды тіркейтін мемлекеттік органдар;
- коммуналдық қызметпен қамтамасыз ететін табиғи монополия субъектілері;
- ақпаратты ұсыну шарты бойынша басқа да тұлғалар.

Несиелік есепті алушыларға жатады:

- банктер, банк операцияларының жекелеген түрлерін жүзеге асыратын ұйымдар, коллекторлық агенттіктері, микроқаржы ұйымдары;
- өнімдерді немесе қызметтерді несие беретін немесе төлем мерзімдерін ұзартатын жеке кәсіпкер немесе корпорация;
- ақпарат беру туралы шарт бойынша басқа тұлғалар;
- несие тарихының субъектісі;
- банктік кепілдік немесе кепілдік берілген тұлға;
- облигацияларды ұстаушының мүдделерін білдіруге келісілген облигация эмитентінің кредиттік есебі бойынша облигацияларды ұстаушының өкілі;
- аккредиттеу (уәкілетті) органы.

Несиелік бюролардың көптеген түрлері бар, олардың барлығы дербес жұмыс істейді. Алайда, несиелендірудің негізгі шешімдері үш ірі несиелік серіктестіктер: Equifax, TransUnion және Experian ақпаратына негізделген.

Жетекші несиелік бюролар несие операциялары туралы ақпаратты несие алушылары болып табылатын немесе азамат ретінде жұмыс істейтін компаниялар мен несие берушілерден алады. Несие беруші сізге шот-фактуралардың, төленбеген немесе төленбеген төлемдердің уақтылы төленгені туралы мезгіл-мезгіл хабарлайды. Несиелік бюролар мемлекеттік және аудандық соттардан салық міндеттемелері және банкроттық туралы ақпарат сияқты тиісті мемлекеттік құжаттарды алады. Бұл ақпаратты несиелік тексерулерде де қолдануға болады.

Несиелік бюролар азаматтардың жеке мәліметтерін қызметтерді алдын-ала көргісі келетін немесе оны жасауға заңды негізі бар компанияларға сата алады. Мысалы, несие алуға өтініш білдірген азамат немесе компания өтініш берушінің несие тарихын көре алады. Әдетте жұмыс берушілер мен пәтер иелері азаматтардың несиелік шоттарына жазбаша рұқсатынсыз кіре алмайды.

Несиелік бюро туралы ақпарат:

1. Жеке мәліметтер

Жеке ақпарат несиелік серіктестіктерге қарыз алушыны анықтауға және оны басқа қарыз алушылардан ажыратуға көмектеседі.

- аты, мекен-жайы, әлеуметтік сақтандыру нөмірі, туған жылы;
- алдыңғы мекен-жайы;
- еңбек кітапшасы.

2. Қоғамдық жауапкершілік

Несиелік бюролар сот органдарынан ақпарат жинайды. Бұған тек қаржылық шешімдер кіреді. Ақпараттың бұл түрі мыналарды қамтиды:

- банкроттық;
- салықтық жеңілдіктер;
- сатып алу құқығынан айыру;
- өндіру;
- өтемақы.

3. Сауда желісі

Бөлшек сауда желілері - бұл несиелер мен несие желілері туралы жазбалар және несиелік бюролар жинайтын маңызды ақпарат.

Олар әр несиенің өмірлік сипаттамаларын егжей-тегжейлі сипаттайды. Оларды несиелік серіктестіктерге және жалпы сипаттамаларға байланысты атауға болады:

- несие түрі;
- кредитордың аты;
- ашылу күні;
- соңғы әрекет күні;
- минималды төлем;
- пікірлер;
- есебіңіздегі жауапкершілік;

- алдыңғы төлем сомасы;
- несие қалдығы;
- максималды теңгерім;
- есептік жазба күйі;
- кешіктірілген сома.

Несиелік бюролар бірнеше дереккөздерден ақпарат алып оларды жинақтайды.

Несие беруші: Деректердің көп бөлігі несие берушіден келеді. Егер азамат бұрын ақша алған болса, несие беруші алынған несиені бір немесе бірнеше несиелік бюроларға хабарлайды. Егер азамат немесе азаматшада несие тарихы болмағандықтан несие алғысы келсе немесе проблемалық тарихты қалпына келтіру керек болса, несие бюросына есеп беретін несие берушілермен жұмыс жасалынады.

Үкімет: Мемлекеттік бюролар несие алушы азамат немесе азаматша туралы көптеген ақпараттарды мемлекеттік жазбалардан алады. Несие алатын азамат немесе азаматша туралы олардың несие тарихында банкроттық туралы өтініш берген немесе несие алушыға байланысты бұрын сот шешімдерінің бар-жоғы анықталады. Несие берушілер де бұл ақпаратты несиелік бюродан сатып алады.

Басқа ақпарат көздері: басқа ақпарат көздері «балама» несие есептерін құру үшін жиі қолданылады. Коммуналдық қызметтерге, мүшелікке және басқаларға төлейтін төлемдер несие алушының несие алу қабілетіне әсер етуі мүмкін.

Несие бюросы жинақтамайтын ақпараттар:

Ірі несиелік серіктестіктер төмендегілер туралы ақпарат жинамайды:

- сақталған (жинақталған) чектер;
- нәсілі;
- ұлты;
- жыны;
- саяси көзқарастары;
- кірісі.

Кейбір ақпарат несиелік серіктестерде сақталады, бірақ азаматтарға хабарланбайды, нақтырақ айтқанда тұлғаның несиелік есептерінде көрсетілмейді.

2.1 Мемлекеттік несие бюросы

Қазақстанда екі несиелік бюро жұмыс істейді - «Мемлекеттік несие бюросы» АҚ және «Бірінші несиелік бюро» ЖШС. Олардың жұмысының негізгі мақсаты, несие беруші мен қарыз алушы арасындағы қарым-қатынастардың айқындығын қамтамасыз ету болып табылады.

«Мемлекеттік несие бюросы» АҚ - бұл Қазақстан Ұлттық Банкінің коммерциялық емес еншілес компаниясы.

Бүгінгі таңда бес жүзден астам ұйымдар «Мемлекеттік несиелік бюро» АҚ несиелік тарихының мәліметтер базасына ақпараттар ұсынады:

- екінші деңгейдегі барлық банктер;
- барлық микроқаржы ұйымдары;
- банк операцияларының жекелеген түрлерін жүзеге асыратын барлық ұйымдар;
- коммуналдық қызмет көрсететін табиғи монополия субъектілері;
- басқа несиелік ұйымдар;
- ұялы байланыс операторлары (несие бойынша немесе бөліп-бөліп байланыс алған тұлғалар туралы ақпарат).

Аталған ұйымдар заңнамада белгіленген тәртіппен ақпараттарды «Мемлекеттік несиелік бюро» АҚ-ға жібереді және жаңартады. «Бірінші несиелік бюро» ЖШС-де ұйымдар ақпаратты өз қалауымен жібереді.

2020 жылдың қаңтарынан бастап, заңнамадағы өзгерістерге сәйкес, «Мемлекеттік несиелік бюро» АҚ барлық ломбардтарға, несиелік серіктестіктерге және несиелік беретін ұйымдарға, оның ішінде онлайн режимінде (қашықтан Интернет арқылы) ақпараттарды жіберуді бастайды. Бұл ұйымдардың кейбіреуі қазірдің өзінде «Мемлекеттік несиелік бюро» АҚ-ға ақпарат береді, бірақ 2020 жылдан бастап бұл ұйымдардың барлығы ақпаратты «Мемлекеттік несиелік бюро» АҚ-ға жібереді.

«Мемлекеттік несиелік бюро» АҚ-да Қазақстандағы барлық қарыз алушылардың несиелік тарихы бар мәліметтер базасы бар, сонымен бірге сақтандыру тарихы бар мәліметтер базасы сақталады.

Қазақстанның барлық сақтандыру (қайта сақтандыру) компаниялары осы дерекқорға ақпарат ұсынады.

2019 жылдан бастап автокөліктерге арналған сақтандыру полисі қазір онлайн режимінде, яғни Интернет арқылы шығарыла бастады. Олардың барлығы «Мемлекеттік несиелік бюро» АҚ деректер базасында сақталған.

«Мемлекеттік несиелік бюро» АҚ Қазақстан Республикасындағы несиелік және сақтандыру тарихының неғұрлым толық мәліметтер базасын құру мақсатында құрылды. Мұның бәрі Қазақстан Республикасы азаматтарының құқықтары мен мүдделерін қорғауға ықпал етеді және жалпы несиелік ұйымдардың тұрақтылығына әсер етеді.

2.2 Бірінші несиелік бюро

Бірінші несиелік бюро - бұл коммерциялық ұйым, жеке кәсіпкерлік субъектісі, оның негізгі қызметі- несиелік тарихты қалыптастыру, несиелік есептерін беру болып табылады. БНБ-ның барлық ерекшеліктері БНБ қолданылатын келесі нормативтік базаға байланысты, яғни:

- Қазақстан Республикасының Азаматтық кодексі.
- «Жауапкершілігі шектеулі серіктестіктер туралы» Қазақстан Республикасының Заңы.
- «Жеке кәсіпкерлік туралы» Қазақстан Республикасының Заңы.

– «Қазақстан Республикасындағы кредиттік бюролар және кредиттік тарихты қалыптастыру туралы» Қазақстан Республикасының Заңы.

– Қазақстан Республикасының нормативтік құқықтық актілері.

Нормативтік құқықтық актілер кредиттік бюроларды лицензиялау рәсімдері мен тәртібі мен ережелерін реттейді, несиелік тарихты қалыптастыру және оларды пайдалану жүйесіне қатысушылардың қызметінде ақпараттық процесті ұйымдастырады, қауіпсіздік жүйесін құру тәртібін реттейді, олардың электрондық жабдықтарына қойылатын минималды талаптарды белгілейді, несиелік тарихтың дерекқорын сақтайды және жеткізушілердің қызметін ұйымдастыруды реттейді ақпарат, несиелік бюролар және ақпараттық нарықтағы кредиттік есептерді алушылар қызметтер және т.б.

Тікелей қызметі кезеңінде БНБ жүйесі Қазақстанның 250-ден астам қаржы институттарын, соның ішінде қазіргі уақытта ақпарат жеткізушілері мен алушылары болып табылатын екінші деңгейдегі банктерді, ипотека, микроқаржыландыру, лизинг, инкассация ұйымдарын, даму институттарын қамтиды.

Ақпарат берушілер:

– банктер, банк операцияларының жекелеген түрлерін жүзеге асыратын ұйымдар, микроқаржы ұйымдары, инкассация агенттіктері;

– жүйелі сипаттамаларын Қазақстан Республикасының Үкіметі айқындайтын тауарлар мен қызметтерді несие бойынша сататын немесе кейінге қалдырылған төлемдерді жүзеге асыратын жеке кәсіпкерлер мен заңды тұлғалар (қазіргі уақытта қызметтердің келесі түрлерін ұсынатын жеке және заңды тұлғалар осындай адамдар деп танылады:

– тұрғын үй-коммуналдық қызметтермен (электр, газ, су, жылумен қамтамасыз ету;

– телефон қызметтерін ұсыну;

– ұялы байланыс қызметтерін ұсыну;

– теледидарлық кабельдік желі қызметтерін ұсыну;

– жылжымайтын мүлікке құқықтарды мемлекеттік тіркеуді жүзеге асыратын «Азаматтарға арналған үкімет» мемлекеттік корпорациясы;

– коммуналдық қызметтерді ұсынатын табиғи монополиялар субъектілері;

– ақпарат беру туралы келісім-шарт негізінде басқа тұлғалар.

1), 2), 3-1) және 4) тармақтарда көрсетілген ақпарат берушілер кредиттік тарих субъектісінің ол туралы ақпаратты кредиттік бюроларға жазбаша келісімінсіз кредиттік бюроға ақпарат беруге құқылы емес.

Бірінші бюросы тиісті келісімге қол қойған және несиелік тарих субъектілері туралы ақпарат беретін ақпарат берушілерден мәліметтер базасында ақпаратты жинайды және сақтайды. «Коммерциялық банктер туралы» заңға сәйкес, ақпарат жеткізушілері заң талаптарына сәйкес несиелік бюроларға өздерінің борышкерлері (несиелік тарих субъектілері) туралы ақпарат береді.

БНБ-мен жұмыс істеуге қызығушылық танытатын ұйымдардың саны халықаралық тәжірибенің ең жақсы стандарттарына сәйкес құрылған мәліметтер базасындағы құнды ақпараттың арқасында тұрақты өсу тенденциясына ие. Сондай-ақ, БНБ-мен ынтымақтастыққа резидент емес компаниялар мен басқа елдердің несиелік бюроларымен үлкен қызығушылық байқалады.

Несиелік есеп алушылар:

- банктер, банк операцияларының жекелеген түрлерін жүзеге асыратын ұйымдар, микроқаржы ұйымдары, инкассация агенттіктері;

- жүйелі сипаттамаларын Қазақстан Республикасының Үкіметі айқындайтын тауарлар мен қызметтерді несие бойынша сататын немесе кейінге қалдырылған төлемдерді жүзеге асыратын жеке кәсіпкерлер мен заңды тұлғалар (қазіргі уақытта қызметтердің келесі түрлерін ұсынатын жеке және заңды тұлғалар осындай адамдар деп танылады:

- тұрғын үй-коммуналдық қызметтермен (электр, газ, су, жылумен қамтамасыз ету;
- телефон қызметтерін ұсыну;
- ұялы байланыс қызметтерін ұсыну;
- теледидарлық кабельдік желі қызметтерін ұсыну;
- ақпарат алу туралы шарттар негізінде басқа адамдар;
- несиелік есеп алушылардың борышкері болып табылатын және несие туралы есепті тек өзі туралы алуға құқығы бар несие тарихының субъектісі;
- пайдасына банктік кепілдік немесе кепілдік берілген тұлға;
- облигацияларды ұстаушылардың мүдделерін білдіру туралы келісім жасалған облигация эмитентінің кредиттік есебіне қатысты облигация ұстаушылардың өкілі;
- уәкілетті орган (Қазақстан Республикасының Ұлттық Банкі).

1), 2), 3) тармақтарда көрсетілген кредиттік есептерді алушылар кредиттік тарих субъектісінің олар бойынша кредиттік есепті алуға жазбаша келісімінсіз несие есебін алуға құқылы емес.

Несиелік есептерді басқа адамдарға беруге жол берілмейді.

Коммерциялық банктер туралы заңға сәйкес кез-келген жеке / заңды тұлға (несие тарихының субъектісі) бір күнтізбелік жыл ішінде өзі туралы бір несие есебін алуға құқылы.

Бірінші несие бюросы ақпараттарды деректер базасында жинайды және сақтайды, оған қол қоюға несие тарихын қалыптастыру үшін несие тарихының субъектілері туралы ақпарат беретін тиісті келісімге қол қойған барлық кредиторлар қол жеткізе алады. «Коммерциялық банктер туралы» заңға сәйкес екінші деңгейдегі банктер, сондай-ақ банк операцияларының жекелеген түрлерін жүзеге асыратын басқа ұйымдар несиелік бюроларға өздерінің борышкерлері (несие тарихының субъектілері) туралы ақпарат ұсынуға міндетті. БНБ-мен келісім жасаған кез келген ұйым заңды және жеке тұлғалардың несиелік тарихы туралы ақпаратқа қол жеткізе алады.

Коммерциялық банктер туралы заңға сәйкес кез-келген жеке / заңды тұлға (несие тарихының субъектісі) бір күнтізбелік жыл ішінде өзі туралы бір несие есебін алуға құқылы.

Бірінші несие бюросынан қандай есептер алуға болады:

Жеке несие есебі - жеке тұлғаның ағымдағы және төленген несиелері туралы жиынтық ақпарат. Бұл қарыз алушының несие тарихы, оны азамат немесе азаматша несие алғысы келетін жағдайда көреді және оның негізінде ол несиелеу туралы шешім қабылдайды.

Жеке несие рейтингі - қарыз алушының несиелік тарихы мен басқа жеке деректері (мысалы, жасы, отбасы жағдайы және т.б.) негізінде несиелік қабілеттілігін бағалау. Рейтингік балл несие берушінің көзінде қарыз алушының тартымдылық деңгейін көрсетеді.

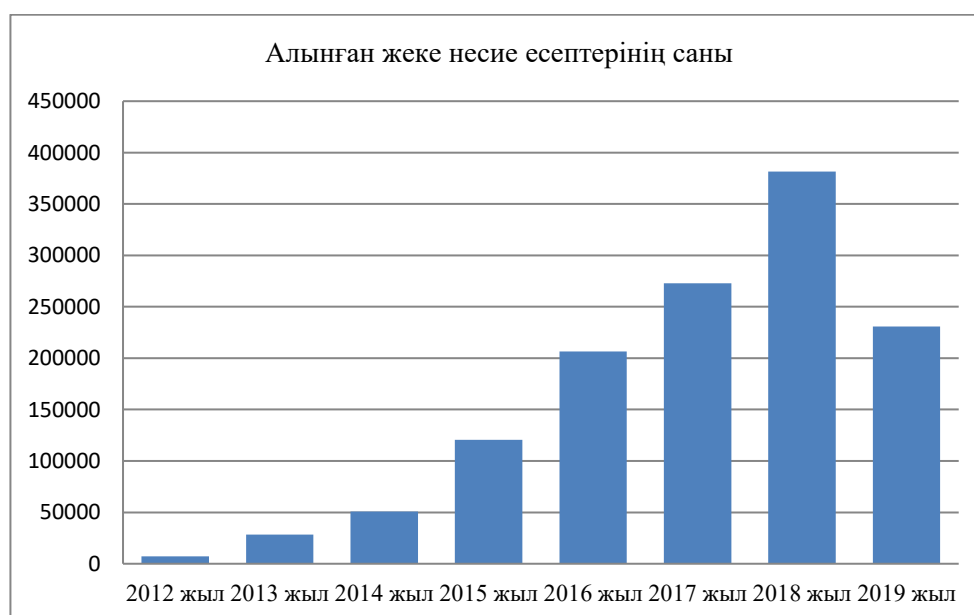
Несиелік тарихты бақылауға жазылу - несие тарихындағы өзгерістер туралы хабарламалар. Азамат немесе азаматшаға sms және электрондық пошта арқылы жаңа қарыздың жабылуы немесе ашылуы, мерзімі өткен несиелер, алаяқтардың несие тарихы туралы мәліметтерді жоспардан тыс пайдалану әрекеттері және тағы басқалары туралы ақпарат алынады.

Корпоративтік несие есебі - заңды тұлғаның ағымдағы және өтелген қарыздары туралы жиынтық ақпарат.

Бірінші несие бюросының 2019 жылғы 25 маусымдағы жеке кредиттік ақпараттары туралы есебі:

Бірінші несиелік бюро 2019 жылдың 25 маусымындағы жағдай бойынша жыл басынан бері берілген жеке несие есептерінің саны 230 700 құрады, бұл алынған жеке несие есептерінің саны болған кезде, 2018 жылдың дәл осы кезеңіндегі берілген жеке несие есептерінің мөлшерінен 1,2 есе жоғары, ол кездегі есеп бойынша 188 688.

Кесте 2 БНБ- дан алынған несие есептерінің саны



Сонымен бірге, бүгінде несие тарихын білудің ең танымал тәсілдері - Egov.kz электронды үкімет порталы және «Азаматтарға арналған үкімет» мемлекеттік корпорациясы» КАҚ және «Қазпочта» АҚ Қазақстанның барлық аймақтарындағы филиалдары. Бұл әдістердің танымалдылығы қарапайым нәтижеге, қолайлылыққа және жылдамдыққа байланысты. 2019 жылдың басынан бастап электронды үкімет порталы арқылы 204 760 дербес несие есептері, 8359 «Қазпочта» АҚ филиалдары және 7 190 «Азаматтарға арналған үкімет» мемлекеттік корпорациясы филиалдары арқылы шығарылды.

2.3 Зейнетақы төлеу жөніндегі мемлекеттік орталық

Зейнетақы төлеу жөніндегі мемлекеттік орталық - заңды тұлға құрмай-ақ, кәсіпкерлік және өзге де қызметті жүзеге асыратын заңды және жеке тұлғалардың міндетті жарналары есебінен зейнетақы алуға құқығы бар азаматтарға зейнетақы төлейтін заңды тұлға, бұндай жарнамалардың мөлшерін Республикалық бюджет туралы заңда қарастырып бекітеді. Орталық міндетті зейнетақы жарналарының есебін жүргізеді және олардың төлеушілерден жинақтаушы зейнетақы қорларына аударылуын бақылайды, сондай-ақ азаматтарды жеке әлеуметтік кодтармен (ӘДК) қамтамасыз етеді.

Заңнамада бекітілген жайдайларда, республикалық бюджеттен зейнетақы төлемдеріне бөлінген қаражаттан зейнетақы алуға құқығы бар азаматтарға, зейнетақы төлемдерін жүзеге асыратын, міндетті зейнетақы жарналарын есепке алуды және оларды төлеушілерден жинақтаушы зейнетақы қорларына аударуды жүзеге асыратын азаматтарға және сондай-ақ дербес код берген заңды тұлға.

Төлем түрлері мен қаржыландыру көздері

Зейнетақы мен жәрдемақыны уақтылы төлеу- зейнетақы төлеу жөніндегі мемлекеттік орталықтың негізгі міндеті. Қаржыландыру республикалық бюджеттен (РБ) және Мемлекеттік әлеуметтік сақтандыру қорынан (МӘСҚ) жүзеге асырылады.

Республикалық бюджеттен:

- базалық зейнетақы төлемі;
- зейнетақы;
- мемлекеттік әлеуметтік төлемдер: мүгедектік туралы, асыраушысынан айрылу жағдайында, жасына байланысты шығарылатын.

Арнайы мемлекеттік жәрдемақы:

- ерекше ауыр еңбек жағдайларында жұмыс жасайтын адамдар, жерасты және ашық жерлерді қазумен айналысатын, ерекше зиянды жұмыстармен айналысатын тұлғалар (№1 тізімге сәйкес);
- қауіпті және ауыр еңбек жағдайларында жұмыс істейтін адамдар (№2 тізім бойынша);
- жаңадан өмірге келген сәбиге біржолғы жәрдемақы;
- бала 1 жасқа толғанға дейін бала күтімі бойынша жәрдемақы;

– мүгедек баланы (мүгедек балаларды), ұл бала (қыз баланы) асырап алған қамқоршысына, тәрбиелеп отырған анасына немесе әкесіне берілетін мемлекеттік жәрдемақы.

Арнайы мемлекеттік жәрдемақы:

- ҰОС қатысушылары мен мүгедектеріне;
- ҰОС- ның қатысушылары мен мүгедектеріне теңестірілген адамдар;
- ҰОС қаза тапқан жауынгерлердің соңынан қалған жесірлері;
- қайтыс болған Ұлы Отан соғысы мүгедектерінің және мүгедектердің әйелдері (күйеуі);
- «Қазақстанның ғарышкер-ұшқышы» құрметті атағына ие адамдар;
- Кеңес Одағының Батырлары, Социалистік Еңбек Ерлері, Үш дәрежелі Даңқ орденінің иегерлері, Үш дәрежелі Еңбек Даңқы ордендерінің иегерлері,
- қайтыс болған әскери қызметкерлердің отбасылары;
- ҰОС 1941 жылғы 22 маусымнан- 1945 жылғы 9 мамырға дейін кемінде алты ай жұмыс істеген (әскери қызметте болған адамдар);
- кезінде ордендермен және медальдармен марапатталған адамдар,
- 16 жасқа дейінгі мүгедек балалар;
- Чернобыль апатының салдарын жоюға қатысушылар;
- 1, 1,2,3 топтағы мүгедектер;
- алтын алқа, күміс алқа тағылған немесе бұрын «Батыр ана» атағын және 1 және 2 дәрежелі «Даңқ анасы» ордендерімен марапатталған көп балалы аналар;
- көп балалы отбасылар- бірге тұратын төрт немесе одан да көп көмелетке толмаған балалары бар;
- саяси қуғын-сүргін құрбандары, мүгедектер немесе отставкадағы саяси қуғын-сүргін құрбандары;
- ҚР тигізген ерен еңбегі үшін зейнетақы алатын адамдар.

2.4 Несиелік тарихты қалыптастыру және ақпаратты пайдалану принциптері

Несиелік тарих пен ақпаратты пайдалану келесі принциптерге негізделеді:

- мемлекеттің қатысуымен кредиттік бюроларға ақпарат беру және осы Заңның 25 (4) -бабында көзделген жағдайларды қоспағанда, несие тарихының субъектісінің келісімі;
- несие тарихындағы барлық субъектілердің теңдігі;
- ақпаратты, ақпараттық жүйелерді және несиелік тарих дерекқорларын тиімді жолмен қолдану;
- ақпараттың құпиялылығы;
- несиелік тарихтың деректер базасын және соған сәйкес ақпараттық жүйелерді сақтау;
- жеке тұлғалардың, дәлірек айтқанда азамат немесе азаматшалардың жеке өмірін, олардың құқықтары, бостандықтары мен азаматтар мен ұйымдардың заңды мүдделерін қорғау.

– несие тарихы деректер базасының қауіпсіздігі мен өзектілігі, сонымен қоса сенімділігі.

2.5 Фотосуреттер мен жеке деректерді өңдеуге келісім беру

1)Биометриялық деректер - дербес деректер субъектісінің физиологиялық және биологиялық сипаттамаларын сипаттайтын дербес деректер. Осыған сүйене отырып, біз оның тұлғасын жасай аламыз.

2)Дербес деректер - ол электрондық мәліметтерге, қағазға немесе басқа ақпарат құралдарына жазылатын және оларды анықтау үшін пайдаланылатын ақпарат.

3)Дербес деректерді бөлу - дербес деректерді жинауды, сақтауды, өзгертуді, толықтыруды, пайдалануды, таратуды, жекешелендіруді және жоюды тоқтата тұру жөніндегі іс-шаралар.

4)Дербес деректерді жинау - дербес деректерді қамтитын дерекқорға енгізу арқылы жеке деректерді жүйелеу шарасы.

5)Дербес деректерді жинау - дербес деректерді жинауға бағытталған әрекеттер.

6)Дербес деректерді жою - дербес деректерді қалпына келтіру мүмкін болмаған кезде орындалатын әрекеттер.

7)Жеке деректерді иесіздендіру- дербес деректердің субъектісіне байланысты, дербес деректерге меншік құқығын анықтай алмауына байланысты дербес деректердің мінез-құлқын жекешелендіру.

8)Дербес деректерді қамтитын деректер базасы (бұдан әрі - мәліметтер базасы) - дербес деректердің реттелген жиынтығы.

9)Дербес деректері бар дерекқордың иесі - ұлттық институттарға, жеке тұлғаларға немесе Қазақстан Республикасының заңнамасына сәйкес жеке деректері бар дерекқорға иелік ету, пайдалану және жою құқығын пайдалану.

10)Дербес деректерді сақтайтын деректер базасының операторлары- бұл жеке деректерді жинайтын, өңдейтін және қорғайтын мемлекеттік органдар, жеке немесе заңды тұлғалар.

11)Дербес деректерді қорғау - осы Заңда айқындалған мақсаттарда жүзеге асырылатын құқықтық, ұйымдастырушылық және техникалық іс-шаралар жиынтығы.

12)Дербес деректерді өңдеу жөніндегі іс-әрекеттер- дербес деректерді жинауға, сақтауға, өзгертуге, қосуға, пайдалануға, таратуға, жекешелендіруге, бөлуге және жоюға арналған іс-шаралар.

13)Дербес деректерді пайдалану- иелер, операторлар және үшінші тұлғалар қызметінің мақсаттарына қол жеткізу үшін жеке мәліметтермен бірге пайдаланылатын іс- әрекеттерді пайдалану.

14)Дербес деректерді сақтау- жеке деректердің тұтастығын, құпиялылығы мен қол жетімділігін қамтамасыз ету жөніндегі іс-шаралар.

15)Дербес деректерді тарату- дербес деректердің берілуін, соның ішінде бұқаралық ақпарат құралдары арқылы және басқа жолдармен қол жеткізуді қамтамасыз ететін іс-шаралар.

16)Дербес деректер субъектісі- жеке мәліметтерге қатысты тұлғалар.

17)Үшінші тұлғалар - заңды тұлғалар, меншік иелері немесе операторлар болып табылмайтын, бірақ жеке деректерді жинауға, өңдеуге немесе қорғауға байланысты құқықтық жағдайларға немесе қатынастарға қатысатын тұлғалар.

Қазақстан банкіне онлайн-өтініш жібере отырып, азаматтар өз еркімен және өз мүдделерімен әрекет етеді және Банкке және / немесе Банкпен заңмен рұқсат етілген тәртіпте берілген жеке деректерді жинайды. Сіз өңдеуге, сақтауға және пайдалануға келісесіз. Сіз банктер ұсынатын қызметтерге / басқа қызметтерге және / немесе өтініштер / сауалнамалар туралы шешім қабылдайсыз.

Бұл келісім құжаттарды сақтау мерзімінің соңына дейін, оның ішінде жеке мәліметтер және / немесе Қазақстан Республикасының заңнамасында және / немесе ішкі құжаттарда көзделген электрондық құжаттар түрінде немесе банктен жазбаша растау түрінде жасалады. Бір ай ішінде біз өтінішті / анкетаны тексеруді аяқтау және / немесе тиісті шешімдер қабылдау және жеке деректерді өңдеуді жалғастыру үшін жеке деректерді өңдеуді жалғастыруға құқығымызды сақтаймыз. Мен боламын. Қолданыстағы заңнамаға түзетулер (бұл жағдайда жеке деректерді өңдеу заң талаптарына сәйкес) банктің ішкі құжаттарына енгізіледі.

Сондай-ақ, сіз азаматтармен немесе азаматтармен (тағайындалған адамдармен) кез-келген байланыс құралдары, оның ішінде хаттар, SMS, телефондар, электронды байланыс, электрондық пошта және басқа да байланыс құралдары арқылы тікелей байланыс орнатуға келісім беріледі.

3 СКОРИНГТІК ЖҮЙЕ

Несие алушының - жеке тұлғаның несиелік қабілеттілігін бағалау көбінесе скорингтік жүйелерді қолдану арқылы жүзеге асырылады. Скоринг моделі несиенің өтелмеу қаупіне әсер ететін факторларды талдайды және өтінімді мақұлдау немесе қабылдамау бойынша ұсыныстар береді. Несие алу үшін қарыз алушыдан алдымен анкета толтыру сұралады. Дәл осы мәліметтер негізінде бағалау жасалады.

Скоринг моделі микроқаржыландыру мен жедел несиелеуде кеңінен қолданылады, несие алуға арыз берген азамат немесе азаматшаның мәліметтерін қарап, шешім қабылдауға 15 минуттай уақыт кетіреді. Несиелік қабілетін тексеру үшін толтырылған өтінімдегі ақпарат арнайы бағдарламаға енгізіледі. Жүйе автоматты түрде ықтимал қарыз алушы көрсеткен деректерді статистикамен салыстырады. Сонымен, егер мәліметтер базасында ұқсас жастағы немесе кәсіпті адамдар несиені қайтармайтындығы туралы ақпарат болса, онда өтініш беру туралы шешім теріс болуы мүмкін. Мұндай жағдайларда, банк немесе микроқаржы ұйымы, әдетте, несие алушыға несие беруден бас тартады.

Скоринктің жүйенің ерекшеліктері:

- Шешім қабылдау жылдамдығы. Егер банк қызметкері қарыз алушының төлем қабілеттілігін талдаумен айналысса, бұл көп уақытты қажет етеді. Маман әр параметрді өздігінен тексеріп, алынған нәтижелерді қолмен енгізіп, қорытынды жасауы керек. Кредиттік рейтингті бағалаудың заманауи жүйелерінің көмегімен мәліметтер тез өңделеді, яғни шешім тез арада қабылданады.

- Объективтілік. Тіпті, тәжірибелі және білікті маман да есептеулерде қате жіберуі немесе клиентпен жеке қарым-қатынас жасауы үшін объективті пікір қалыптастыруы мүмкін. Скорингтік жүйе несие қабілеттілігінің объективті көрсеткіші болып табылады, өйткені ол автоматты түрде есептеледі. Банк қызметкері алгоритмнің жұмысына әсер ете алмайды.

- Қаржылық рентабельділік. Несиелік қабілеттілікті бағалау үшін скоринг моделін қолдану несиені қайтармаудың пайыздық үлесін айтарлықтай төмендетуі мүмкін. Бұл банктің кірісін арттырып қана қоймайды, сонымен қатар клиенттерге неғұрлым қолайлы тарифтерді ұсынуға мүмкіндік береді.

Кез-келген скорингтік модельді қолданудағы қорытынды балл бірқатар көрсеткіштерден тұрады. Ең алдымен, қарыз алушының төлқұжат деректері, тұрғылықты жері туралы ақпарат және басқа байланыс ақпараты тексеріледі. Содан кейін басқа факторларға талдау жасалады.

- Клиент туралы жеке мәліметтер. Скорингтік бағалау қарыз алушының отбасылық жағдайын және кәмелетке толмаған балалардың болуын ескереді. Соңғы жұмыс кезіндегі жұмыс өтілі де ескеріледі.

- Өтініш берушінің төлем қабілеттілігі. Рұқсат алу үшін несиені өтеуге жеткілікті қаражат қана емес, сонымен қатар тұрақты төлемдер бар екенін дәлелдеу керек.

– Несие тарихы. Клиенттердің несие қабілеттілігін бағалау кезінде бұрын алынған несиелер бойынша қарыздар мен бұзушылықтарды тексеру міндетті болып табылады. Банк келісім бойынша өтініш беруші туралы барлық қажетті ақпаратты көрсететін несиелік бюролардан (НБ) ақпарат ала алады. Сондай-ақ, жүйе қолданыстағы несиелер бойынша тұрақты төлемдердің болуын немесе болмауын ескереді.

Скоринг жүйесі барлық деректерді жеке тексереді және ықтимал қайшылықтарды анықтау үшін оларды бір-бірімен салыстырады. Көрсетілген ақпараттың дұрыстығын растау - бұл ықтимал қарыз алушының кірісі мен шығыстары, жұмыс орны және тұрғылықты мекен-жайы және т.б. арасындағы байланыстың болуы.

Нәтижеге сүйене отырып, жүйе шешім қабылдайды:

- Мақұлдау - жоғары рейтинг, өтінімді келесі деңгейге беруге болады.
- Бас тарту - өтініш беруші өте төмен балл жинаған, сондықтан өтінішті қарау тоқтатылады.

- Қосымша талдау қажет - жүйеде тиісті рейтингті орнату үшін мәліметтер жеткіліксіз. Бұл жағдайда банктің маманы өтініш берушінің сауалнамасын дербес зерттейді және ақпаратты жаңартады. Анықталынбаған жақтарды растау үшін өтініш берушіден қосымша құжаттар сұралуы мүмкін.

3.1 Бағдарламалық қосымша

Біз құрастырған бағдарламалық қосымшада клиенттердің несиені төлеу қабілетіне анализ жасап, несие беріп- бермеу туралы шешім қабылдайтын скорингтік жүйе 1-суретте көрсетілген.

Скоринг клиента

ИИН: 951210401855

Номер основного сотового телефона: 8747-123-45-67

Email- электронная почта: daniel@gmail.com

Создать и отправить код проверки

29:47

Код проверки: код

проверить код

Data from GBDFL

IIN	951210401855
Date of birth	1995, 12, 10
First name	Даниэль
Last name	Касенов
Middle name	Дани
Place of birth	Казахстан
Region	Туркестанская область
City	Туркестан
Street	Толы би
House	2
Flat	12
Marital status	не женат

1 Сурет – Негізгі бетте енгізілген мәліметтер бойынша ГБФЛ базасынан клиент туралы ақпарат шығару

Несие алу шарттары келесі қадамдармен жүзеге асырылады:

- Әр өтінімге бір келісімшарт нөмірі сәйкес келеді.
- Мәліметтер қорында өтінім берушінің бар-жоғы анықталады.
- Соған сәйкес мәліметтер жиынтығы жинақталып, арнайы скорингтік жүйеге келіп түседі.
- Несиеге өтінім берушінің мәліметтері «бағаланып» клиентке арнайы «жиынтық баға» сәйкестендіріледі.
- Несие беруші ұйымның «бағалау» саясатына сәйкес, скорингтік жүйе өтінім бойынша оң немесе теріс шешім шығарады.
- Оң шешім қабылданса, ары қарай келісімшартқа қол қою шаралары жүзеге асырылады.

Клиенттің төлем қабілетін тексеру мақсатында клиенттен келесі мәліметтер жинақталады:

- Персоналды мәліметтер (аты-жөні, туған күні, иин, мобильді телефоны, тұрғылықты мекенжайы).
- Байланысу жолдарына қатысты мәліметтер.
- Жұмыс орны жайлы мәліметтер.

Скоринг клиента

ИИН: 951210401855

Номер основного сотового телефона: 8747-123-45-67

Email- электронная почта: daniel@gmail.com

Создать и отправить код проверки

Код проверки: 3456

Data from GBDFL

IIN	951210401855
Date of birth	1995, 12, 10
City	Туркестан
Street	Толе би
House	2
Flat	12
Marital status	не женат

Контакт успешно подтвержден. Предварительная обработка в процессе. Пожалуйста, подождите

Close

проверить код

2 Сурет – Клиент нөмеріне СМС код жіберу арқылы клиентті растау және пре- скорингтік процессті бастау

Клиенттің террористер тізімінде, банктің «қара тізімінде» жоқ екені расталып, жасы 22- 73 арасында болса, пре- скоринг сәтті аяқталады. Ары қарай жалғастыруға клиенттің рұқсаты сұралады. Клиент келісімін берген соң, негізгі скоринг басталады. Ол максимум 15 мин уақытты қажет етеді. Осы ақпараттар 3 және 4- суреттерде көрсетілген.

Скоринг клиента

ИИН:

Номер основного сотового телефона:

Email- электронная почта:

Код проверки:

Data from GBDFL

IIN	951210401855
Date of birth	1995, 12, 10
First name	Даниэль
Last name	Касенов

Ваша заявка пре-одобрено. Хотите продолжить?
 Cancel OK

Street	Толе би
House	2
Fiat	12
Marital status	не женат

3 Сурет – Сәтті аяқталған пре- скоринг

Scoring system
127.0.0.1

Select cb_data_table to change...
Select application to change [...]
Select gcvp_data_table to cha...
Select gbdfl_data_table to cha...
Select regional_gcvp_table to...
Select application to change [...]

Ваша заявка принята. Скоринг начался. Пожалуйста подождите.
 X

ИИН:

Номер основного сотового телефона:

Email- электронная почта:

Код проверки:

Data from GBDFL

IIN	951210401855
Date of birth	1995, 12, 10
First name	Даниэль
Last name	Касенов
Middle name	Дани
Place of birth	Казахстан
Region	Туркестанская область
City	Туркестан
Street	Толе би
House	2
Fiat	12
Marital status	не женат

4 Сурет – Негізгі скорингтік процесс

Негізгі скоринг кезінде клиенттің НБ базасынан басқа банктердегі несиелері туралы төлем уақытын кешігіп немесе кешіктірмеуі және төлейтін айлық несие мөлшері, зейнетақы төлеу жөніндегі мемлекеттік орталық базасынан клиенттің айлық жалақы мөлшері туралы мәліметтер алынады. Егер несие алушы азамат немесе азаматшаның айлық табысының жартысы басқа банктердегі несие мөлшерінен артық болса, сонымен қатар несие тарихында мерзімінен кешіктіріп төлеу 60 күннен артық болмаса, клиентке несие беруге рұқсат етіледі.

Егер жарты айлық табысы мен басқа банктердегі айлық төлемінің айырмасы 1000 теңгеден артық болса, осы сумма 4 жылға, яғни 48 айға көбейтіліп, максималды несие мөлшері шығады. Басқа жағдайлардың бәрінде банк несие беруден бас тартады. Оны 5-7- суреттерден көрсе болады.

Запрос успешно завершён. Максимальная сумма кредита 3360000 тг

ИИН: 951210401855

Номер основного сотового телефона: 8747-123-45-67

Email- электронная почта: daniel@gmail.com

Создать и отправить код проверки

Код проверки: 3456

проверить код

Data from GBDFL

IIN	951210401855
Date of birth	1995, 12, 10
First name	Даниэль
Last name	Касенов
Middle name	Дани
Place of birth	Казахстан
Region	Туркестанская область
City	Туркестан
Street	Толе би
House	2
Flat	12
Marital status	не женат

5 Сурет – Сәтті аяқталған скоринг

Сеанс завершён

ИИН: 951210401855

Номер основного сотового телефона: 8747-123-45-67

Email- электронная почта: daniel@gmail.com

Создать и отправить код проверки

Код проверки: 3456

проверить код

Data from GBDFL

IIN	951210401855
Date of birth	1995, 12, 10
First name	Даниэль
Last name	Касенов
Middle name	Дани
Place of birth	Казахстан
Region	Туркестанская область
City	Туркестан
Street	Толе би
House	2
Flat	12
Marital status	не женат

6 Сурет – Сеанстың аяқталуы

Скоринг клиента

ИИН: 031205135677

Номер основного сотового телефона: 8747-123-45-67

Email- электронная почта: daniel@gmail.com

Создать и отправить код проверки

Код проверки: 6690

проверить код

Data from GBDFL

IIN	031205135677
Date of birth	2003, 12, 5
First name	Марианна
Last name	Марианна
Middle name	Марианна
Place of birth	Казахстан
Region	Карагандинская область
City	Karagandy
Street	Косаева
House	246
Flat	22
Marital status	не замужем

Запрос был отклонен.

Close

7 Сурет – Скоринг нәтижесінде банктің несие беруден бас тартуы

Егер несие алуға өтініш берген азамат немесе азаматшаның жасы 22-73 жас аралығында болмаған жағдайда пре- скорингтен өтпейді. Оны 8- суреттен көруге болады.

Скоринг клиента

ИИН:
031205135677

Номер основного сотового телефона:
8747-123-45-67

Email- электронная почта:
daniel@gmail.com

Создать и отправить код проверки

Код проверки:
6690

проверить код

Data from GBDFL

IIN	031205135677
Date of birth	2003, 12, 5
First name	Мадина
Last name	Даулетова
Middle name	Madina
Place of birth	Казахстан
Region	Карагандинская область
City	Karagandy
Street	Косаева
House	246
Flat	22
Marital status	не замужем

8 Сурет – Жас шектеуіне байланысты сәтсіз аяқталған пре- скоринг

Скоринг клиента

ИИН:
031205135677

Номер основного сотового телефона:
8777-777-77-77

Email- электронная почта:
madina@mail.ru

Создать и отправить код проверки

Код проверки:
КОД

проверить код

Поля обязательны для заполнения

9 Сурет – Барлық сұралынған ақпарат толтылуы керек

Басқа банктерге төленуі қажет несиені 60 күннен кешіктіргені белгілі болса, біздің банк бірден несие беруден бас тартады. Оны 10-11- суреттерден көруге болады.

Скоринг клиента

ИИН:
541114638690

Номер основного сотового телефона:
8700-125-36-78

Email- электронная почта:
gulim@mail.ru

Создать и отправить код проверки

Код проверки:
7732

проверить код

Data from GBDFL

IIN	541114638690
Date of birth	1954, 11, 14
City	Павлодар
Street	Байзакова
House	34
Flat	12
Marital status	не замужем

Контакт успешно подтвержден. Предварительная обработка в процессе. Пожалуйста, подождите

Close

10 Сурет – Клиент нөмеріне СМС код жіберу арқылы клиентті растау және пре-скорингтік процессті бастау

Скоринг клиента

ИИН:

Номер основного сотового телефона:

Email- электронная почта:

Создать и отправить код проверки

Код проверки:

проверить код

Data from GBDFL

IIN	541114638690
Date of birth	1954, 11, 14
First name	Гулим
Last name	Оспанова
Street	Байзакова
House	34
Flat	12
Marital status	не замужем

Ваша заявка пре-одоброено. Хотите продолжить?

Cancel OK

11 Сурет – Клиенттен жалғастыруға келісім сұрау

Запрос был отклонен. Банк не может дать вам кредит.

ИИН:

Номер основного сотового телефона:

Email- электронная почта:

Создать и отправить код проверки

Код проверки:

проверить код

Data from GBDFL

IIN	541114638690
Date of birth	1954, 11, 14
First name	Гулим
Last name	Оспанова
Middle name	Gulim
Place of birth	Казахстан
Region	Северо-Казахстанская область
City	Павлодар
Street	Байзакова
House	34
Flat	12
Marital status	не замужем

12 Сурет – Несиені уақытынан кешіктіріліп төлеген клиентке жауап

Егер несие алушы зейнетақы төлеу жөніндегі мемлекеттік орталық базасынан клиенттің айлық табысы туралы мәлімен ала алмаған жағдайда Regional_gcvr кестесінен сұраныс жіберіліп, клиенттің тіркелген облысындағы орташа жалақы мөлшері алынады. Осы ақпарат клиенттің жалақы мөлшері болып есептелінеді. Бұның мысалы 13-15 суреттерде көрсетілген.

Скоринг клиента

ИИН:

Номер основного сотового телефона:

Email- электронная почта:

Создать и отправить код проверки

Код проверки:

проверить код

Data from GBDFL

IIN	950505123465
Date of birth	1995, 5, 5
First name	Нуржан
Last name	Талғатов
Middle name	Нуржан
Place of birth	Казахстан
Region	Костанайская область
City	Костанай
Street	Ауэзов
House	57
Flat	22
Marital status	женат

29:33

13 Сурет – Клиенттің сұраныс беруі

Скоринг клиента

ИИН: 950505123465

Номер основного сотового телефона: 8700-125-36-78

Email- электронная почта: nurzhan@mail.ru

Создать и отправить код проверки

Код проверки: 3754

проверить код

Data from GBDL

IIN	950505123465
Date of birth	1995, 5, 5
First name	Нуржан
Last name	Талгатов
Street	Ауэзов
House	57
Flat	22
Marital status	женат

Ваша заявка пре-одоброено. Хотите продолжить?

Cancel OK

14 Сурет – Сәтті аяқталған пре- скоринг

Скоринг клиента

ИИН: 950505123465

Номер основного сотового телефона: 8700-125-36-78

Email- электронная почта: nurzhan@mail.ru

Создать и отправить код проверки

Код проверки: 3754

проверить код

Data from GBDL

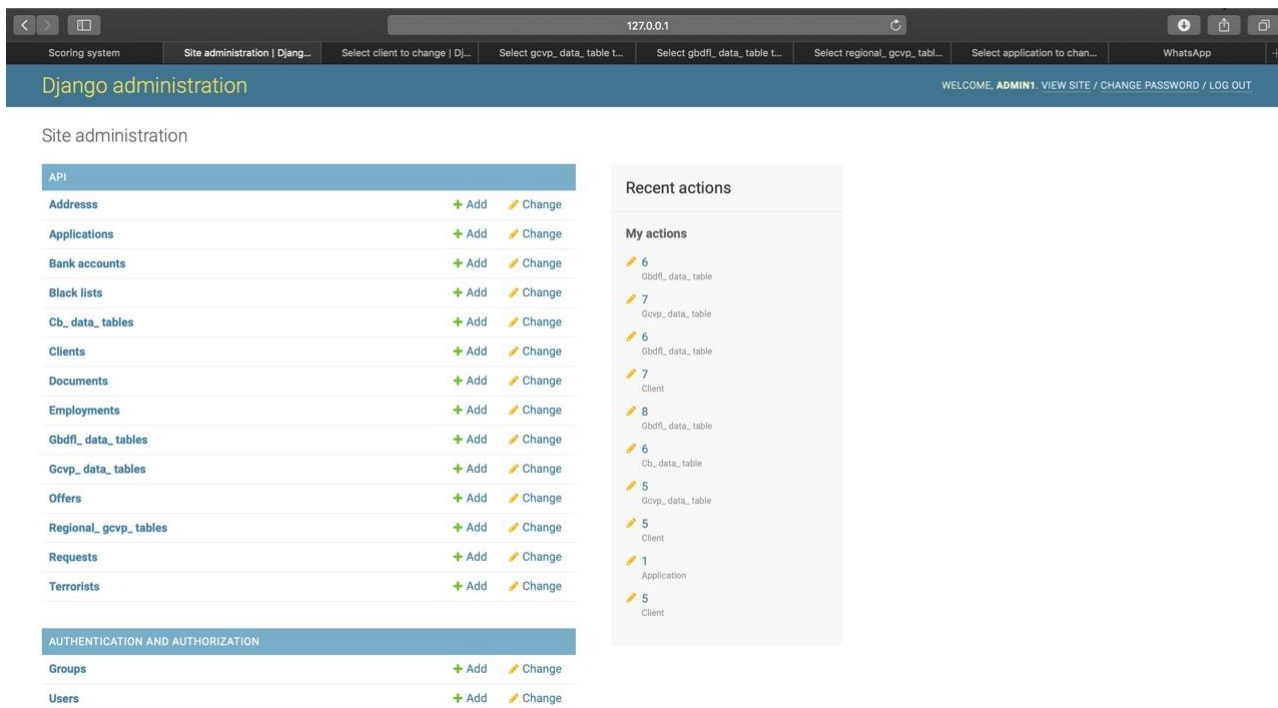
IIN	950505123465
Date of birth	1995, 5, 5
First name	Нуржан
Last name	Талгатов
Middle name	Нуржан
Place of birth	Казахстан
Region	Костанайская область
City	Костанай
Street	Ауэзов
House	57
Flat	22
Marital status	женат

Запрос успешно завершен. Максимальная сумма кредита 1920000 тт

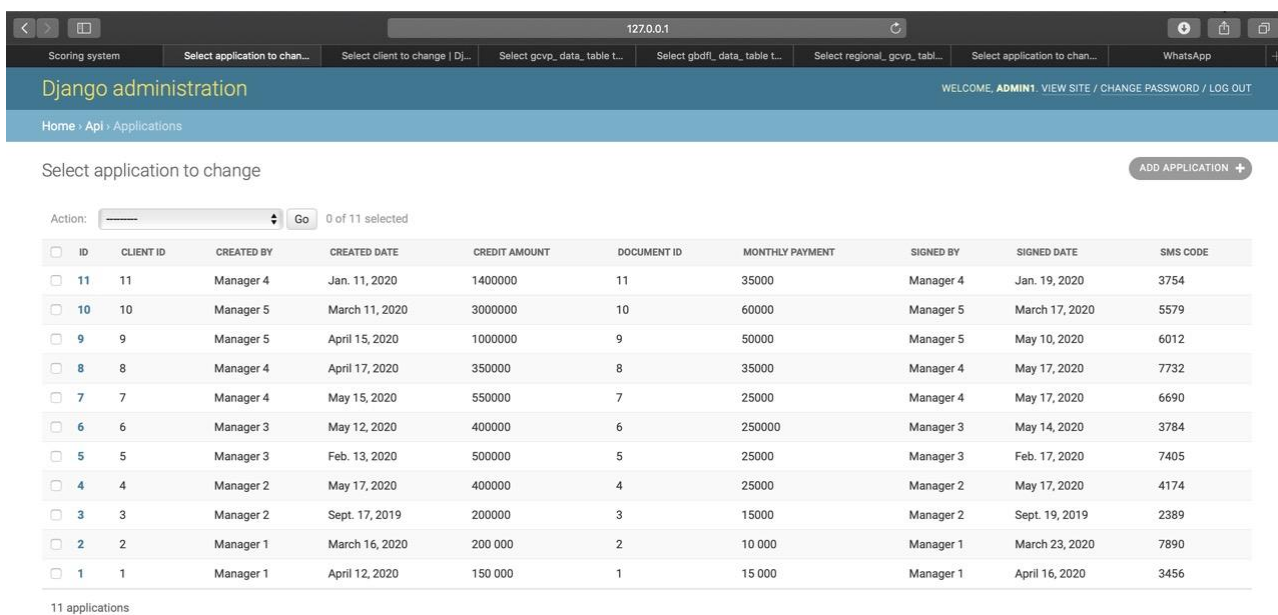
15 Сурет – Клиентке несиенің тағайындалуы

3.2 Базалық құрылым

Скорингтік жүйені құруға арналған барлық ақпараттар мен кестелер біздің базамызда сақталынған. База ретінде Jango- administration базалық құрылымын қолдандым. Қазір «Жеке тұлғалар» мемлекеттік деректер базасына, зейнетақы төлеу жөніндегі мемлекеттік орталық базасына, несие бюросына тікелей рұқсат жоқ болғандықтан, жүйенің қалай жұмыс істейтіндігін демонстрациялау үшін өз базамызда сондай ақпараттармен кестелер құрдық. 16-25- суреттерде барлық кестелер туралы детальді информация көрсетілген.



16 Сурет – Базада тіркелген барлық кестелер тізімі



17 Сурет – Application кестесі

127.0.0.1

Scoring system Select black list to change... Select client to change | Dj... Select gcvp_data_table t... Select gbdf_data_table t... Select regional_gcvp_tabl... Select application to chan... WhatsApp +

Django administration WELCOME, **ADMIN1** VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Api > Black lists

Select black list to change ADD BLACK LIST +

Action: Go 0 of 11 selected

ID	IIN
☐ 11	980420450207
☐ 10	990416820586
☐ 9	970504720586
☐ 8	960523969745
☐ 7	920318693678
☐ 6	860928806555
☐ 5	740427693058
☐ 4	901215473957
☐ 3	890325729586
☐ 2	770524302535
☐ 1	450215869495

11 black lists

18 Сурет – Банктің «қара тізімі»

127.0.0.1

Scoring system Select cb_data_table to c... Select client to change | Dj... Select gcvp_data_table t... Select gbdf_data_table t... Select regional_gcvp_tabl... Select application to chan... WhatsApp +

Django administration WELCOME, **ADMIN1** VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Api > Cb_data_tables

Select cb_data_table to change ADD CB_DATA_TABLE +

Action: Go 0 of 11 selected

ID	IIN	ACTUAL DPD	HISTORICAL DPD	EXTERNAL CONTRACT NUMBER	MONTHLY PAYMENT
☐ 11	950505123465	0	0	449731	20000
☐ 10	951210401855	0	0	0	30000
☐ 9	780510123456	0	0	238456	50000
☐ 8	960421964745	0	70	нет	0
☐ 7	861204650184	2	2	623093	45000
☐ 6	740328912888	0	15	нет	0
☐ 5	940612547685	0	0	237635	30000
☐ 4	980318457601	0	3	654276	20000
☐ 3	541114638690	0	60	нет	0
☐ 2	711118452595	0	4	нет	0
☐ 1	961113643856	0	0	873402	30000

11 cb_data_tables

19 Сурет – Несиелік бюро кестесі

Django administration

WELCOME, ADMIN1. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Api » Clients

Select client to change

ADD CLIENT +

Action: Go 0 of 11 selected

☐	ID	DATE OF BIRTH	FIRST NAME	IIN	LAST NAME	LATIN FIRST NAME	LATIN LAST NAME	MIDDLE NAME	PLACE OF BIRTH
☐	11	May 5, 1995	Нуржан	950505123465	Талғатов	Nurzhan	Talgatov	Нуржан	Костанай
☐	10	Nov. 13, 1996	Аружан	961113643856	Ерболова	Aruzhan	Yerbolova	Aruzhan	Oskemen
☐	9	Nov. 18, 1971	Махмуд	711118452595	Жасуланов	Makhmud	Zhasulanov	Makhmud	Shymkent
☐	8	Nov. 14, 1954	Гүлим	541114638690	Оспанова	Gulim	Ospanova	Gulim	Pavlodar
☐	7	Dec. 5, 2003	Мадина	031205135677	Даулетова	Madina	Dauletova	Madina	Karagandy
☐	6	June 12, 1994	Аяжан	940612547685	Насырова	Ayazhan	Nasirova	Ayazhan	Aktau
☐	5	March 28, 1974	Ерлан	740328912888	Мирасов	Erlan	Mirasov	Erlan	Aktobe
☐	4	Dec. 4, 1986	Улан	861204650184	Асанов	Ulan	Assanov	Ulan	Kyzylorda
☐	3	April 21, 1996	Арман	960421964745	Нуржанов	Arman	Nurzhanov	Arman	Kostanay
☐	2	May 10, 1978	Жанна	780510123456	Айбарова	Zhanna	Aibarova	Zhanna	Nur-Sultan
☐	1	Dec. 10, 1995	Даниэль	951210401855	Касенов	Daniel	Kassenov	Dani	Almaty

11 clients

20 Сурет – Базада тіркелген клиенттер туралы ақпараттар

Home » Api » Documents

Select document to change

ADD DOCUMENT +

Action: Go 0 of 11 selected

☐	ID	CLIENT ID	DOC NUMBER	DOC TYPE	ISSUER	VALID FROM	VALID TO
☐	11	11	554762390	удостоверение личности	МВД РК	Feb. 17, 2018	Feb. 17, 2023
☐	10	10	884932056	удостоверение личности	МВД РК	July 13, 2015	July 13, 2020
☐	9	9	672349001	удостоверение личности	МВД РК	Jan. 11, 2020	Jan. 11, 2025
☐	8	8	106548342	удостоверение личности	МВД РК	Feb. 13, 2018	Feb. 13, 2023
☐	7	7	553906510	удостоверение личности	МВД РК	Dec. 4, 2016	Dec. 4, 2021
☐	6	6	110453828	удостоверение личности	МВД РК	Jan. 21, 2020	Jan. 21, 2026
☐	5	5	256743895	удостоверение личности	МВД РК	Feb. 14, 2018	Feb. 14, 2024
☐	4	4	122985437	удостоверение личности	МВД РК	Feb. 11, 2018	Feb. 11, 2025
☐	3	3	455630867	удостоверение личности	МВД РК	Feb. 17, 2011	May 17, 2023
☐	2	2	342356485	удостоверение личности	МВД РК	Dec. 9, 2013	Dec. 9, 2020
☐	1	1	050433432	удостоверение личности	МВД РК	Feb. 16, 2016	Feb. 16, 2023

21 Сурет – Базада тіркелген әр клиенттің құжаттары

Select gbdfi_data_table to change

ADD GBDFI_DATA_TABLE +

Action: Go 0 of 11 selected

☐	ID	IIN	DATE OF BIRTH	FIRST NAME	LAST NAME	MIDDLE NAME	PLACE OF BIRTH	REGION	CITY	STREET	HOUSE	FLAT	MARITAL STATUS
☐	11	950505123465	May 5, 1995	Нуржан	Талғатов	Нуржан	Казахстан	Костанайская область	Костанай	Ауэзов	57	22	женат
☐	10	960421964745	April 21, 1996	Арман	Нуржанов	Арман	Казахстан	Костанайская область	Kostanay	Масанчи	66	7	не женат
☐	9	861204650184	Dec. 4, 1986	Улан	Асанов	Ulan	Казахстан	Кызылординская область	Kyzylorda	Суйинбай	478	34	не женат
☐	8	740328912888	March 28, 1974	Ерлан	Мирасов	Erlan	Казахстан	Западно-Казахстанская область	Aktobe	Жангельдина	45	1	разведен
☐	7	940612547685	June 12, 1994	Аяжан	Насырова	Ayazhan	Казахстан	Западно-Казахстанская область	Актау	Муканова	56	23	замужем
☐	6	031205135677	Dec. 5, 2003	Мадина	Даулетова	Madina	Казахстан	Карагандинская область	Karagandy	Косаева	246	22	не замужем
☐	5	541114638690	Nov. 14, 1954	Гүлим	Оспанова	Gulim	Казахстан	Северо-Казахстанская область	Павлодар	Байзакова	34	12	не замужем
☐	4	711118452595	Nov. 18, 1971	Махмуд	Жасуланов	Makhmud	Казахстан	Алматинская область	Алматы	Гоголя	34	12	женат
☐	3	961113643856	Nov. 13, 1996	Аружан	Ерболова	Aruzhan	Kazakhstan	Восточно-Казахстанская область	Усть Каменогорск	Абай	4	14	не замужем
☐	2	780510123456	May 10, 1978	Жанна	Айбарова	Жанна	Казахстан	Северо-Казахстанская область	Нур-Султан	Кабанбай батыр	3	14	замужем
☐	1	951210401855	Dec. 10, 1995	Даниэль	Касенов	Дани	Казахстан	Туркестанская область	Туркестан	Толе би	2	12	не женат

22 Сурет – «Жеке тұлғалар» мемлекеттік деректер базасы

Action: Go 0 of 10 selected

☐ ID	IIN	MONTHLY INCOME
☐ 10	961113643856	450000
☐ 9	711118452595	120000
☐ 8	541114638690	120000
☐ 7	031405135677	300000
☐ 6	940612547685	65000
☐ 5	740328912888	130000
☐ 4	861204650184	55000
☐ 3	960421964745	200000
☐ 2	780510123456	180000
☐ 1	951210401855	200000

10 gcvp_ data_ tables

23 Сурет – Зейнетақы төлеу жөніндегі мемлекеттік орталық базасы

Action: Go 0 of 14 selected

☐ ID	REGION	AVG INCOME
☐ 14	Туркестанская область	180000
☐ 13	Северо-Казахстанская область	190000
☐ 12	Павлодарская область	175000
☐ 11	Мангистауская область	155000
☐ 10	Кызылординская область	140000
☐ 9	Костанайская область	120000
☐ 8	Карагандинская область	145000
☐ 7	Западно-Казахстанская область	160000
☐ 6	Жамбылская область	100000
☐ 5	Восточно-Казахстанская область	150000
☐ 4	Атырауская область	170000
☐ 3	Алматинская область	230000
☐ 2	Актюбинская область	250000
☐ 1	Акмолинская область	150000

14 regional_ gcvp_ tables

24 Сурет – Regional базасы

Action: Go 0 of 10 selected

☐ ID	IIN
☐ 10	731114868311
☐ 9	902817529542
☐ 8	760412205739
☐ 7	910215502458
☐ 6	851204650284
☐ 5	810325720573
☐ 4	920415739574
☐ 3	581121040503
☐ 2	672305483710
☐ 1	740328912888

10 terrorists

25 Сурет – Террористердің базасы

3.3 Python және Django web программалары

Скоринг жүйесін жасау барысында, python программалау тілін және django фреймворкты қолдандым. Django өзінің қарапайымдылығымен, фронт

және бэк жақтарын бірге реализациялауға мүмкіндік беретіндігімен ерекшеленеді. Сонымен қатар, django-administration базасында кестелермен жұмыс істеу, ақпараттарды енгізу, өңдеу процесстері өте оңай жүзеге асады. Керекті барлық кестелерді models.py файлында құрастырып, оларды admin.py файлында тіркедім. Ол админ арқылы өзгерістер енгізуге рұқсат береді. Әр кестенің ішіндегі ақпараттарды алу үшін views.py файлында функциялар жазып, оларды жүйенің фронт бөлігіне жіберемін. index.html файлы негізгі бетке жауап береді. Қажет етілген барлық формалар мен ақпараттар осы жерде тәгтар арқылы бейнеленген. Жүйенің логикасы, яғни, шақырылатын функциялардың барлығы осы файлдың ішінде, <script> тәгі ішінде көрсетілген. Әр батырманың өз функционалы болғандықтан, олар сұралған кезде сәйкес методтар шақырылады. Егер, қызмет көрсетіліп отырылған клиент барлық сұралымдарға сәйкес болса, керекті ақпараттар алынып, олардың дұрыстығы расталса, клиентке несие тағайындалады және рұқсат етілген максимум несие мөлшері есептелінеді. Басқа жағдайлардың бәрінде жүйе несие беруден бас тартады. Бастапқы беттің визуалды әдемі болуына main.css файлы жауап береді. Ол жерде әрбір элементтің id, класс немесе тәг атына байланысты стилизациялау процессі жүреді.

ҚОРЫТЫНДЫ

Дипломдық жұмысқа қойылған мақсат: қарыз алушының несиелік қабілеттілігін және оның қаржылық жағдайын бағалаушы скорингтік жүйе құрастыру. Осы мақсатта біз қарыз алушылардың несиелік қабілеттілік ұғымын, критерийлерін, факторларын, несие қабілеттілігін бағалаудың ақпараттық базасын, сондай-ақ несие қабілеттілігін бағалау кезінде қолданылатын көрсеткіштерді зерттедік.

Сонымен, қарыз алушының несиелік қабілеттілігі - бұл оның болашақтағы қабілетін толық және уақтылы бағалауға, борыштық міндеттемелерді өтеуге, сонымен қатар нақты қарыз алушыға несие берген кезде банктің тәуекел дәрежесін анықтауға мүмкіндік беретін қаржылық және қаржылық емес көрсеткіштермен көрсетілген оның толық құқықтық және қаржылық сипаттамалары.

Ұсынылған дипломдық жұмыстың артықшылығы- мен құрастырған скорингтің бағдарлама көмегімен келесі нәтижелерге қол жеткізу:

- Клиенттің және өтінім қабылдаушы менеджердің уақытын үнемдеу.
- Нақты дерек көздеріне сүйену арқылы адами факторға байланысты жүйеде мәліметтердегі қателіктерді 100 пайызға дерлік жою.
- Нақты деректер негізінде регуляторлық отчеттарды қатесіз жіберу.

Келесі жолдарда көрсетілген мәліметтерді енгізу арқылы кез келген клиенттің несие толеу қабілеттілігін бағалауға болады:

- Персоналды мәліметтер (аты-жөні, туған күні, иин, мобильді телефоны, тұрғылықты мекенжайы).
- Байланысу жолдарына қатысты мәліметтер.
- Жұмыс орны жайлы мәліметтер.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Мишкин Ф.С. The Economics of Money, Banking and Financial Markets. 4 басылым, АҚШ, 201.
- 2 Бизнестегі есеп және талдау. Т А Тасмағанбетов Алматы 2002
- 3 Қаржы теориясы. Құлпыбаев С, Баязитова А Алматы мерей 2001ж
- 4 Методика финансового анализа. Шеремет А.А., Сайфулин Р.С. М.: Инфра-М, 2003.
- 5 Управление оборотным капиталом. Стоянова Е.С. Оқу құралы. М.: Перспектива, 1998. 127 с.
- 6 Инвестиции. Игошин Н.В М. Финансы, 2000г
- 7 Қазақстан Республикасының Азаматтық Кодексі (Жалпы бөлім).
- 8 «Акционерлік қоғамдар туралы» ҚР Заңы № 281 – 1 10.06.98 ж.
- 9 «Салықтар және басқа да бюджетке төленетін міндетті төлемдер туралы» ҚР Заңы №2223 24.04.95 ж. (өзгертілген және толықтырылған).
- 10 «Ұлттық банкі туралы» ҚР Президентінің Заң күші бар Жарлығы 1995 жылғы 30 наурыздағы № 2155.
- 11 «Кредиттік бюролар және кредиттік тарихты қалыптастыру туралы» ҚР Заңы №15 шілде 2004 ж.
- 12 «Кредитная деятельность банков в Казахстане». Давлетов М.Т. А.: Экономика, 2011.
- 13 Скоринговый метод оценки кредитоспособности частных лиц. Напольских М.А. // Банки Казахстана, № 4, 2013 г.
- 14 Процесс оформления ипотечных жилищных кредитов в банке. Павлова И.В. // Банковское дело, № 4, 2004 г.
- 15 Рамазанов С. Скоринговое кредитование в Казахстане. // Банки Казахстана, № 5, 2015г.
- 16 Система кредитного бюро в Казахстане. Расширение доступа в кредитным ресурсам – успех сбалансированного подхода. // Банки Казахстана, № 9, 2013 г.
- 17 <https://mail.kz/kz/news/economic/bankterdin-kyzmetteri-men-rol-i-kurstyk-zhumys#hcq=SSofhZr>
- 18 http://adilet.zan.kz/eng/docs/Z040000573_
- 19 <https://developer.here.com/documentation/>
- 20 <https://www.w3schools.com>

Қосымша А

models.py- кестелер мен сызбалар

```
# main.css  admin.py  models.py X  views.py  serializers.py  urls.py  index.html
scoring > api > models.py > Application
1  from django.db import models
2
3  # Create your models here.
4
5  class Client(models.Model):
6      date_of_birth = models.DateField()
7      first_name = models.CharField(max_length=255)
8      IIN = models.CharField(max_length=12, null=False)
9      last_name = models.CharField(max_length = 255)
10     latin_first_name = models.CharField(max_length = 255)
11     latin_last_name = models.CharField(max_length = 255)
12     middle_name = models.CharField(max_length = 255)
13     place_of_birth = models.CharField(max_length = 255)
14
15     def __str__(self):
16         return '{}'.format(self.id)
17
18 class Address(models.Model):
19     city = models.CharField(max_length = 255)
20     client_id = models.ForeignKey(Client, on_delete=models.CASCADE)
21     flat = models.CharField(max_length = 255)
22     house = models.CharField(max_length = 255)
23     region = models.CharField(max_length = 255)
24     request_id = models.CharField(max_length = 255)
25     def __str__(self):
26         return '{}'.format(self.id)
27
28
29
30 class BankAccount(models.Model):
31     account_number = models.CharField(max_length = 255)
32     bank_name = models.CharField(max_length = 255)
33     BIC = models.CharField(max_length = 255)
34     BIN = models.CharField(max_length = 255)
35     client_id = models.ForeignKey(Client, on_delete= models.CASCADE)
36
37     def __str__(self):
38
39
40
41
42
43
44
45
46
47
48
49 class Employment(models.Model):
50     client_id = models.ForeignKey(Client, on_delete= models.CASCADE)
51     income_amount = models.CharField(max_length = 255)
52     income_type = models.CharField(max_length = 255)
53     request_id = models.CharField(max_length = 255)
54
55     def __str__(self):
56         return '{}'.format(self.id)
57
58
59 class Document(models.Model):
60     client_id = models.ForeignKey(Client, on_delete= models.CASCADE)
61     doc_number = models.CharField(max_length = 255)
62     doc_type = models.CharField(max_length = 255)
63     issuer = models.CharField(max_length = 255)
64     valid_from = models.DateField()
65     valid_to = models.DateField()
66     def __str__(self):
67         return '{}'.format(self.id)
68
69
70 class Application(models.Model):
71     client_id = models.ForeignKey(Client, on_delete= models.CASCADE)
72     created_by = models.CharField(max_length = 255)
73     created_date = models.DateField()
74     credit_amount = models.CharField(max_length=255)
75     document_id = models.ForeignKey(Document, on_delete=models.CASCADE)
76     monthly_payment = models.CharField(max_length=255)
77     signed_by = models.CharField(max_length=255)
78     signed_date = models.DateField()
79     sms_code = models.CharField(max_length = 4, default=1111)
80     def __str__(self):
81         return '{}'.format(self.id)
82
83
84 class Request(models.Model):
85     client_id = models.ForeignKey(Client, on_delete= models.CASCADE)
86     request_body = models.CharField(max_length=255)
87     request_date = models.DateField()
88     request_status = models.CharField(max_length=255)
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

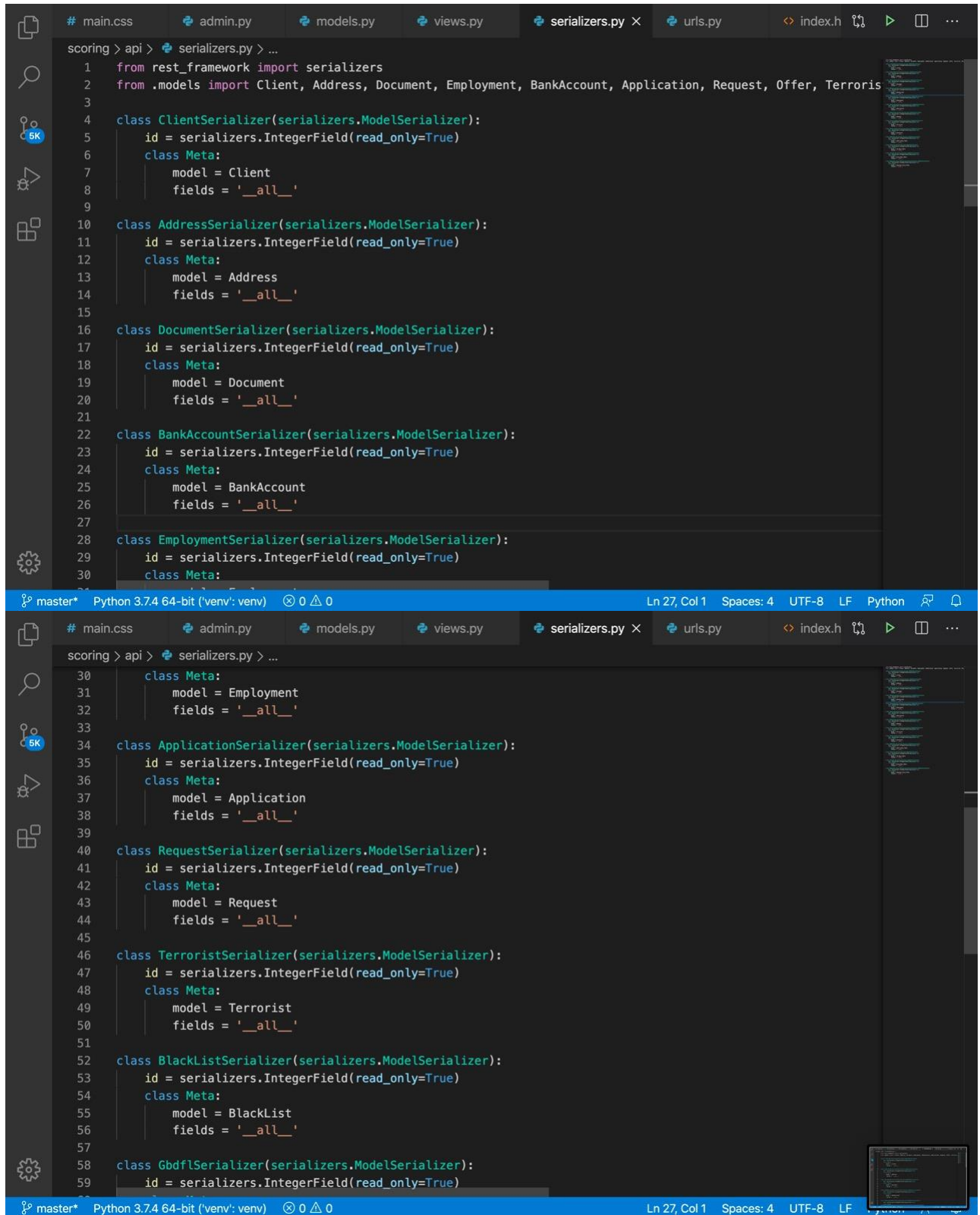
Қосымша А жалғасы

models.py- кестелер мен сызбалар

```
# main.css admin.py models.py X views.py serializers.py urls.py index.html
scoring > api > models.py > Application
76 request_status = models.CharField(max_length=255)
77 request_type = models.CharField(max_length=255)
78 response_body = models.CharField(max_length=255)
79 def __str__(self):
80     return '{}'.format(self.id)
81
82 class Offer(models.Model):
83     application_id = models.ForeignKey(Application, on_delete= models.CASCADE)
84     chosen = models.CharField(max_length=255)
85     created_date = models.DateField(max_length=255)
86     offer_amount = models.CharField(max_length=255)
87     offered_monthly_payment = models.CharField(max_length=255)
88     valid_from = models.DateField()
89     valid_to = models.DateField()
90     def __str__(self):
91         return '{}'.format(self.id)
92
93 class Terrorist(models.Model):
94     IIN = models.CharField(max_length=12, null=False)
95     def __str__(self):
96         return '{}'.format(self.id)
97
98 class BlackList(models.Model):
99     IIN = models.CharField(max_length=12, null=False)
100     def __str__(self):
101         return '{}'.format(self.id)
102
103 class GbdfL_Data_Table(models.Model):
104     IIN = models.CharField(max_length=12, null=False)
105     date_of_birth = models.DateField()
106     first_name = models.CharField(max_length=255)
107     last_name = models.CharField(max_length = 255)
108     middle_name = models.CharField(max_length = 255)
109     place_of_birth = models.CharField(max_length = 255)
110     region = models.CharField(max_length = 255)
111     city = models.CharField(max_length = 255)
112     street = models.CharField(max_length = 255)
113
114     street = models.CharField(max_length = 255)
115     house = models.CharField(max_length = 255)
116     flat = models.CharField(max_length = 255)
117     marital_status = models.CharField(max_length = 255)
118     def __str__(self):
119         return '{}'.format(self.id)
120
121 class Ch_Data_Table(models.Model):
122     IIN = models.CharField(max_length=12, null=False)
123     monthly_payment = models.IntegerField()
124     actual_dpd = models.IntegerField()
125     historical_dpd = models.IntegerField()
126     external_contract_number = models.CharField(max_length = 25, default='0')
127     def __str__(self):
128         return '{}'.format(self.id)
129
130 class Gcvp_Data_Table(models.Model):
131     IIN = models.CharField(max_length=12, null=False)
132     monthly_income = models.IntegerField()
133     def __str__(self):
134         return '{}'.format(self.id)
135
136 class Regional_Gcvp_Table(models.Model):
137     region = models.CharField(max_length=30)
138     avg_income = models.IntegerField()
```

Қосымша Б

serializers.py- кестелерді python тілінің мәлімет типтеріне айналдыру

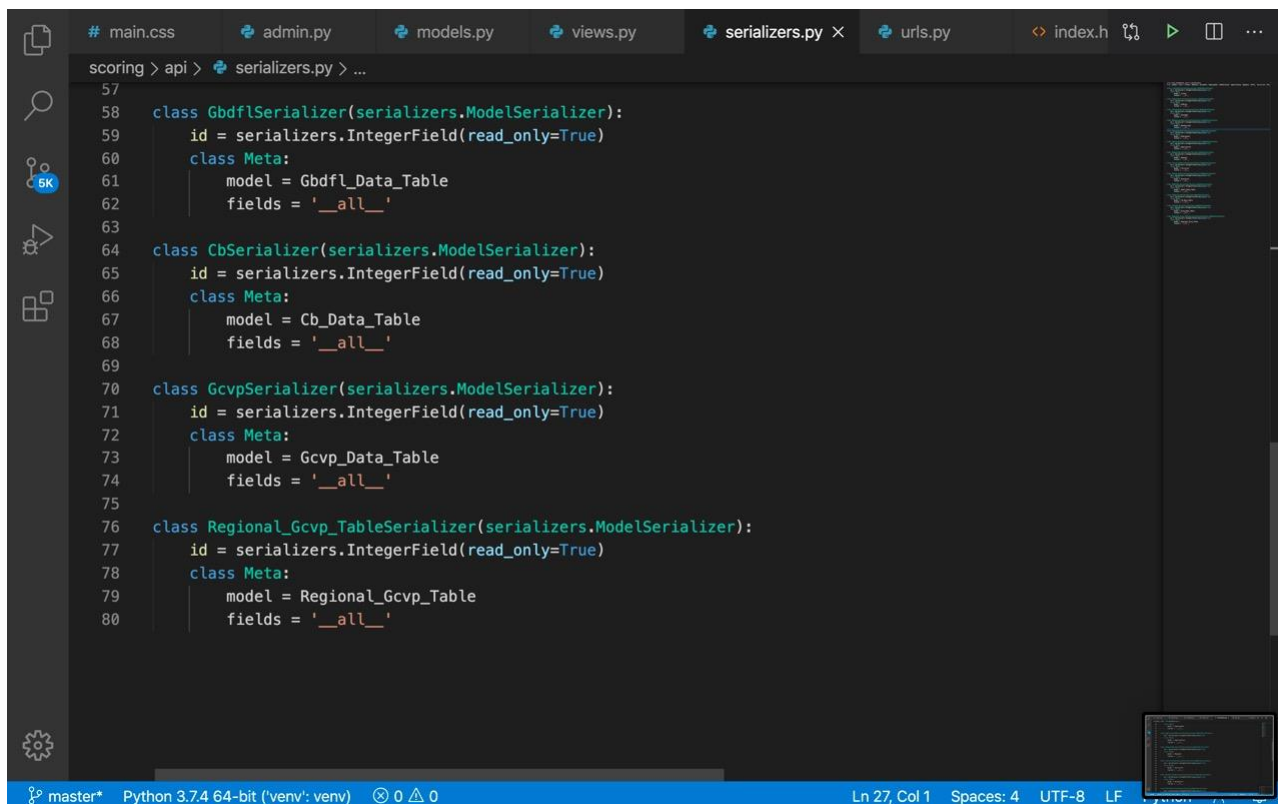


```
# main.css  admin.py  models.py  views.py  serializers.py x  urls.py  index.h  ...
scoring > api > serializers.py > ...
1  from rest_framework import serializers
2  from .models import Client, Address, Document, Employment, BankAccount, Application, Request, Offer, Terroris
3
4  class ClientSerializer(serializers.ModelSerializer):
5      id = serializers.IntegerField(read_only=True)
6      class Meta:
7          model = Client
8          fields = '__all__'
9
10 class AddressSerializer(serializers.ModelSerializer):
11     id = serializers.IntegerField(read_only=True)
12     class Meta:
13         model = Address
14         fields = '__all__'
15
16 class DocumentSerializer(serializers.ModelSerializer):
17     id = serializers.IntegerField(read_only=True)
18     class Meta:
19         model = Document
20         fields = '__all__'
21
22 class BankAccountSerializer(serializers.ModelSerializer):
23     id = serializers.IntegerField(read_only=True)
24     class Meta:
25         model = BankAccount
26         fields = '__all__'
27
28 class EmploymentSerializer(serializers.ModelSerializer):
29     id = serializers.IntegerField(read_only=True)
30     class Meta:
31         model = Employment
32         fields = '__all__'
33
34 class ApplicationSerializer(serializers.ModelSerializer):
35     id = serializers.IntegerField(read_only=True)
36     class Meta:
37         model = Application
38         fields = '__all__'
39
40 class RequestSerializer(serializers.ModelSerializer):
41     id = serializers.IntegerField(read_only=True)
42     class Meta:
43         model = Request
44         fields = '__all__'
45
46 class TerroristSerializer(serializers.ModelSerializer):
47     id = serializers.IntegerField(read_only=True)
48     class Meta:
49         model = Terrorist
50         fields = '__all__'
51
52 class BlackListSerializer(serializers.ModelSerializer):
53     id = serializers.IntegerField(read_only=True)
54     class Meta:
55         model = BlackList
56         fields = '__all__'
57
58 class GbdfLSerializer(serializers.ModelSerializer):
59     id = serializers.IntegerField(read_only=True)
```

master* Python 3.7.4 64-bit ('venv': venv) 0 0 0 Ln 27, Col 1 Spaces: 4 UTF-8 LF Python

Қосымша Б жалғасы

serializers.py- кестелерді python тілінің мәлімет типтеріне айналдыру

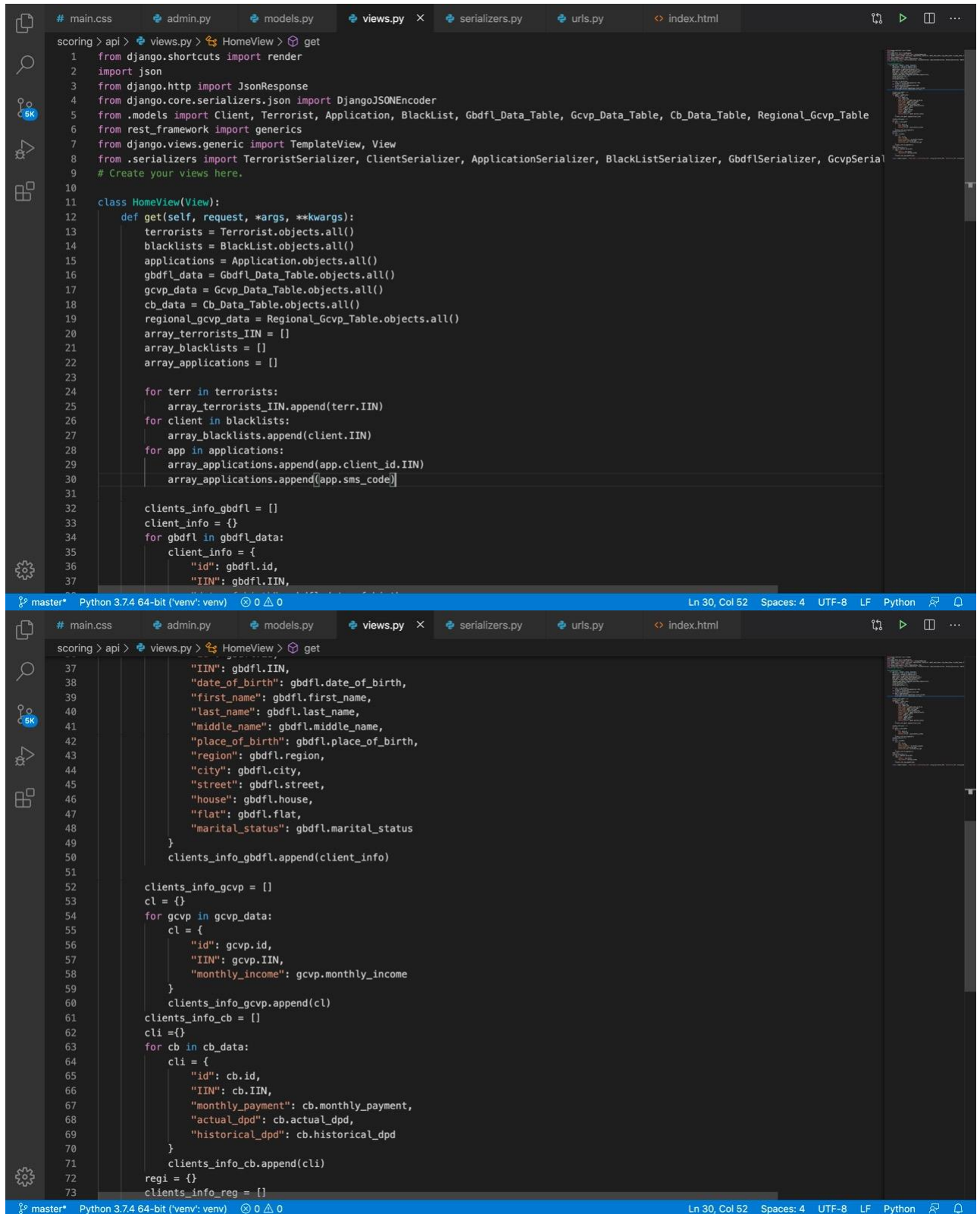


```
# main.css  admin.py  models.py  views.py  serializers.py x  urls.py  index.h  ...
scoring > api > serializers.py > ...
57
58 class GbdfLSerializer(serializers.ModelSerializer):
59     id = serializers.IntegerField(read_only=True)
60     class Meta:
61         model = GbdfL_Data_Table
62         fields = '__all__'
63
64 class CbSerializer(serializers.ModelSerializer):
65     id = serializers.IntegerField(read_only=True)
66     class Meta:
67         model = Cb_Data_Table
68         fields = '__all__'
69
70 class GcvpSerializer(serializers.ModelSerializer):
71     id = serializers.IntegerField(read_only=True)
72     class Meta:
73         model = Gcvp_Data_Table
74         fields = '__all__'
75
76 class Regional_Gcvp_TableSerializer(serializers.ModelSerializer):
77     id = serializers.IntegerField(read_only=True)
78     class Meta:
79         model = Regional_Gcvp_Table
80         fields = '__all__'
```

master* Python 3.7.4 64-bit ('venv': venv) 0 0 Ln 27, Col 1 Spaces: 4 UTF-8 LF

Қосымша В

views.py- сызбаларымыздағы барлық объектілерді алуға арналған функциялар



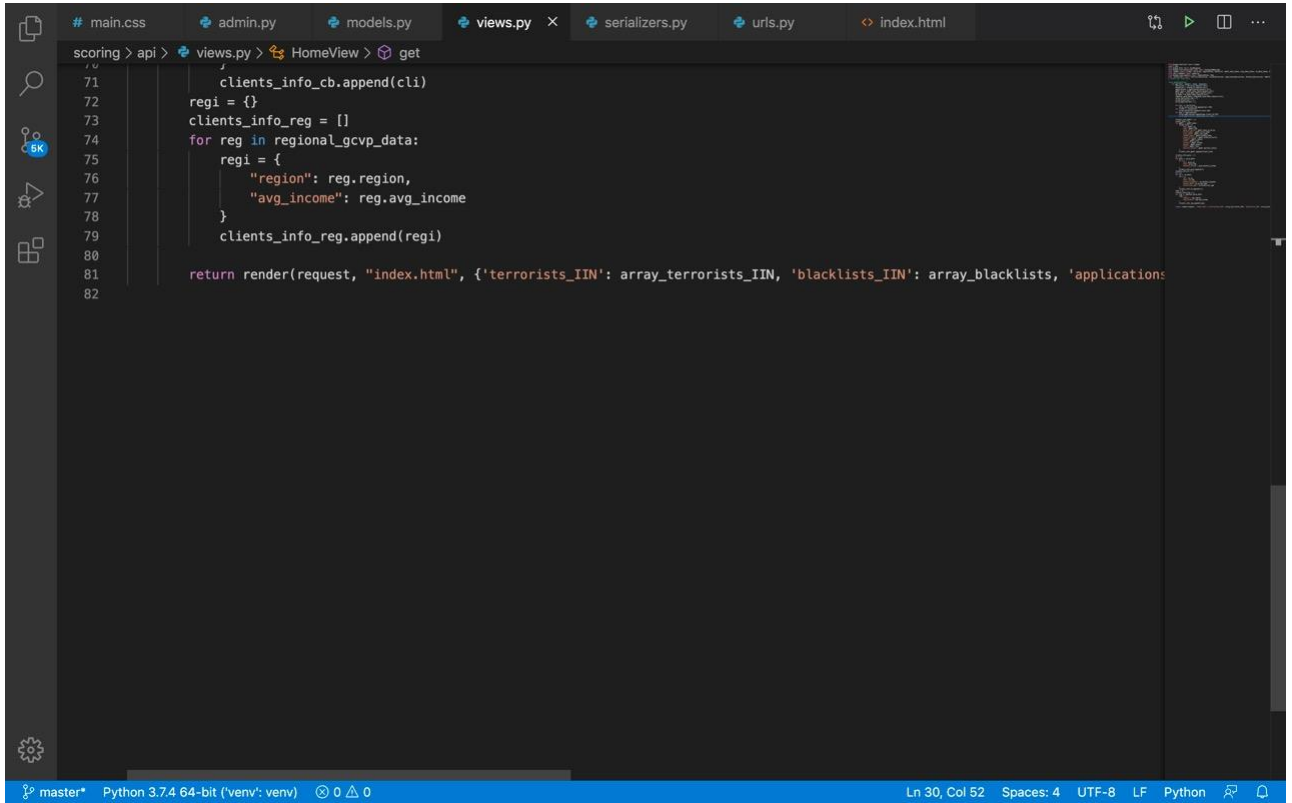
```
# main.css admin.py models.py views.py x serializers.py urls.py index.html
scoring > api > views.py > HomeView > get
1 from django.shortcuts import render
2 import json
3 from django.http import JsonResponse
4 from django.core.serializers.json import DjangoJSONEncoder
5 from .models import Client, Terrorist, Application, BlackList, Gbdfldata_Table, Gcvpdata_Table, Cbdata_Table, Regional_Gcvpdata_Table
6 from rest_framework import generics
7 from django.views.generic import TemplateView, View
8 from .serializers import TerroristSerializer, ClientSerializer, ApplicationSerializer, BlackListSerializer, GbdfldataSerializer, GcvpdataSerializer
9 # Create your views here.
10
11 class HomeView(View):
12     def get(self, request, *args, **kwargs):
13         terrorists = Terrorist.objects.all()
14         blacklists = BlackList.objects.all()
15         applications = Application.objects.all()
16         gbdfldata = Gbdfldata_Table.objects.all()
17         gcvpdata = Gcvpdata_Table.objects.all()
18         cbdata = Cbdata_Table.objects.all()
19         regional_gcvpdata = Regional_Gcvpdata_Table.objects.all()
20         array_terrorists_IIN = []
21         array_blacklists = []
22         array_applications = []
23
24         for terr in terrorists:
25             array_terrorists_IIN.append(terr.IIN)
26         for client in blacklists:
27             array_blacklists.append(client.IIN)
28         for app in applications:
29             array_applications.append(app.client_id.IIN)
30             array_applications.append(app.sms_code)
31
32         clients_info_gbdfldata = []
33         client_info = {}
34         for gbdfldata in gbdfldata:
35             client_info = {
36                 "id": gbdfldata.id,
37                 "IIN": gbdfldata.IIN,
```

```

37         "IIN": gbdfldata.IIN,
38         "date_of_birth": gbdfldata.date_of_birth,
39         "first_name": gbdfldata.first_name,
40         "last_name": gbdfldata.last_name,
41         "middle_name": gbdfldata.middle_name,
42         "place_of_birth": gbdfldata.place_of_birth,
43         "region": gbdfldata.region,
44         "city": gbdfldata.city,
45         "street": gbdfldata.street,
46         "house": gbdfldata.house,
47         "flat": gbdfldata.flat,
48         "marital_status": gbdfldata.marital_status
49     }
50     clients_info_gbdfldata.append(client_info)
51
52     clients_info_gcvpdata = []
53     cl = {}
54     for gcvpdata in gcvpdata:
55         cl = {
56             "id": gcvpdata.id,
57             "IIN": gcvpdata.IIN,
58             "monthly_income": gcvpdata.monthly_income
59         }
60         clients_info_gcvpdata.append(cl)
61     clients_info_cb = []
62     cli = {}
63     for cb in cbdata:
64         cli = {
65             "id": cb.id,
66             "IIN": cb.IIN,
67             "monthly_payment": cb.monthly_payment,
68             "actual_dpd": cb.actual_dpd,
69             "historical_dpd": cb.historical_dpd
70         }
71         clients_info_cb.append(cli)
72     regi = {}
73     clients_info_reg = []
```

Қосымша В жалғасы

views.py- сызбаларымыздағы барлық объектілерді алуға арналған функциялар

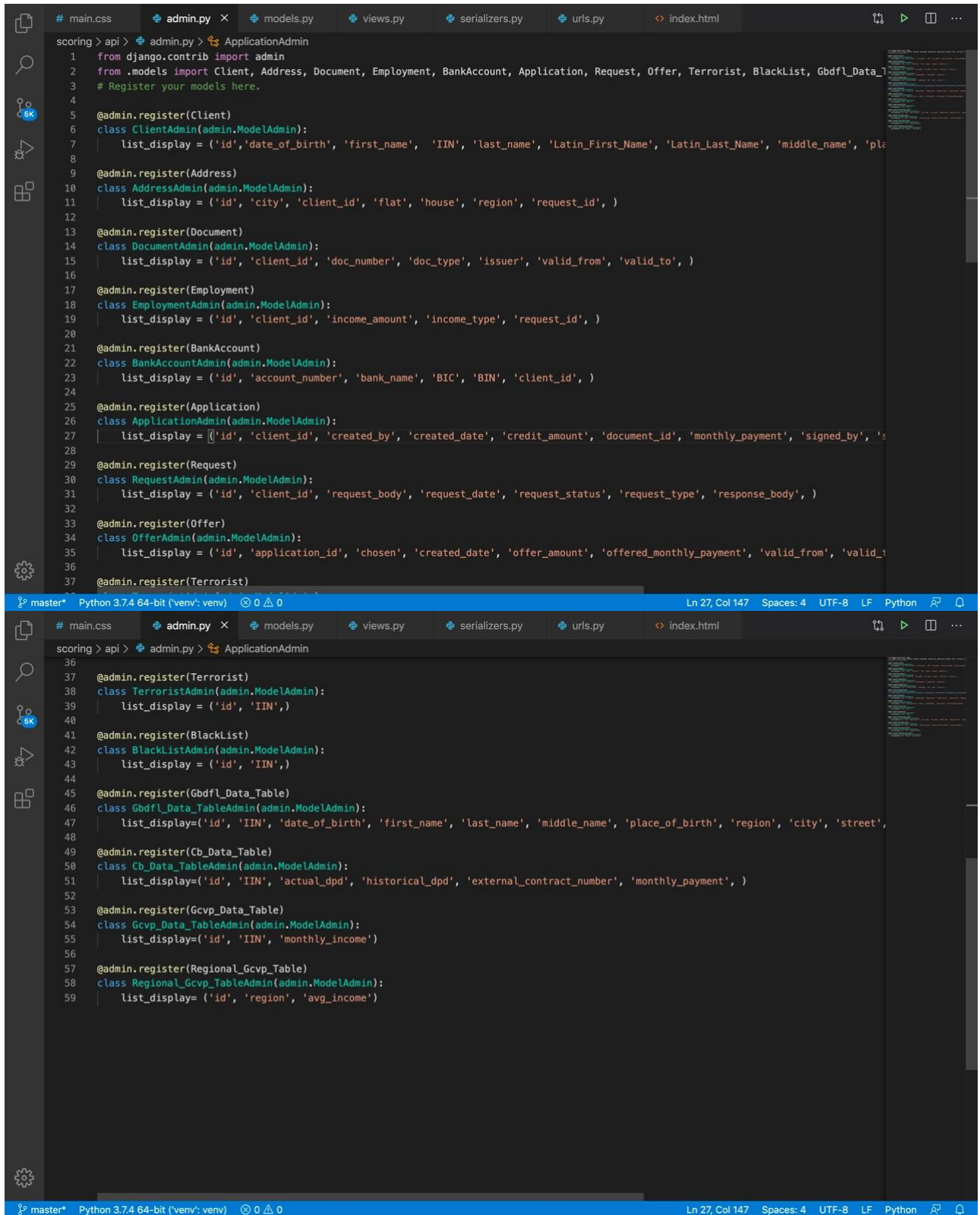


```
# main.css  admin.py  models.py  views.py  serializers.py  urls.py  index.html
scoring > api > views.py > HomeView > get
71         clients_info_cb.append(cli)
72         regi = {}
73         clients_info_reg = []
74         for reg in regional_gcvp_data:
75             regi = {
76                 "region": reg.region,
77                 "avg_income": reg.avg_income
78             }
79             clients_info_reg.append(regi)
80
81         return render(request, "index.html", {'terrorists_IIN': array_terrorists_IIN, 'blacklists_IIN': array_blacklists, 'applications'})
82
```

master* Python 3.7.4 64-bit (venv: venv) 0 0 0 Ln 30, Col 52 Spaces: 4 UTF-8 LF Python

Қосымша Г

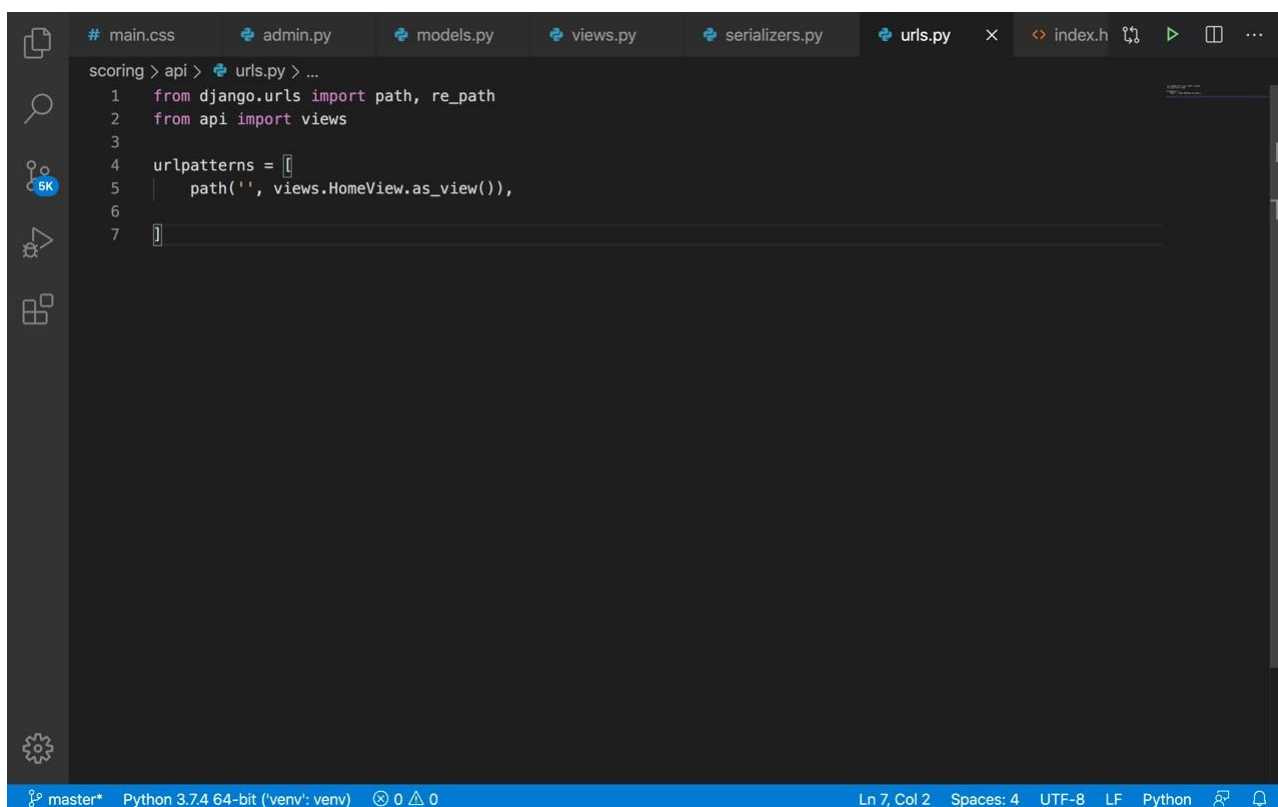
admin.py- барлық кестелерімізді Django администрациясына тіркеу



```
# main.css  admin.py  models.py  views.py  serializers.py  urls.py  index.html
scoring > api > admin.py > ApplicationAdmin
1  from django.contrib import admin
2  from .models import Client, Address, Document, Employment, BankAccount, Application, Request, Offer, Terrorist, BlackList, GbdfL_Data_Table
3  # Register your models here.
4
5  @admin.register(Client)
6  class ClientAdmin(admin.ModelAdmin):
7      list_display = ('id', 'date_of_birth', 'first_name', 'IIN', 'last_name', 'Latin_First_Name', 'Latin_Last_Name', 'middle_name', 'place_of_birth')
8
9  @admin.register(Address)
10 class AddressAdmin(admin.ModelAdmin):
11     list_display = ('id', 'city', 'client_id', 'flat', 'house', 'region', 'request_id', )
12
13 @admin.register(Document)
14 class DocumentAdmin(admin.ModelAdmin):
15     list_display = ('id', 'client_id', 'doc_number', 'doc_type', 'issuer', 'valid_from', 'valid_to', )
16
17 @admin.register(Employment)
18 class EmploymentAdmin(admin.ModelAdmin):
19     list_display = ('id', 'client_id', 'income_amount', 'income_type', 'request_id', )
20
21 @admin.register(BankAccount)
22 class BankAccountAdmin(admin.ModelAdmin):
23     list_display = ('id', 'account_number', 'bank_name', 'BIC', 'BIN', 'client_id', )
24
25 @admin.register(Application)
26 class ApplicationAdmin(admin.ModelAdmin):
27     list_display = ('id', 'client_id', 'created_by', 'created_date', 'credit_amount', 'document_id', 'monthly_payment', 'signed_by', 'status')
28
29 @admin.register(Request)
30 class RequestAdmin(admin.ModelAdmin):
31     list_display = ('id', 'client_id', 'request_body', 'request_date', 'request_status', 'request_type', 'response_body', )
32
33 @admin.register(Offer)
34 class OfferAdmin(admin.ModelAdmin):
35     list_display = ('id', 'application_id', 'chosen', 'created_date', 'offer_amount', 'offered_monthly_payment', 'valid_from', 'valid_to', )
36
37 @admin.register(Terrorist)
38 class TerroristAdmin(admin.ModelAdmin):
39     list_display = ('id', 'IIN',)
40
41 @admin.register(BlackList)
42 class BlackListAdmin(admin.ModelAdmin):
43     list_display = ('id', 'IIN',)
44
45 @admin.register(GbdfL_Data_Table)
46 class GbdfL_Data_TableAdmin(admin.ModelAdmin):
47     list_display = ('id', 'IIN', 'date_of_birth', 'first_name', 'last_name', 'middle_name', 'place_of_birth', 'region', 'city', 'street', 'house', 'flat', )
48
49 @admin.register(Cb_Data_Table)
50 class Cb_Data_TableAdmin(admin.ModelAdmin):
51     list_display = ('id', 'IIN', 'actual_dpd', 'historical_dpd', 'external_contract_number', 'monthly_payment', )
52
53 @admin.register(Gcvp_Data_Table)
54 class Gcvp_Data_TableAdmin(admin.ModelAdmin):
55     list_display = ('id', 'IIN', 'monthly_income', )
56
57 @admin.register(Regional_Gcvp_Table)
58 class Regional_Gcvp_TableAdmin(admin.ModelAdmin):
59     list_display = ('id', 'region', 'avg_income', )
```

Қосымша Д

urls.py- негізгі беттің жолын жазу

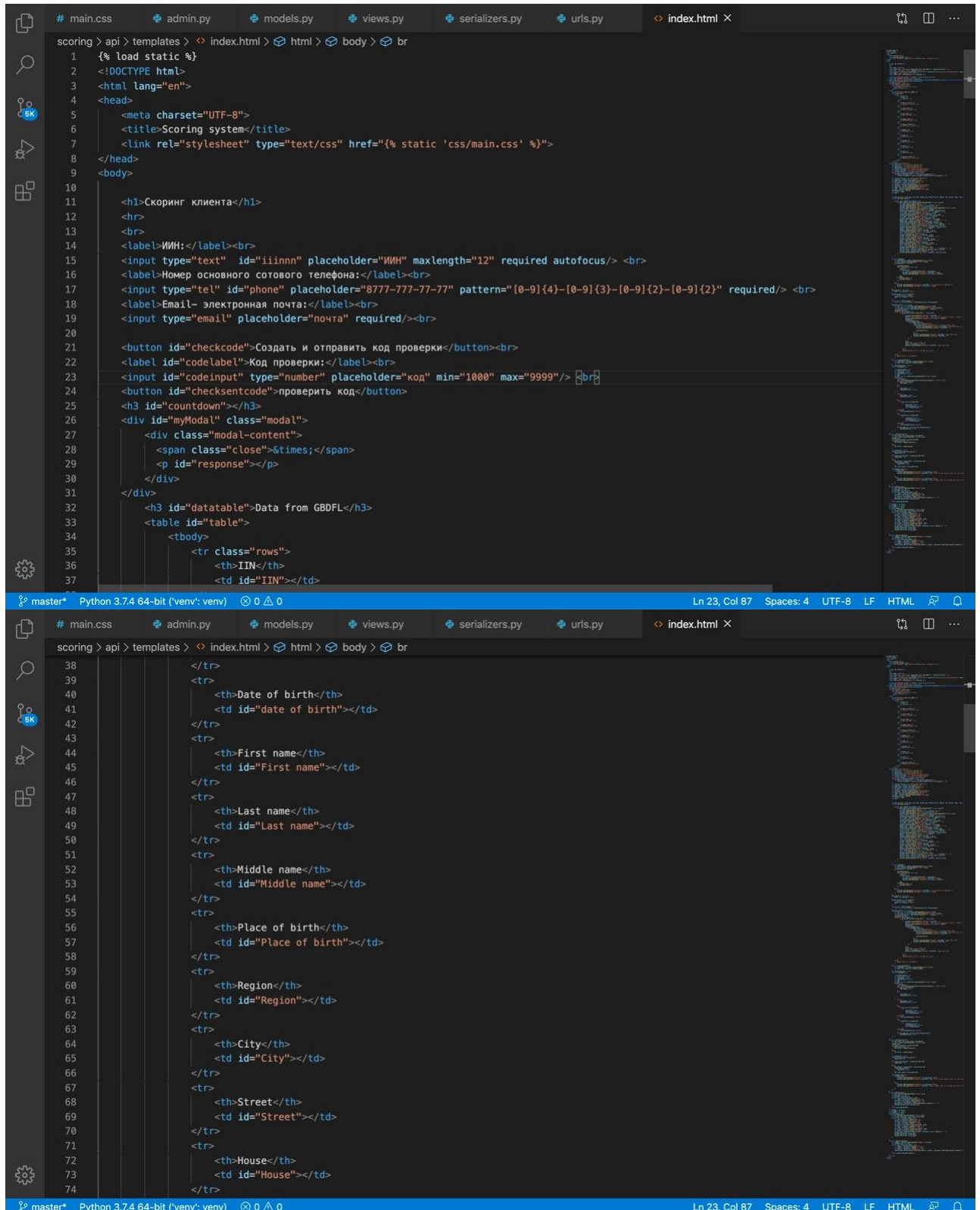


```
scoring > api > urls.py > ...
1  from django.urls import path, re_path
2  from api import views
3
4  urlpatterns = [
5      path('', views.HomeView.as_view()),
6
7  ]
```

Ln 7, Col 2 Spaces: 4 UTF-8 LF Python

Қосымша Ж

index.html- негізгі бетті демонстрациялауға арналған барлық тэгтер мен функциялар



```
# main.css admin.py models.py views.py serializers.py urls.py index.html X
scoring > api > templates > index.html > html > body > br
1 {% load static %}
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5 <meta charset="UTF-8">
6 <title>Scoring system</title>
7 <link rel="stylesheet" type="text/css" href="{% static 'css/main.css' %}">
8 </head>
9 <body>
10
11 <h1>Скоринг клиента</h1>
12 <hr>
13 <br>
14 <label>ИИН:</label><br>
15 <input type="text" id="iinnn" placeholder="ИИН" maxlength="12" required autofocus/> <br>
16 <label>Номер основного сотового телефона:</label><br>
17 <input type="tel" id="phone" placeholder="8777-777-77" pattern="[0-9]{4}-[0-9]{3}-[0-9]{2}-[0-9]{2}" required/> <br>
18 <label>Email- электронная почта:</label><br>
19 <input type="email" placeholder="почта" required/><br>
20
21 <button id="checkcode">Создать и отправить код проверки</button><br>
22 <label id="codeLabel">Код проверки:</label><br>
23 <input id="codeinput" type="number" placeholder="код" min="1000" max="9999"/> <br>
24 <button id="checkscode">проверить код</button>
25 <h3 id="countdown"></h3>
26 <div id="myModal" class="modal">
27 <div class="modal-content">
28 <span class="close">&times;</span>
29 <p id="response"></p>
30 </div>
31 </div>
32 <h3 id="datatable">Data from GBDL</h3>
33 <table id="table">
34 <tbody>
35 <tr class="rows">
36 <th>IIN</th>
37 <td id="IIN"></td>
38 </tr>
39 <tr>
40 <th>Date of birth</th>
41 <td id="date of birth"></td>
42 </tr>
43 <tr>
44 <th>First name</th>
45 <td id="First name"></td>
46 </tr>
47 <tr>
48 <th>Last name</th>
49 <td id="Last name"></td>
50 </tr>
51 <tr>
52 <th>Middle name</th>
53 <td id="Middle name"></td>
54 </tr>
55 <tr>
56 <th>Place of birth</th>
57 <td id="Place of birth"></td>
58 </tr>
59 <tr>
60 <th>Region</th>
61 <td id="Region"></td>
62 </tr>
63 <tr>
64 <th>City</th>
65 <td id="City"></td>
66 </tr>
67 <tr>
68 <th>Street</th>
69 <td id="Street"></td>
70 </tr>
71 <tr>
72 <th>House</th>
73 <td id="House"></td>
74 </tr>
```

Қосымша Ж жалғасы

index.html- негізгі бетті демонстрациялауға арналған барлық тэгтер мен функциялар

```
# main.css  admin.py  models.py  views.py  serializers.py  urls.py  index.html X
scoring > api > templates > index.html > html > body > br
75      <tr>
76          <th>Flat</th>
77          <td id="Flat"></td>
78      </tr>
79      <tr>
80          <th>Marital status</th>
81          <td id="Marital status"></td>
82      </tr>
83  </tbody>
84  </table>
85  <script type="text/javascript">
86      var terrorists = "{{ terrorists_IIN|safe }}";
87      var blacklists = "{{ blacklists_IIN|safe }}";
88      var applications = "{{ applications|safe }}";
89      var clients_info_gbdfl = "{{ clients_info_gbdfl|safe }}";
90      var clients_info_gcvp = "{{ clients_info_gcvp|safe }}";
91      var clients_info_cb = "{{ clients_info_cb|safe }}";
92      for(var i=0; i<clients_info_gbdfl.length; i++){
93          if(clients_info_gbdfl[i]!='' || clients_info_gbdfl[i]!='{')
94              clients_info_gbdfl= clients_info_gbdfl.replace(clients_info_gbdfl[i], "");
95      }
96      var clients_info_reg = "{{ clients_info_reg|safe }}";
97      var reg_data = clients_info_reg.split(",");
98      var gbdfl_data = clients_info_gbdfl.split(",");
99      var gcvp_data = clients_info_gcvp.split(",");
100     var cb_data = clients_info_cb.split(",");
101     var sendcode = document.getElementById("checksentcode");
102     var checkcode = document.getElementById("checkcode");
103     var modal = document.getElementById("myModal");
104     var span = document.getElementsByClassName("close")[0];
105     checkcode.onclick = timerMe;
106     var counter = 1800;
107     var func;
108
109
110     var date_of_birth, First_name, Last_name, Middle_name, Place_of_birth, Region, City, Street, House, Flat, Marital_status;
111     function get_gbdfl_data(){
112
113         for(var i=0; i<gbdfl_data.length; i++){
114             if (gbdfl_data[i].includes(document.getElementById("iinnn").value)){
115                 var datas =gbdfl_data[i].split(", ");
116                 document.getElementById("datatable").style.display= "block";
117                 document.getElementById("table").style.display= "block";
118                 document.getElementById("IIN").innerHTML = document.getElementById("iinnn").value;
119                 date_of_birth = datas[2].substr(30, 42).replace(", ", "");
120                 document.getElementById("date of birth").innerHTML = date_of_birth;
121                 First_name = datas[3].replace("first_name:", "").replace(" ", "");
122                 First_name = First_name.substr(0, First_name.length-1);
123                 document.getElementById("First name").innerHTML = First_name;
124                 Last_name = datas[4].replace("last_name:", "").replace(" ", "");
125                 Last_name = Last_name.substr(0, Last_name.length-1);
126                 document.getElementById("Last name").innerHTML = Last_name;
127                 Middle_name = datas[5].replace("middle_name:", "").replace(" ", "");
128                 Middle_name = Middle_name.substr(0, Middle_name.length-1);
129                 document.getElementById("Middle name").innerHTML = Middle_name;
130                 Place_of_birth = datas[6].replace("place_of_birth:", "").replace(" ", "");
131                 Place_of_birth = Place_of_birth.substr(0, Place_of_birth.length-1);
132                 document.getElementById("Place of birth").innerHTML = Place_of_birth;
133                 Region = datas[7].replace("region:", "").replace(" ", "");
134                 Region = Region.substr(0, Region.length-1);
135                 document.getElementById("Region").innerHTML = Region;
136                 City = datas[8].replace("city:", "").replace(" ", "");
137                 City = City.substr(0, City.length-1);
138                 document.getElementById("City").innerHTML = City;
139                 Street = datas[9].replace("street:", "").replace(" ", "");
140                 Street = Street.substr(0, Street.length-1);
141                 document.getElementById("Street").innerHTML = Street;
142                 House = datas[10].replace("house:", "").replace(" ", "");
143                 House = House.substr(0, House.length-1);
144                 document.getElementById("House").innerHTML = House;
145                 Flat = datas[11].replace("flat:", "").replace(" ", "");
146                 Flat = Flat.substr(0, Flat.length-1);
147                 document.getElementById("Flat").innerHTML = Flat;
148                 Marital_status = datas[12].replace("marital_status:", "").replace(" ", "");
149                 Marital_status = Marital_status.substr(0, Marital_status.length-1);
```

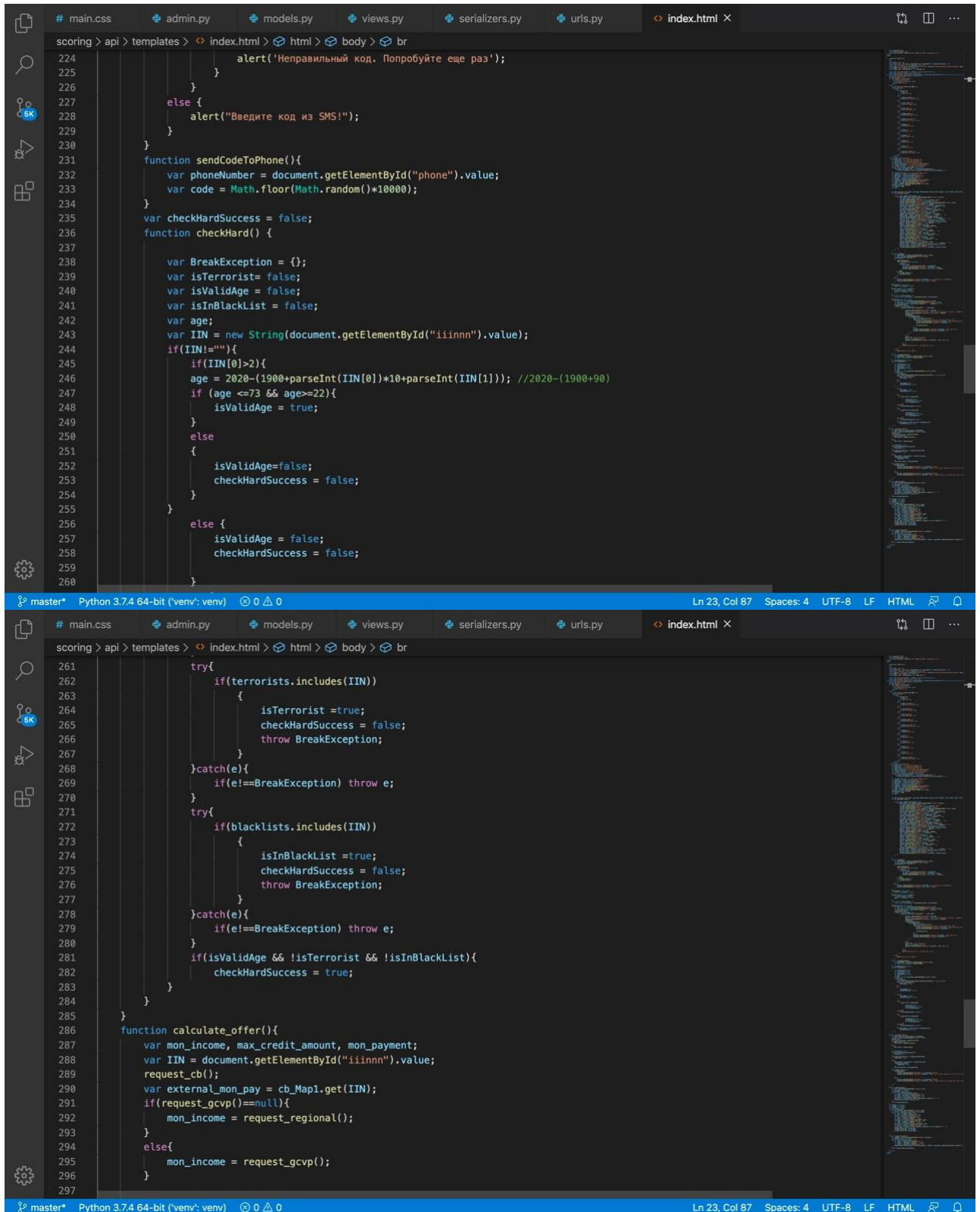
Қосымша Ж жалғасы

index.html- негізгі бетті демонстрациялауға арналған барлық тэгтер мен функциялар

```
# main.css  admin.py  models.py  views.py  serializers.py  urls.py  index.html X
scoring > api > templates > index.html > html > body > br
150     Marital_status = Marital_status.replace("'", "");
151     document.getElementById("Marital_status").innerHTML = Marital_status;
152   }
153 }
154
155
156 function timerMe(){
157   var phoneNumber = document.getElementById("phone").value;
158   var IIN = document.getElementById("iinnn").value;
159   if(phoneNumber!="" && IIN!="")
160   {
161     sendCodeToPhone();
162     func = setInterval(function(){
163       counter--;
164       if(counter>=0){
165         var result = parseInt(counter/60)+''+counter%60;
166         document.getElementById("countdown").innerHTML = result;
167         document.getElementById("countdown").style.color = '#000000';
168       }
169     }, 1000)
170     sleep(3000).then(()=>{
171       get_gbdf_data();
172     })
173   }
174   else {
175     document.getElementById("countdown").innerHTML="Поля обязательны для заполнения";
176     document.getElementById("countdown").style.color = '#d00';
177   }
178 }
179 span.onclick = function(){
180   modal.style.display = "none";
181 }
182 window.onclick = function(event) {
183   if (event.target == modal) {
184     modal.style.display = "none";
185   }
186 }
187
188 const sleep = (milliseconds) => {
189   return new Promise(resolve => setTimeout(resolve, milliseconds))
190 }
191 sendcode.onclick = function(){
192   var IIN = new String(document.getElementById("iinnn").value);
193   var sms_code = new String(document.getElementById("codeinput").value);
194   var array_IIN_smscode = applications.replace('[', '').replace(']', '');
195   clearInterval(func);
196   if(IIN!="" && sms_code!=""){
197     if(array_IIN_smscode.includes(IIN+'', ''+sms_code))
198     {
199       document.getElementById("countdown").innerHTML = '';
200       alert("Контакт успешно подтвержден. Предварительная обработка в процессе. Пожалуйста, подождите");
201       document.getElementById("countdown").style.color = '#000000';
202       checkHard();
203       if(checkHardSuccess){
204         sleep(30000).then(()=>{
205           modal.style.display="block";
206           if(confirm("Ваша заявка пре-одоброено. Хотите продолжить? ")){
207             document.getElementById("response").innerHTML = "Ваша заявка принята. Скоринг начался. Пожа";
208             document.getElementById("countdown").innerHTML = "";
209             calculate_offer();
210           }
211           else {
212             document.getElementById("response").innerHTML = "Сеанс завершен";
213             document.getElementById("countdown").innerHTML = "";
214           }
215         })
216       }
217     }
218     else{
219       alert("Запрос был отклонен.");
220       modal.style.display = "block";
221       document.getElementById("response").innerHTML = "Сеанс завершен";
222     }
223   }
224   else {
225     document.getElementById("countdown").innerHTML="Поля обязательны для заполнения";
226     document.getElementById("countdown").style.color = '#d00';
227   }
228 }
```


Қосымша Ж жалғасы

index.html- негізгі бетті демонстрациялауға арналған барлық тэгтер мен функциялар



```
# main.css  admin.py  models.py  views.py  serializers.py  urls.py  index.html X
scoring > api > templates > index.html > html > body > br
224         alert('Неправильный код. Попробуйте еще раз');
225     }
226 }
227 else {
228     alert("Введите код из SMS!");
229 }
230 }
231 function sendCodeToPhone(){
232     var phoneNumber = document.getElementById("phone").value;
233     var code = Math.floor(Math.random()*10000);
234 }
235 var checkHardSuccess = false;
236 function checkHard() {
237
238     var BreakException = {};
239     var isTerrorist= false;
240     var isValidAge = false;
241     var isInBlackList = false;
242     var age;
243     var IIN = new String(document.getElementById("iinnn").value);
244     if(IIN!=""){
245         if(IIN[0]>2){
246             age = 2020-(1900+parseInt(IIN[0])*10+parseInt(IIN[1])); //2020-(1900+90)
247             if (age <=73 && age>=22){
248                 isValidAge = true;
249             }
250             else
251             {
252                 isValidAge=false;
253                 checkHardSuccess = false;
254             }
255         }
256         else {
257             isValidAge = false;
258             checkHardSuccess = false;
259         }
260     }
261     try{
262         if(terrorists.includes(IIN))
263         {
264             isTerrorist =true;
265             checkHardSuccess = false;
266             throw BreakException;
267         }
268     }catch(e){
269         if(e!==BreakException) throw e;
270     }
271     try{
272         if(blacklists.includes(IIN))
273         {
274             isInBlackList =true;
275             checkHardSuccess = false;
276             throw BreakException;
277         }
278     }catch(e){
279         if(e!==BreakException) throw e;
280     }
281     if(isValidAge && !isTerrorist && !isInBlackList){
282         checkHardSuccess = true;
283     }
284 }
285 }
286 function calculate_offer(){
287     var mon_income, max_credit_amount, mon_payment;
288     var IIN = document.getElementById("iinnn").value;
289     request_cb();
290     var external_mon_pay = cb_Map1.get(IIN);
291     if(request_gcvp()==null){
292         mon_income = request_regional();
293     }
294     else{
295         mon_income = request_gcvp();
296     }
297 }
```

Қосымша Ж жалғасы

index.html- негізгі бетті демонстрациялауға арналған барлық тэгтер мен функциялар

```
# main.css  admin.py  models.py  views.py  serializers.py  urls.py  index.html X
scoring > api > templates > index.html > html > body > br
297
298   var isRejected = false;
299   if(mon_income/external_mon_pay<=2){
300       isRejected=true;
301   }
302   if(cb_Map2.get(IIN)>=60 || cb_Map3.get(IIN)>=60){
303       isRejected = true;
304   }
305   else {
306       mon_payment = mon_income/2 - external_mon_pay;
307       if(mon_payment<1000){
308           isRejected = true;
309       }
310       max_credit_amount = mon_payment*48;
311   }
312   sleep(60000).then(()=>{
313       if(isRejected){
314           document.getElementById("datatable").style.display= "block";
315           document.getElementById("response").innerHTML = "Запрос был отклонен. Банк не может дать вам кредит.";
316       }
317       else{
318           document.getElementById("datatable").style.display= "block";
319           document.getElementById("response").innerHTML = "Запрос успешно завершен. Максимальная сумма кредита "+ max_credit_amou
320       }
321   })
322   })
323
324
325   function request_gcvp(){
326       var IIN = document.getElementById("iinnn").value;
327       let gcvp_Map = new Map();
328       var gcvp_IIN, gcvp_mon_inc;
329       for(var i=0; i<gcvp_data.length; i++){
330           var index1 = gcvp_data[i].indexOf("IIN:");
331           gcvp_IIN = gcvp_data[i].substr(index1+7, 12);
332           var index2 = gcvp_data[i].indexOf("ome: ");
333           gcvp_mon_inc = gcvp_data[i].substring(index2+6).replace("}", "");
334           gcvp_Map.set(gcvp_IIN, gcvp_mon_inc);
335       }
336       return gcvp_Map.get(IIN);
337   }
338   var cb_Map1 = new Map();
339   var cb_Map2 = new Map();
340   var cb_Map3 = new Map();
341   function request_cb(){
342       var IIN = document.getElementById("iinnn").value;
343       var cb_IIN, cb_mon_pay, cb_act_dpd, cb_hist_dpd;
344       for(var i=0; i<cb_data.length; i++){
345           var ind1 = cb_data[i].indexOf("IIN:");
346           cb_IIN = cb_data[i].substr(ind1+7, 12);
347           var ind2 = cb_data[i].indexOf("ment");
348           var ind3 = cb_data[i].indexOf(", 'act");
349           cb_mon_pay = cb_data[i].substring(ind2+6, ind3);
350           var ind4 = cb_data[i].indexOf("ual_dpd: ");
351           var ind5 = cb_data[i].indexOf(", 'hist");
352           cb_act_dpd = cb_data[i].substring(ind4+10, ind5);
353           var ind6 = cb_data[i].indexOf("ical_dpd: ");
354           cb_hist_dpd = cb_data[i].substr(ind6+10, cb_data[i].length).replace("}", "");
355           cb_Map1.set(cb_IIN, cb_mon_pay);
356           cb_Map2.set(cb_IIN, cb_act_dpd);
357           cb_Map3.set(cb_IIN, cb_hist_dpd);
358       }
359   }
360
361   function request_regional(){
362       var reegion = document.getElementById("Region").innerText;
363       var regional_Map = new Map();
364       for(var i=0; i<reg_data.length; i++){
365           var index1 = reg_data[i].indexOf("ion:");
366           var index2 = reg_data[i].indexOf(", 'avg");
367           var index3 = reg_data[i].indexOf("ome: ");
368           regional_Map.set(reg_data[i].substring(index1+7, index2), reg_data[i].substring(index3+6).replace("}", ""));
369       }
370   }
```

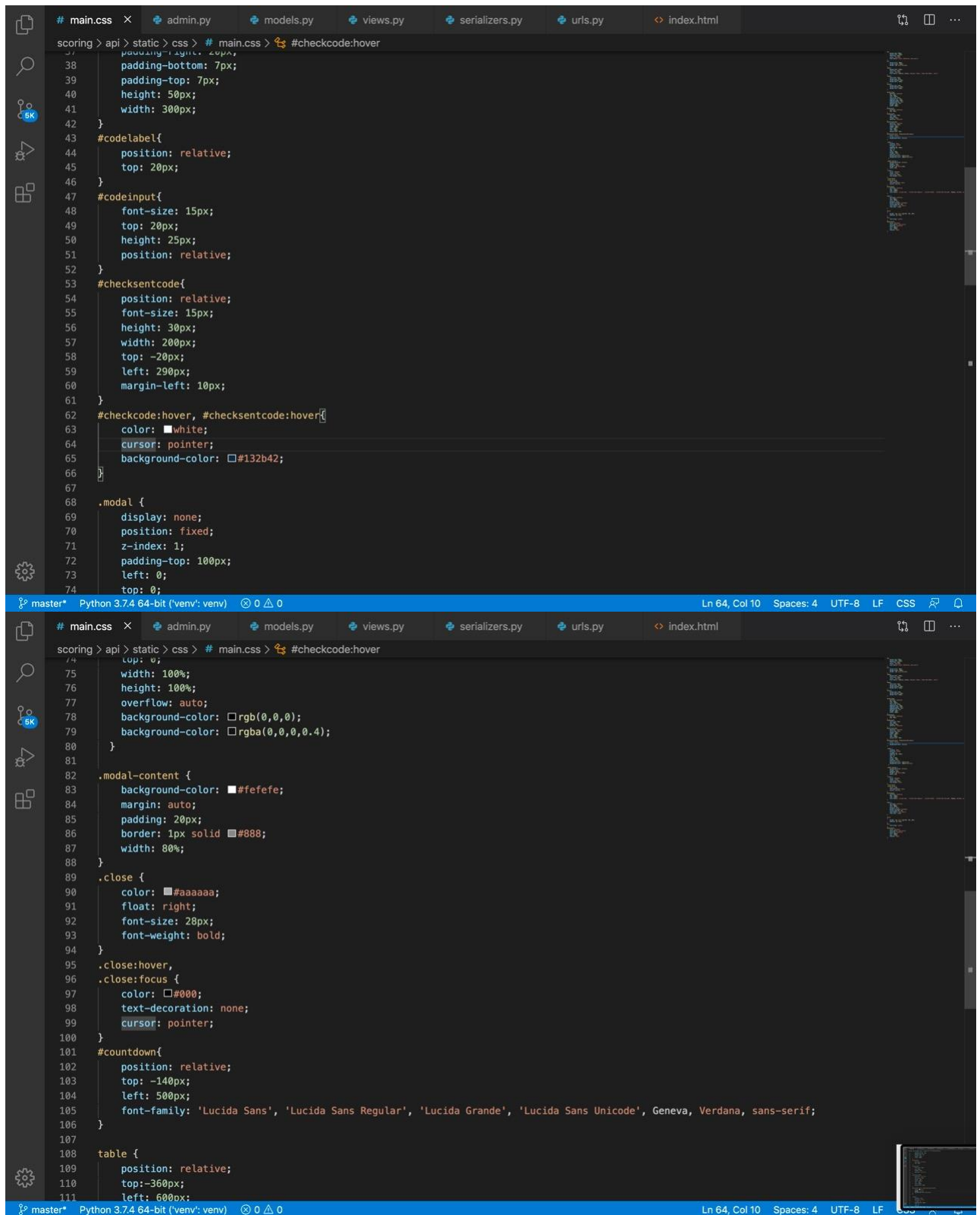
Қосымша Ж жалғасы

index.html- негізгі бетті демонстрациялауға арналған барлық тэгтер мен функциялар

```
scoring > api > templates > index.html > html > body > br
361     function request_regional(){
362         var reeegion = document.getElementById("Region").innerText;
363         var regional_Map = new Map();
364         for(var i=0; i<reg_data.length; i++){
365             var index1 = reg_data[i].indexOf("ion:");
366             var index2 = reg_data[i].indexOf("'", 'avg');
367             var index3 = reg_data[i].indexOf("ome: ");
368             regional_Map.set(reg_data[i].substring(index1+7, index2), reg_data[i].substring(index3+6).replace("}", ""));
369         }
370         return regional_Map.get(reeegion);
371     }
372     </script>
373 </body>
374 </html>
```

Қосымша Л

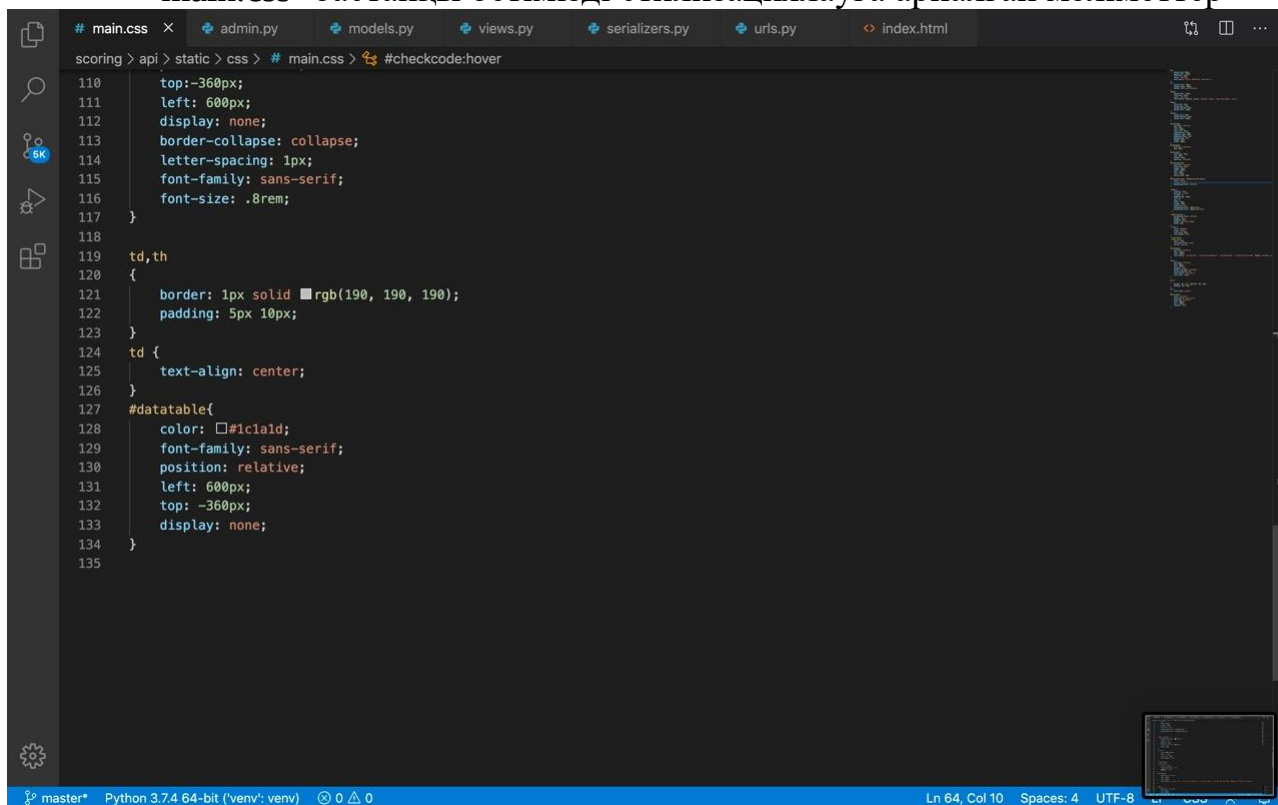
main.css- бастапқы бетімізді стилизациялауға арналған мәліметтер



```
# main.css x admin.py models.py views.py serializers.py urls.py index.html
scoring > api > static > css > # main.css > #checkcode: hover
38 padding-bottom: 7px;
39 padding-top: 7px;
40 height: 50px;
41 width: 300px;
42 }
43 #codelabel{
44 position: relative;
45 top: 20px;
46 }
47 #codeinput{
48 font-size: 15px;
49 top: 20px;
50 height: 25px;
51 position: relative;
52 }
53 #checksente{
54 position: relative;
55 font-size: 15px;
56 height: 30px;
57 width: 200px;
58 top: -20px;
59 left: 290px;
60 margin-left: 10px;
61 }
62 #checkcode: hover, #checksente: hover{
63 color: white;
64 cursor: pointer;
65 background-color: #132b42;
66 }
67
68 .modal {
69 display: none;
70 position: fixed;
71 z-index: 1;
72 padding-top: 100px;
73 left: 0;
74 top: 0;
75
76 width: 100%;
77 height: 100%;
78 overflow: auto;
79 background-color: rgb(0,0,0);
80 background-color: rgba(0,0,0,0.4);
81 }
82 .modal-content {
83 background-color: #fefefe;
84 margin: auto;
85 padding: 20px;
86 border: 1px solid #888;
87 width: 80%;
88 }
89 .close {
90 color: #aaaaaa;
91 float: right;
92 font-size: 28px;
93 font-weight: bold;
94 }
95 .close: hover,
96 .close: focus {
97 color: #000;
98 text-decoration: none;
99 cursor: pointer;
100 }
101 #countdown{
102 position: relative;
103 top: -140px;
104 left: 500px;
105 font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
106 }
107
108 table {
109 position: relative;
110 top: -360px;
111 left: 600px;
```

Қосымша Л жалғасы

main.css- бастапқы бетімізді стилизациялауға арналған мәліметтер



```
# main.css x admin.py models.py views.py serializers.py urls.py index.html
scoring > api > static > css > # main.css > #checkcode: hover
110     top: -360px;
111     left: 600px;
112     display: none;
113     border-collapse: collapse;
114     letter-spacing: 1px;
115     font-family: sans-serif;
116     font-size: .8rem;
117 }
118
119 td,th
120 {
121     border: 1px solid #rgb(190, 190, 190);
122     padding: 5px 10px;
123 }
124 td {
125     text-align: center;
126 }
127 #datatable{
128     color: #1c1c1d;
129     font-family: sans-serif;
130     position: relative;
131     left: 600px;
132     top: -360px;
133     display: none;
134 }
135
```

Ln 64, Col 10 Spaces: 4 UTF-8