

**Національний університет кораблебудування
імені адмірала Макарова**

Навчально-науковий інститут комп'ютерних та інженерно-технологічних наук

(повне найменування інституту, назва факультету)

Кафедра комп'ютерних наук та інформаційних технологій
(повна назва кафедри)

Пояснювальна записка

до дипломної роботи

бакалавр

(освітньо-кваліфікаційний рівень)

на тему «Розробка проєкту інформаційної системи управління кредитного відділу
банку»

Виконав: студент 4 курсу, групи 4142
напряму підготовки (спеціальності)

121 Комп'ютерні науки та інформаційні
технології

(шифр і назва напрямку підготовки, спеціальності)

Бабенко Я.А.

(прізвище та ініціали)

Керівник Маршак О. І.

(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

м. Миколаїв – 2020 року

АНОТАЦІЯ

Темою роботи є розробка проєкту інформаційної системи управління кредитного відділу банку. Система спрямована на підвищення ефективності роботи працівників у кредитному відділі.

В роботі виконано аналіз діяльності об'єкта та його організаційної структури, визначенні недоліки системи управління кредитного відділу, сформовані вимоги до проєкту інформаційної системи управління кредитного відділу та проведено аналіз існуючих рішень. На основі вимог до інформаційної системи розроблено концепцію та програмні рішення. Розробка програмних рішень складається з загальносистемних рішень, рішень з інформаційного, програмного та технічного забезпечення.

Робота викладена на 60 сторінках друкованого тексту, містить 20 рисунків та список використаної літератури з 19 найменувань.

					ДР.122.4142.02.ПЗ	Арк.
						3
Зм.	Арк	№ докум.	Підпис	Дата		

ABSTRACT

The theme of the work is the development of the project of the information management system of the credit department of the bank. The system is aimed at improving the efficiency of employees in the credit department.

The paper analyzes the activities of the object and its organizational structure, identifies the shortcomings of the management system of the credit department, formed the requirements for the project of the information management system of the credit department and analyzed the existing solutions. Based on the requirements for the information system, the concept and software solutions have been developed. Software development consists of system-wide solutions, solutions for information, software and hardware.

The work is spread over 60 pages of printed text, contains 20 pictures, a list of used literature from 19 items.

					ДР.122.4142.02.ПЗ	Арк.
						4
Зм.	Арк	№ докум.	Підпис	Дата		

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ I. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА РОЗРОБКА КОНЦЕПЦІЇ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМ	8
1.1 Опис об'єкта дослідження та його організаційна структура	8
1.2 Аналіз предметної області процесів обробки інформації	12
1.3 Формування вимог до проєкту інформаційної системи	13
1.4 Цільове призначення та задача дослідження	14
1.5 Розробка функціональної моделі(блок-схеми) системи	15
1.6 Формування вхідної, довідкової та вихідної інформації бази даних системи	17
РОЗДІЛ II. РОЗРОБКА ТЕХНІЧНОГО ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ КРЕДИТНОГО ВІДДІЛУ БАНКУ	18
2.1 Загальносистемні рішення	19
2.1.1 Діаграма використання	19
2.1.2 Діаграма діяльності	24
2.2 Рішення з інформаційного забезпечення	27
2.2.1. Концептуальна та логічна модель бази даних	27
2.2.2 Аналіз нормалізації сутностей бази даних	29
2.3 Рішення з програмного забезпечення.....	30
2.3.1 Структура програмного забезпечення	30
2.3.2 Фізична модель бази даних	32
2.3.3 Організаційна структура	32
2.3.4 Методи і засоби розробки програмного забезпечення	33
2.4 Рішення з технічного забезпечення.....	35

					ДР.122.4142.02.ПЗ.Р1		
Зм.	Арк.	№ докум.	Підпис	Дата	Аналіз предметної області та розробка концепції проєкту інформаційної системи		
Студент		Бабенко Я.А.					
Консультант							
Керівник		Маршак О.І.					
Н.контр.							
Зав.кафедрою		Михелєв І.І.			Літ. Арк. Аркушів		
					5 13		
					НУК імені адмірала Макарова		

2.5 Рішення Розробка інтерфейсів.....	36
РОЗДІЛ III. РЕАЛІЗАЦІЯ ПРОЄКТУ ІНФОРМАЦІОННОЇ СИСТЕМИ	
КРЕДИТНОГО ВІДДІЛУ БАНКУ	38
3.1 Розробка документації системи	39
3.2 Розгортання системи	39
РОЗДІЛ IV. ОХОРОНА ПРАЦІ	40
4.1 Характеристика приміщення	41
4.2 Мікрокліматичні умови.....	42
4.3 Освітлення	42
4.4 Шум та вібрація	42
4.5 Випромінювання	43
ВИСНОВОК	44
Список використаних джерел.....	45
ДОДАТОК А	48
ДОДАТОК Б.....	51
ДОДАТОК В.....	52
ДОДАТОК Г	53

ВСТУП

На сьогоднішній день існує чимало банків. Деякі з них процвітають, а деякі, навпаки, приносять лише збитки. Постає питання: як же вирішити цю проблему?

З часом усе більше актуальність набирають інформаційні системи, які вже помітно покращили та спростували роботу різноманітних підприємств. Одною з головних складових сучасного світу є інформація. Процес переробки інформації є дуже довгим та ресурсномістким процесом. Цей процес потребує його прискорення, для економії часу та ресурсів.

Кожен день у банках обслуговується безліч клієнтів та не усі можуть справитися з таким потоком інформації.

Одним із варіантів вирішення цієї проблеми є впровадження нових технологій для обробки інформації. Впровадження нових технологій може бути вирішене за допомогою нової інформаційної системи. Тому в даній роботі буде розроблено проект інформаційної системи для кредитного відділу банку.

Основною задачею проекту є розробка інформаційної системи, яка спрямована на покращення робочого місця менеджера банку.

Для вирішення цього питання потрібно вирішити такі задачі:

- здійснити аналіз предметної області;
- розробити базу даних;
- розробити web-додаток, що взаємодіє з базою даних;

					ДР.122.4142.02.ПЗ.Р1	Арк.
						7
Зм.	Арк	№ докум.	Підпис	Дата		

РОЗДІЛ I. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА РОЗРОБКА КОНЦЕПЦІЇ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ

1.1 Опис об'єкта дослідження та його організаційна структура

Розглянемо структуру банку «SimpleBank», яка є об'єктом дослідження даної роботи.

Банк - фінансово-кредитна установа, яка проводить різноманітні види операцій з грошима і цінними паперами і що надає фінансові послуги уряду, юридичним і фізичним особам. Банк - це комерційне юридична особа, яка створена з метою отримання прибутку, має право здійснювати банківські операції, має виключне право на залучення грошових коштів юридичних і фізичних осіб з метою їх подальшого розміщення від свого імені, а також на відкриття та ведення банківських рахунків юридичних і фізичних осіб. [8]

У функції банків входить безпечне зберігання грошей клієнтів. Так як у банку є багато клієнтів, які зберігають в ньому свої гроші, то банк стає здатний переводити гроші від одного з них іншому шляхом зміни записів в банківських рахунках (безготівкові розрахунки). Безготівкові розрахунки можливі і між клієнтами різних банків завдяки системі кореспондентських рахунків. Банки видають кредити. При цьому фактично створюється додаткова грошова маса. Кредит або кредитні відносини - суспільні відносини, що виникають між суб'єктами економічних відносин з приводу руху вартості. Кредитні відносини можуть виражатися в різних формах кредиту (комерційний кредит, банківський кредит), позику та інші. [8]

«SimpleBank» використовує лінійну організаційну структуру, яка являє собою систему, в якій кожний підлеглий підпорядкований лише одному керівнику. У кожному підрозділі виконується повний комплекс робіт. Схема організаційної структури представлена на Рисунку 1.1.

					ДР.122.4142.02.ПЗ.Р1	Арк.
						8
Зм.	Арк	№ докум.	Підпис	Дата		

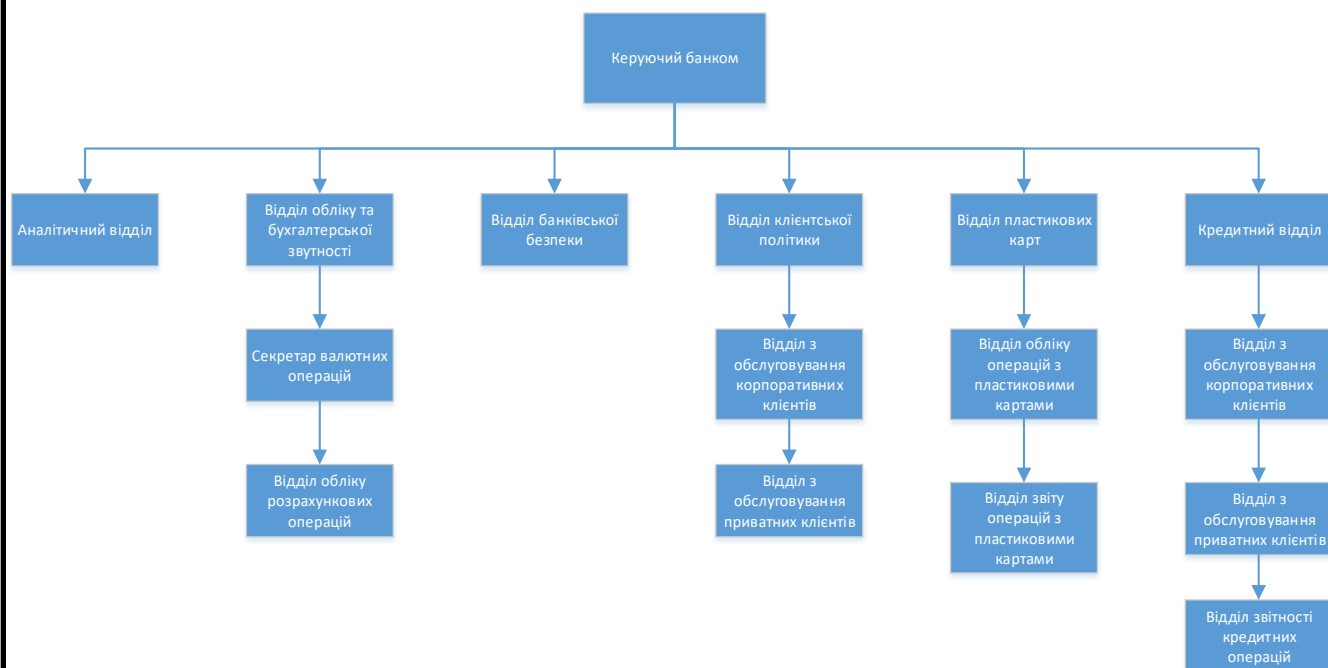


Рисунок 1.1 – Схема організаційної структури «SimpleBank».

Кредитне управління є найважливішим і найбільшим підрозділом банку, оскільки основною функцією і основним джерелом його прибутку є надання різного роду позик. [9]

Виникнення кредиту як особливої форми вартісних відносин відбувається тоді, коли вартість, вивільнена в одного економічного суб'єкта, якийсь час не вступає в новий відтворювальний цикл. Завдяки кредиту вона переходить від суб'єкта, що не використовує її (кредитор), до іншого суб'єкта, що відчуває потребу в додаткових коштах (позичальник). [10]

Кредит відіграє важливу роль в саморегулювання величини коштів, необхідних для здійснення господарської діяльності. Завдяки кредиту підприємства мають у своєму розпорядженні в будь-який момент таку суму грошових коштів, яка необхідна для нормальної роботи. Так само роль кредиту важлива для поповнення оборотних коштів, потреба в яких у кожного підприємства не стабільна, змінюється в залежності від умов роботи: ринкових, природних, кліматичних, політичних та ін. Роль кредиту велика для відтворення

основних фондів. Використовуючи кредит, підприємство може удосконалювати, збільшувати виробництво значно швидше, ніж при його відсутності. [11]

Важлива роль кредиту в регулюванні ліквідності банківської системи, а також у створенні ефективного механізму фінансування державних витрат. [11]

Споживчий кредит - кредит, що надається безпосередньо громадянам (домашнім господарствам) для придбання предметів споживання. Такий кредит беруть не тільки для покупки товарів тривалого користування (квартири, меблі, автомобілі). Але і для інших покупок (мобільні телефони, побутова техніка, продукти харчування).[12]

Відділ кредитування ділитися на 3 відділи:

- Робота з приватними особами (Фізичні особи)
- Робота з корпоративними клієнтами (підприємства)
- Відділ звіту

У кожному з перших двох відділі певна кількість співробітників, у яких є свої клієнти.

Фізичні особи: звичайна людина - громадянин України. Для видачі кредиту таким особам необхідна особиста інформація клієнта (ПІБ, адреса, паспортні дані, дата народження, члени сім'ї, довідка про доходи).

Співробітник банку визначає категорію клієнта, по якій вирішується видавати кредит чи ні.

Так само існує категорія сумнівних клієнтів (у яких сумнівні доходи), перш ніж видати такому клієнтові кредит його дані ретельно вивчаються, в першу чергу довідку про доходи. До таких клієнтів більше уваги і суворіше правила (термін виплат).

Корпоративні клієнти (юридичні особи): це зареєстрована в установленому законом порядку організація, фірма, компанія, яка має у власності, господарському віданні або оперативному управлінні відокремлене майно. Для видачі кредиту таким особам необхідна інформація клієнта (Назва підприємства, адреса, начальник, баланс, звіт про доходи).

					ДР.122.4142.02.ПЗ.Р1	Арк.
						10
Зм.	Арк	№ докум.	Підпис	Дата		

Договір укладається між 2 сторін: позичальник (клієнт) і кредитор (банк).
Договір складається тільки в письмовій формі і в нього входить:

- мета кредиту.
- сума кредиту;
- термін кредитування. За строками користування кредити поділяють на короткострокові (до 1 року), середньострокові (до 3 років) і довгострокові (більше 3 років);
- умови і порядок видачі та погашення кредиту;
- види забезпечення зобов'язань позичальника (наприклад: застава, поручительство)
- процентні ставки;
- порядок внесення плати за кредит;
- обов'язки, права і відповідальність сторін щодо видачі та погашення кредиту.

Працівники відділу звіту контролюють виплати, складають графіки виплат та формують місячний звіт для керівництва.

Також у кредитному відділі є менеджер, який слідкує за розподілом клієнтів, формуванням звітів та виплат, роботою усіх відділів кредитного відділу банку.

					ДР.122.4142.02.ПЗ.Р1	Арк.
						11
Зм.	Арк	№ докум.	Підпис	Дата		

1.2 Аналіз предметної області процесів обробки інформації

Інформаційна система - це взаємозв'язана сукупність засобів, методів і персоналу, використовуваних для зберігання, обробки та видачі інформації для досягнення мети управління. В сучасних умовах основним технічним засобом обробки інформації є персональний комп'ютер. Більшість сучасних інформаційних систем перетворюють не інформацію, а дані. Тому часто їх називають системами обробки даних. [5]

Проаналізувавши інформаційну систему, яку використовує SimpleBank, було виявлено, що ця система не має повного функціоналу та є недосконалою. Вона має велику кількість недоліків, не пов'язує клієнтів зі співпрацівниками та кредитами. Ця система відіграє дуже важливу роль, адже завдяки їй можливо відстежити усі дії робітників з клієнтами, та дії клієнтів.

В інформаційній системі кредитного відділу банку SimpleBank було виявлено наступні недоліки:

1. Всім працівникам відділу доводиться постійно подавати окремі звіти по клієнтам та кредитам.
2. Менеджеру відділу постійно необхідно переглядати та формувати окремі графіки виплат кожного клієнта письмово.
3. Працівники відділу повинні самостійно контролювати кількість навантаження (кількість клієнтів).
4. Менеджеру необхідно в ручному режимі закріплювати клієнтів та кредити за працівниками.

Через усі виявлені недоліки працездатність відділу є не дуже високою. Для підвищення працездатності відділу необхідно замінити інформаційну систему на нову, більш багатофункціональну, яка допоможе зробити роботу швидше та зручніше для кожного працівника відділу. Система допоможе швидко відстежити усі операції, структурує роботу з середини підприємства. Це зможе позитивно відобразитись на подальшій роботі усього банку та збільшити клієнтську базу.

					ДР.122.4142.02.ПЗ.Р1	Арк.
						12
Зм.	Арк	№ докум.	Підпис	Дата		

1.3 Формування вимог до проєкту інформаційної системи

Дивлячись на недоліки системи, яку використовує об'єкт дослідження, можна сформулювати вимоги, яким повинна відповідати нова система для того, щоб підвищити працездатність та збільшити клієнтську базу.

Система по-перше повинна бути зрозумілою та зручною у використанні для кожного співробітника. Кращим вирішенням цього питання є веб-сайт, доступ до якого можна отримати з будь-якого місця де є інтернет та комп'ютер. Робота стане простою навіть для нового співробітника, завдяки простому інтерфейсу та не потрібно витрачати багато часу для пристосування до нової системи.

Перша проблема полягає у тому, що працівникам відділу доводиться кожен місяць робити та подавати звіти по новим клієнтам та по усім кредитам й операціям окремо. Це не зручно та займає багато додаткового часу. Тому наступною вимогою є формування щомісячних звітів за певними критеріями. Наприкінці місяця у розділі звіти працівник обирає категорію звіту:

1. Звіт про обслугованих клієнтів за місяць по відділам. Цей звіт потрібен для оцінки роботи відділу та працівників.
2. Звіт про виплати по кредиту. Фіксуються усі виплати які були проведені до банку, як виплати по кредиту.
3. Звіт по списку виплачених кредитів. Звіт передає тенденцію покращення чи погіршення клієнтської бази.
4. Звіт підрахунок прибутку відділу з виплат та виданих кредитах

Звіт зберігається на комп'ютері робітника. Після перегляду, співробітник банку відсилає звіт менеджеру або керівнику установи. Також кожен співробітник формує окремі звіти по виплатам кожного клієнта враховуючи графік виплат.

Наступною проблемою є те, що менеджеру необхідно формувати графіки виплат в ручному режимі враховуючи дату видачі кредиту, суму кредиту та строк. Ще однією вимогою є підрахунок щомісячних виплат та формування графіку виплат по кожному кредиту згідно з договором. Це дозволить швидко надати клієнту графік виплат згідно з яким будуть відбуватися виплати.

					ДР.122.4142.02.ПЗ.Р1	Арк.
						13
Зм.	Арк	№ докум.	Підпис	Дата		

Важливою проблемою є те, що працівники зобов'язанні самостійно контролювати навантаження (кількість клієнтів). Це дуже не зручно та займає робочий час у працівників відділу. Для вирішення цієї проблеми, менеджер повинен буде розподіляти нових клієнтів по відділам і співробітниками за допомогою нової інформаційної системи.

Не мала важливо щоб реєстрація клієнтів та кредитів була швидкою та зручною, адже це впливає на якість та швидкість виконання роботи.

Пошук стане набагато простіше з новим функціоналом та співробітник з легкістю повинен знаходити потрібну інформацію.

Для формування вимог до проєкту інформаційної системи використовувалась [5].

1.4 Цільове призначення та задача дослідження

Цільовим призначенням дослідження є підвищення ефективності роботи кредитного відділу банку та збільшення клієнтської бази за рахунок створення проєкту сучасної інформаційної системи для управління кредитного відділу банку.

Головною задачею дослідження є розробка концепції проєкту інформаційної системи. Для того, щоб виконати концепцію необхідно провести роботи, які включають в себе:

- Дослідження організаційної структури банку.
- Дослідження системи управління кредитного відділу, яка використовується установою.
- Виявлення проблем, які присутні в системі управління кредитного відділу.
- Дослідження можливих шляхів усунення знайдених проблем в системі управління кредитного відділу, приймаючи до уваги організаційну структуру установи.

- Формулювання вимог, яким повинна відповідати інформаційна система.
- Дослідження можливості додаткових засобів для підвищення продуктивності співробітників відділу, які потрібно додати в нову інформаційну систему.
- На основі вимог до інформаційної системи розробити функціональну модель системи.
- Вибір підходящих засобів, за допомогою яких буде досягнуто поставлених вимог до інформаційної системи.
- На основі розроблено концепції системи зробити висновок, про те наскільки кращою буде нова система від вже існуючої.

1.5 Розробка функціональної моделі(блок-схеми) системи

Розроблений проєкт системи управління кредитного відділу включає в себе наступні процеси:

1. Реєстрація в системі співробітника. Взяття співробітника на облік установи. З процесом взаємодії менеджер, який надає системі персональні дані співробітника.
2. Реєстрація в системі клієнта. Взяття клієнта на облік. З процесом взаємодії менеджер, який надає системі персональні дані клієнта.
3. Створення в системі кредиту. Внесення усіх даних за договором. З процесом взаємодії співробітник, який надає системі дані договору, який склав з клієнтом.
4. Формування графіків виплат. З процесом взаємодії менеджер, враховуючи усі данні за договором по кредиту.
5. Формування списків виплачених кредитів. З процесом взаємодії співробітник відділу, враховуючи усі виплати по кредиту.
6. Формування усіх виплат які були зроблені клієнтами. З процесом взаємодії співробітник, який надає системі данні виплат та отримує звіт.

					ДР.122.4142.02.ПЗ.Р1	Арк.
						15
Зм.	Арк	№ докум.	Підпис	Дата		

7. Розподіл кредитів та клієнтів по співробітникам. З процесом взаємодії менеджер враховуючи данні клієнтів, кредитів та персональні данні працівників.

Всі описані вище процеси було висвітлено та представлено у виді блок-схеми, яка представлена на Рисунку 1.2.

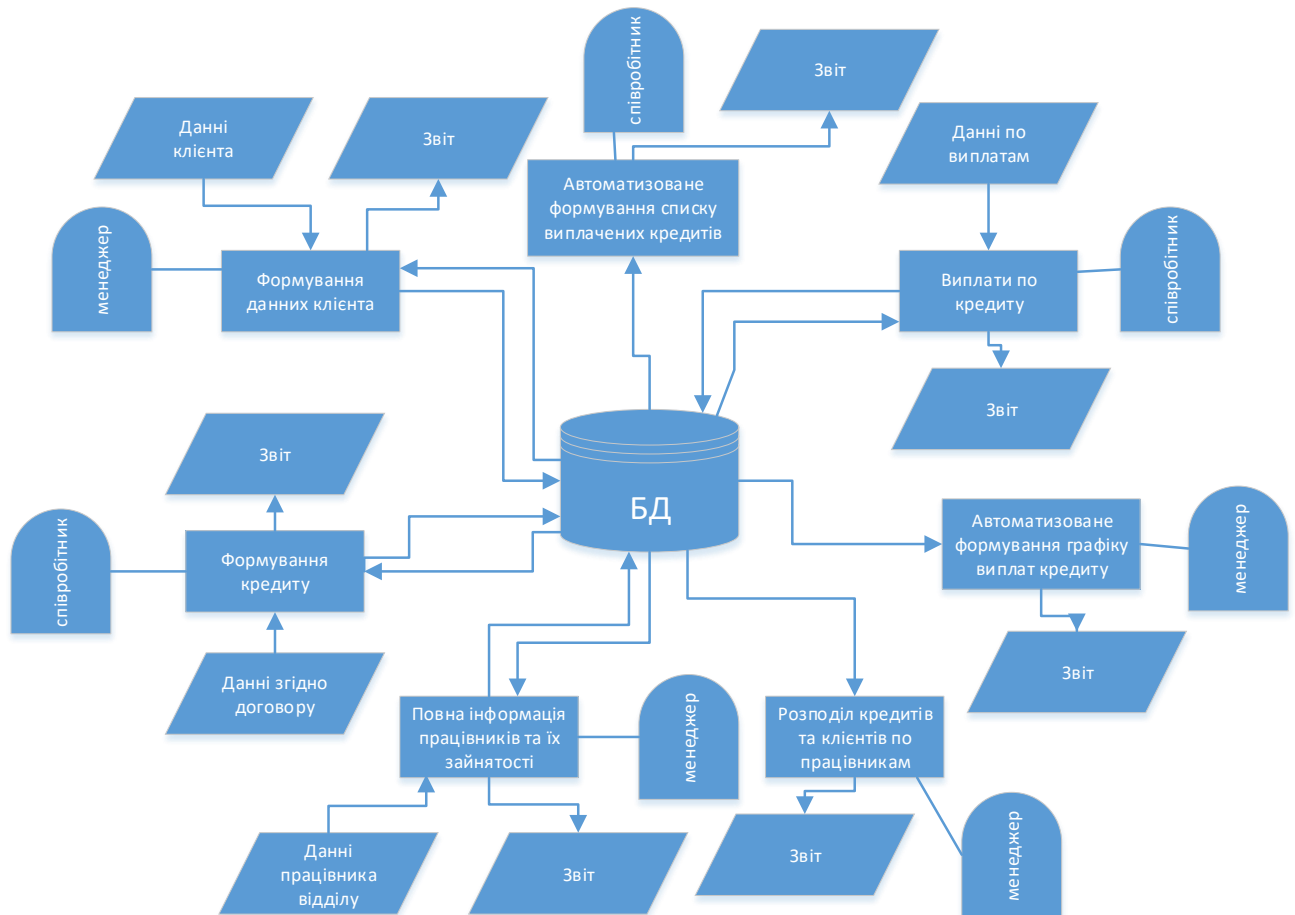


Рисунок 1.2 – Функціональна модель(блок-схема) системи.

Для створення та опису функціональної моделі використовувалась[2, 6].

1.6 Формування вхідної, довідкової та вихідної інформації бази даних системи

Відповідно до кожного процесу, які описані в пункті 2.2 потрібно виділити вхідну, довідкову та вихідну інформацію бази даних системи:

1. Реєстрація в системі співробітника. Надаються дані співробітника: ім'я, по-батькові та прізвище, назва відділу, назва категорії кредитування, номер телефону. Вихідною інформацією є ідентифікатор нового співробітника.
2. Реєстрація в системі клієнта. Надаються дані клієнта: ім'я, по-батькові та прізвище, адреса, паспортні дані, дата народження, члени сім'ї, довідка про доходи, номер телефону або назва підприємства, адреса, ім'я, по-батькові та прізвище керівника, баланс, звіт про доходи, номер телефону. Вихідною інформацією є ідентифікатор нового клієнту, звіти про клієнтів за відділом.
3. Створення в системі кредиту. Надаються дані договору по кредиту: дата, сума, термін, призначення, розмір початкового платежу, розмір%, розмір комісії за заборгованість. Вихідною інформацією є ідентифікатор нового кредиту, звіти про кредит за відділом та терміном видачі.
4. Формування графіків виплат. Надається ід кредиту. Вихідною інформацією є данні по виплату кредиту, місяць та сума виплат.
5. Формування списків виплачених кредитів. Надаються данні по усім виплатам кредитів. Вихідною інформацією є перелік кредитів які вже були сплачені.
6. Формування усіх виплат які були зроблені клієнтами. Вхідною інформацією є усі данні по виплаті кредиту. Вихідною інформацією є підрахунок остатку виплат.
7. Розподіл кредитів та клієнтів по співробітникам. Вхідні данні усі присутні у системі. Вихідні данні сформовані списки працівників та за кріплені за ними кредити та клієнти.

При формуванні вхідної, довідкової та вихідної інформації бази даних системи використовувалась [5].

					ДР.122.4142.02.ПЗ.Р1	Арк.
						17
Зм.	Арк	№ докум.	Підпис	Дата		

					ДР.122.4142.02.ПЗ.Р2									
Зм.	Арк.	№ докум.	Підпис	Дата										
Студент		Бабенко Я. А.			Розробка технічного проєкту інформаційної системи кредитного відділу банку				Літ.		Арк.	Аркушів		
Консультант											18	20		
Керівник		Маршак О. І.							НУК імені адмірала Макарова					
Н.контр.														
Зав.кафедрою		Михелєв І.Л.												

РОЗДІЛ II. РОЗРОБКА ТЕХНІЧНОГО ПРОЄКТУ ІНФОРМАЦІОННОЇ СИСТЕМИ КРЕДИТНОГО ВІДДІЛУ БАНКУ

2.1 Загальносистемні рішення

2.1.1 Діаграма використання

Діаграма прецедентів — в UML, діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання. Діаграма прецедентів є графом, що складається з множини акторів, варіантів використання обмежених границею системи, асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання. [14]

Суть даної діаграми полягає в наступному: проєктована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором. При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою. [6, 14]

З метою виявлення сутностей, та властивостей, розробимо діаграму використання на основі усіх взаємодій, процесів та учасників які є у концепції проєкту інформаційної системи.

					ДР.122.4142.02.ПЗ.Р2	Арк.
						19
Зм.	Арк	№ докум.	Підпис	Дата		

Діаграма використання процесу “ Реєстрація співробітника та взаємодія з ним” представлена на Рисунку 2.1

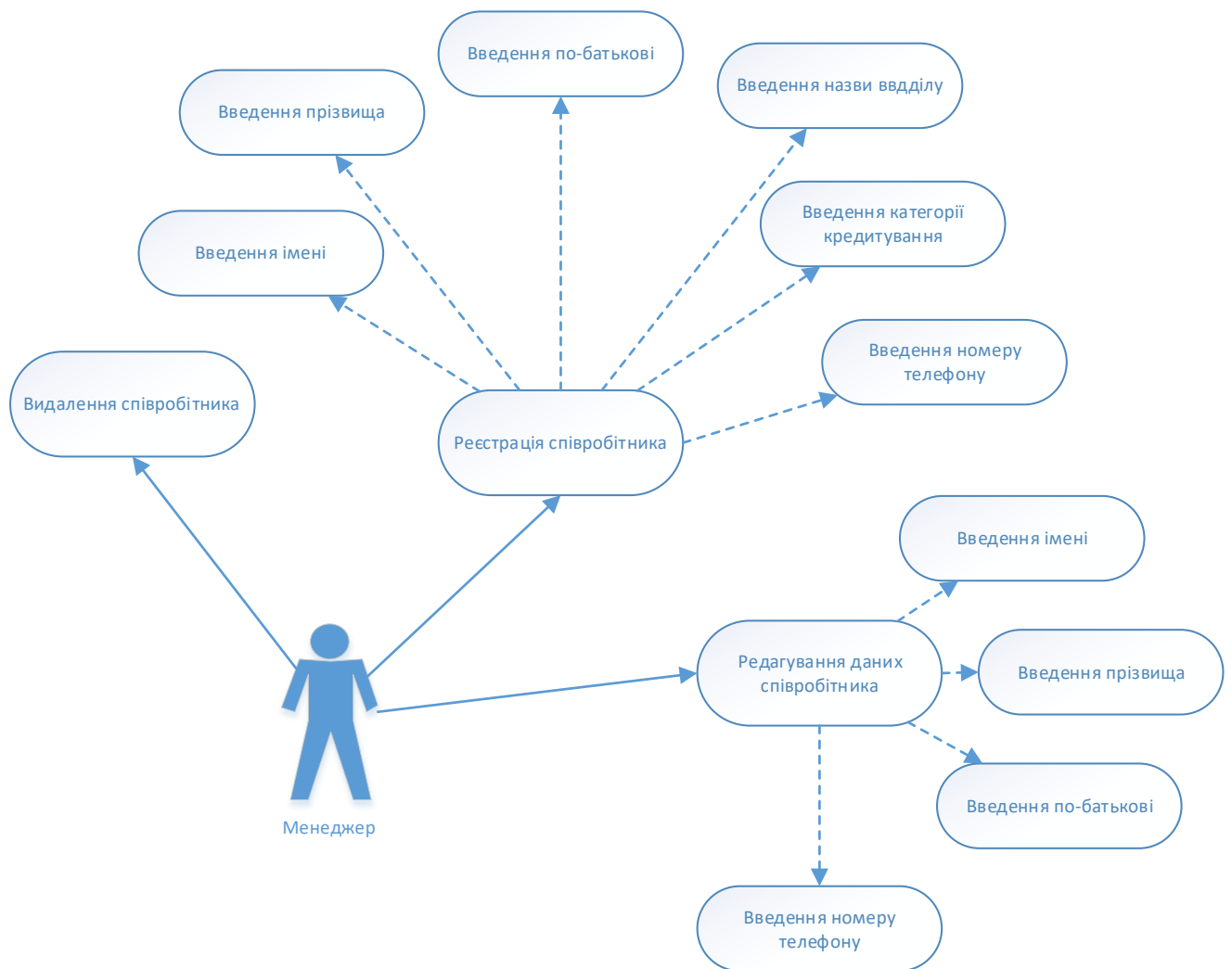


Рисунок 2.1 - Діаграма використання процесу “Реєстрація співробітника та взаємодія з ним”

Діаграма використання процесу “Реєстрація клієнта та взаємодія з ним” представлена на Рисунку 2.2

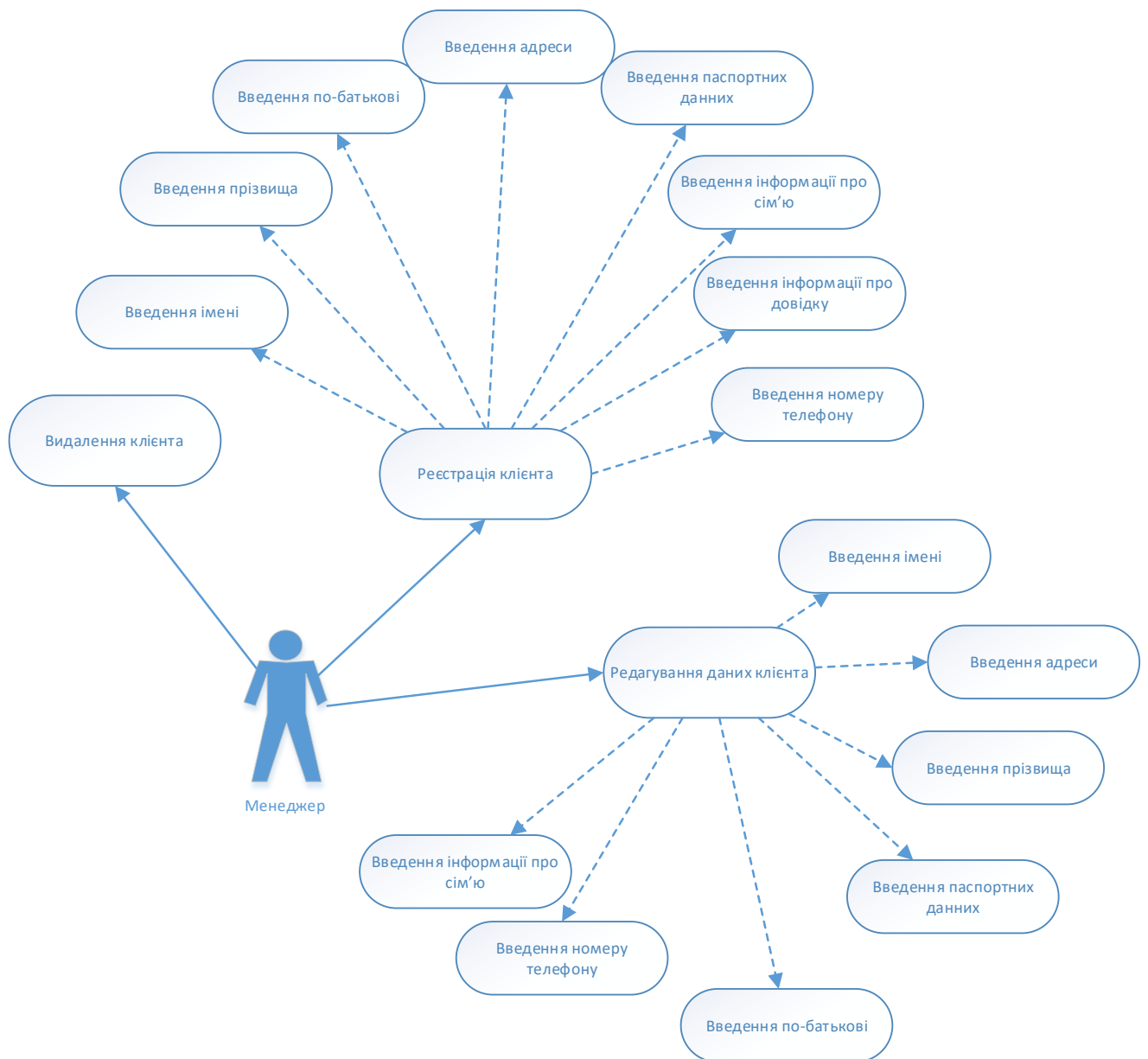


Рисунок 2.2 - Діаграма використання процесу “ Реєстрація клієнта та взаємодія з ним ”

Діаграма використання процесу “Створення кредиту та взаємодія з ним” представлена на Рисунку 2.3

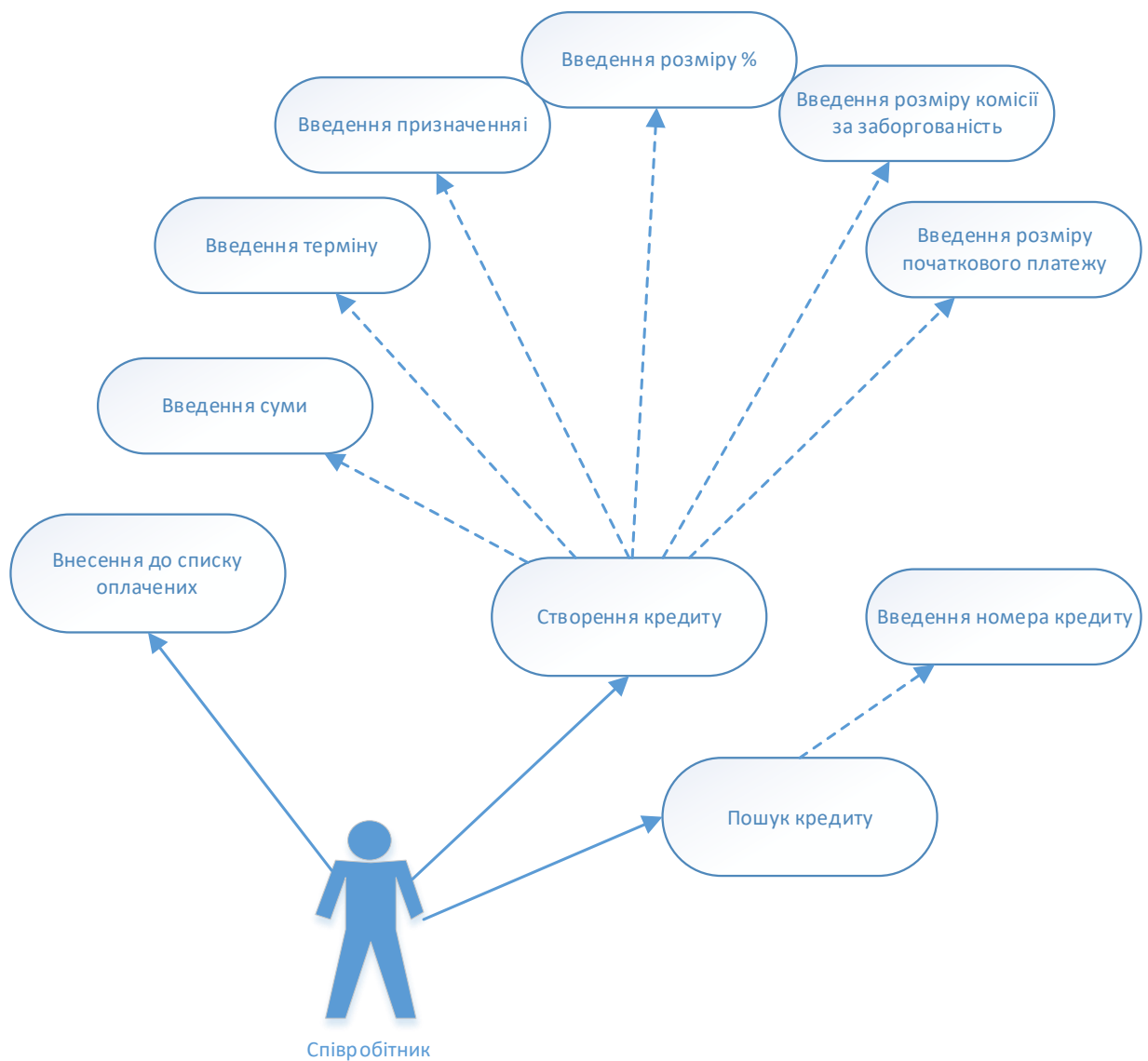


Рисунок 2.3 - Діаграма використання процесу “ Створення кредиту та взаємодія з ним ”

Діаграма використання процесу “Формування графіків” представлена на Рисунок 2.4

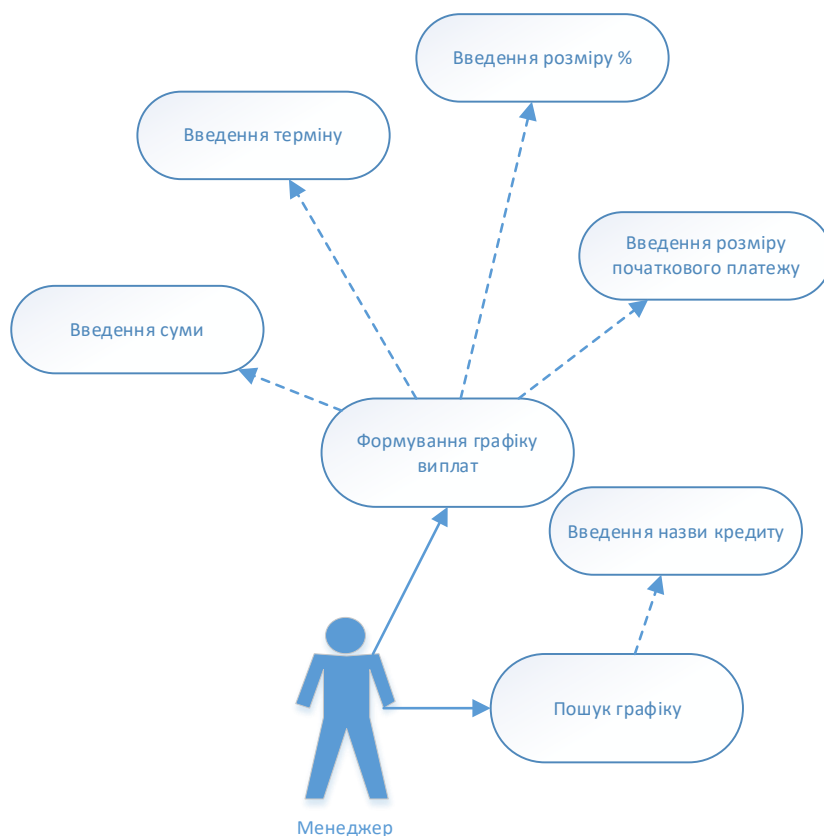


Рисунок 2.4 - Діаграма використання процесу “Формування графіку та взаємодія з ним”

Діаграма використання процесу “Розподіл клієнтів” представлена на Рисунок 2.5

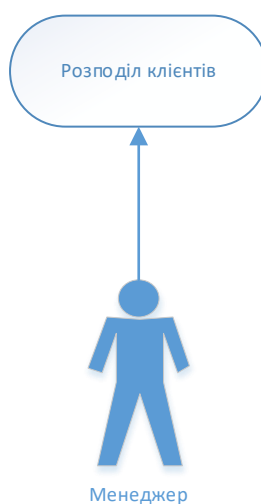


Рисунок 2.5 – Діаграма використання процесу “Розподіл клієнтів”

Діаграма використання процесу “Формування звітів” представлена на Рисунку 2.6

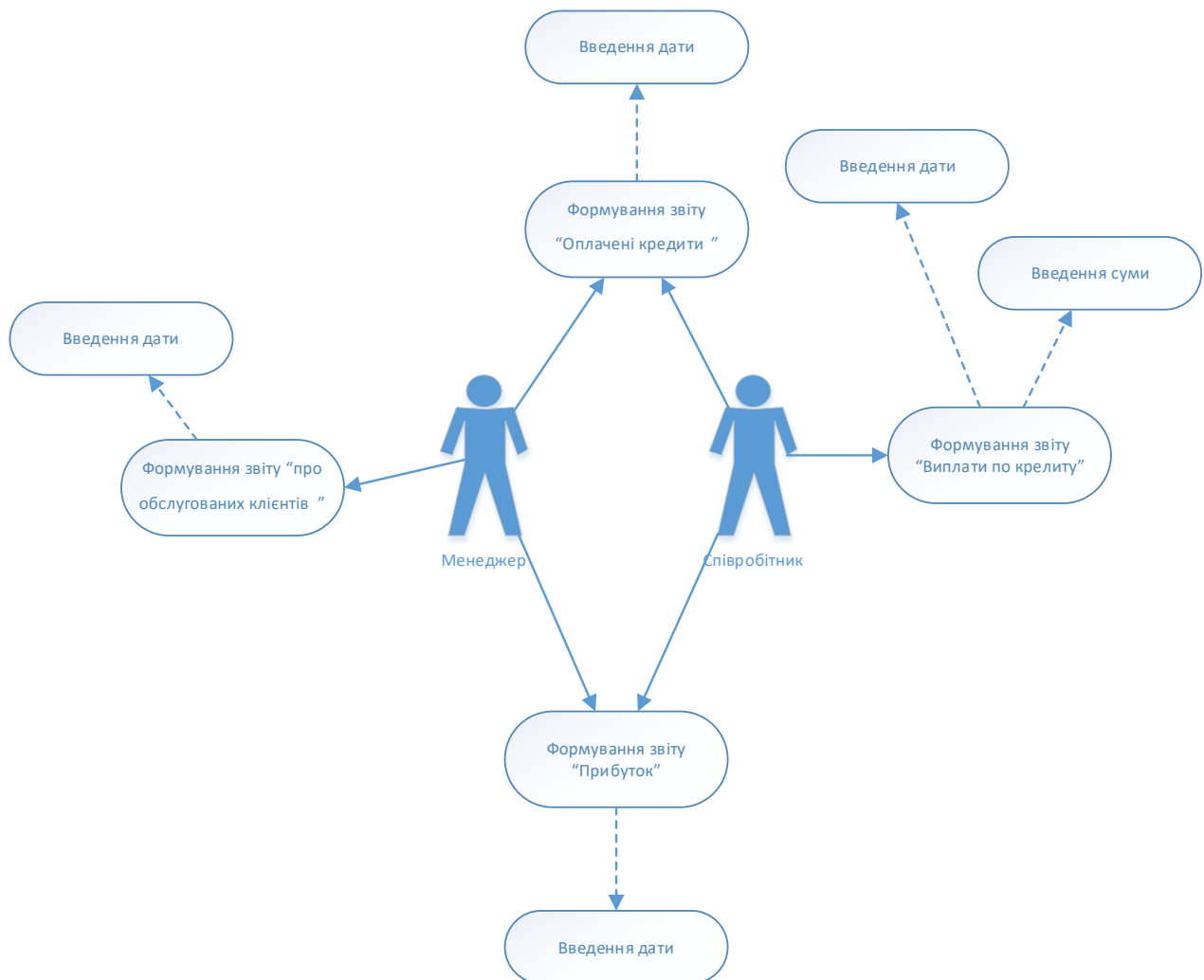


Рисунок 2.6. - Діаграма використання процесу “Формування звітів”

2.1.2 Діаграма діяльності

Діаграма діяльності - в UML, візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. [15]

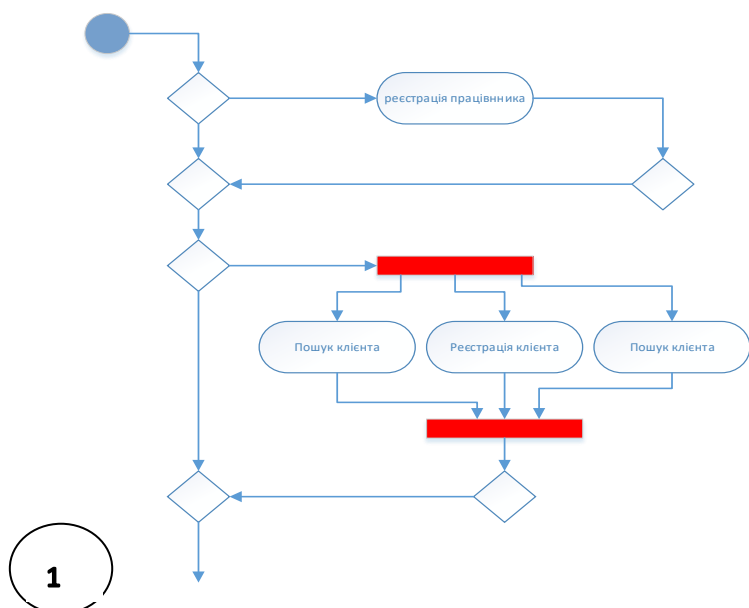
Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних

сигналів. Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності. Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку. [15]

Діаграми активностей будуються з обмеженої кількості фігур, з'єднаних стрілочками. Найважливіші типи фігур:

- скруглені прямокутники позначають дії;
- ромби позначають рішення;
- червоні риски позначають початок (розподіл) чи кінець (об'єднання) паралельних активностей;
- синій кружок позначає старт (початковий стан) процесу;
- синій кружок в колі позначає кінець (кінцевий стан).

Стрілки ведуть від старту до кінця і позначають порядок в якому відбуваються активності. [15]



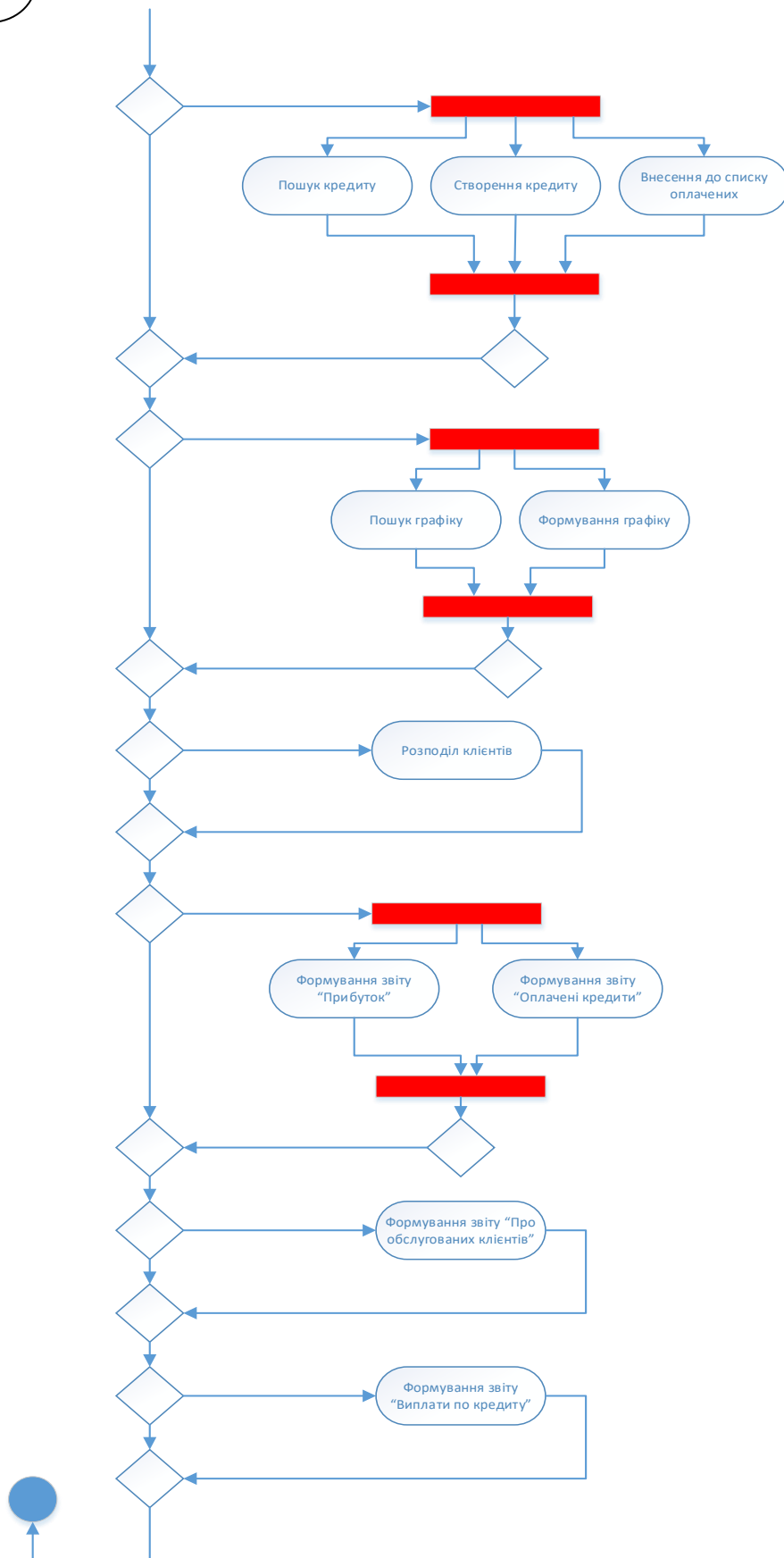


Рисунок 2.7 - Діаграма діяльності

2.2 Рішення з інформаційного забезпечення

2.2.1 Концептуальна та логічна модель базт даних

На основі причинно-наслідкових зв'язків було створено концептуальну модель бази даних, яка створюється для подальшого проектування бази даних. Концептуальна модель представлена на Рисунку 2.8

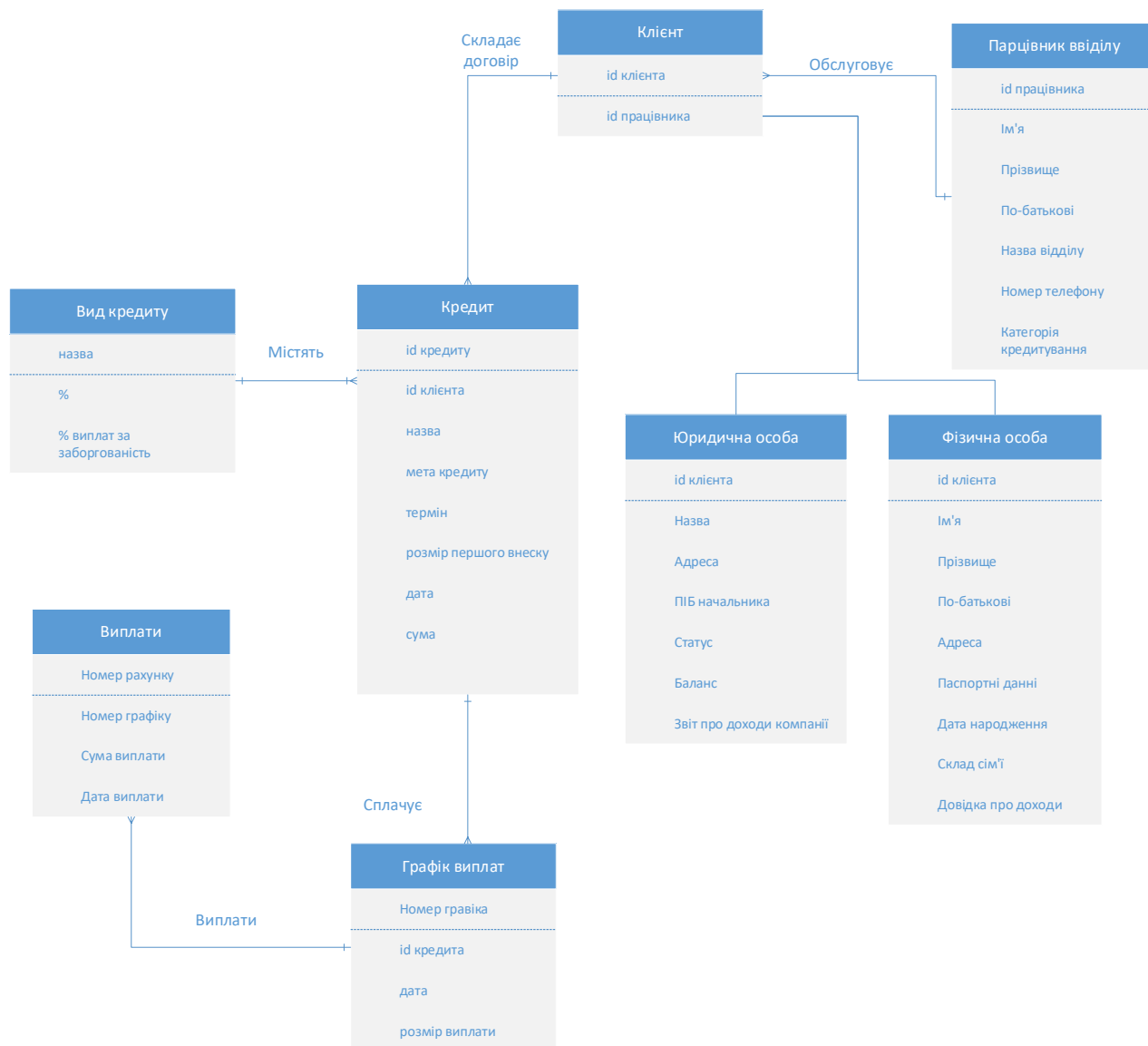


Рисунок 2.8 - Концептуальна модель бази даних

На основі концептуальної моделі було створено логічну модель даних. Модель представлена на Рисунку 2.9

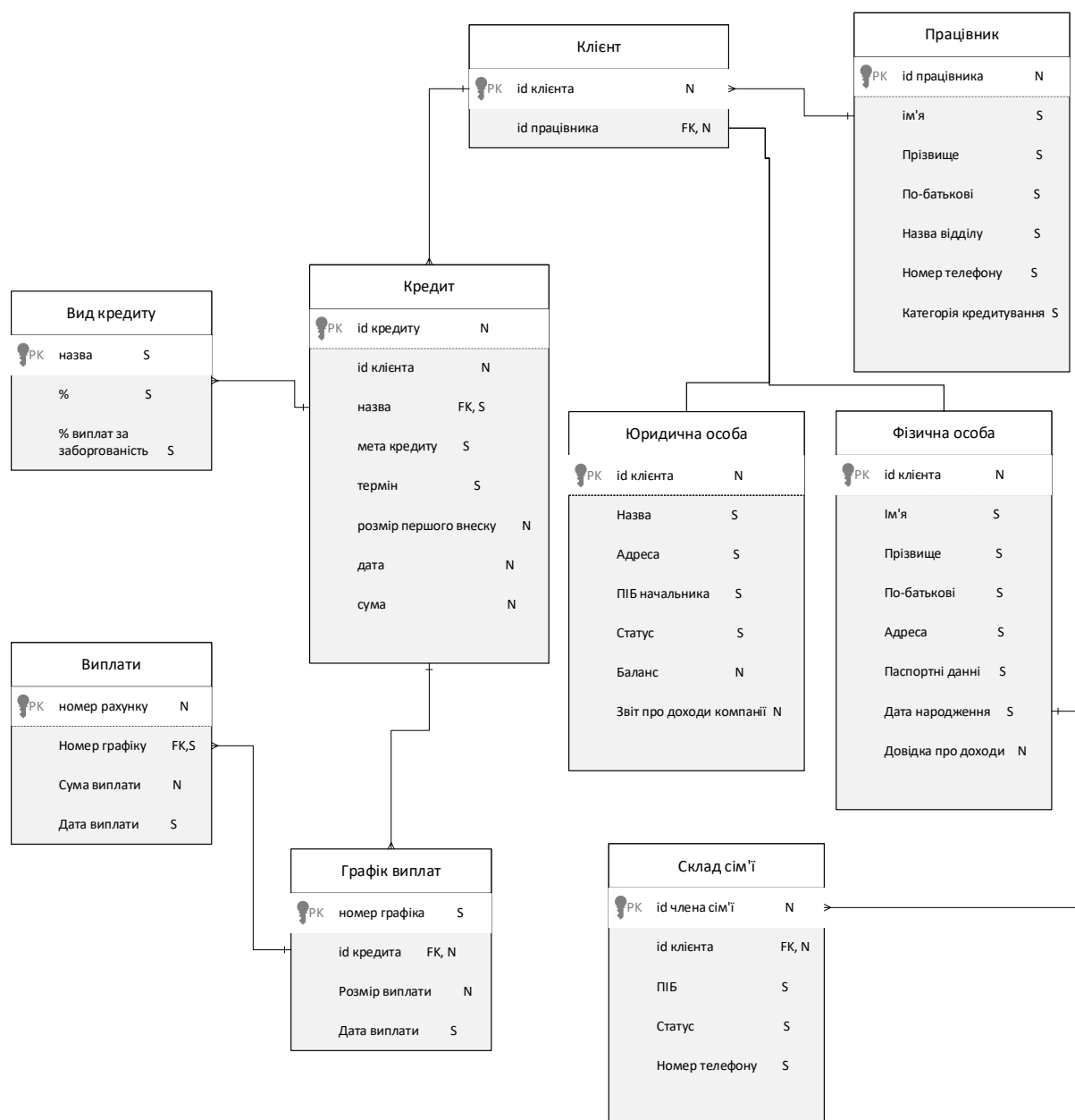


Рисунок 2.9 - Логічна модель бази даних

Згідно моделі, яка була представлена на рисунку 2.9 база даних містить наступні сутності:

Клієнт – сутність, яка описує клієнта, який був зареєстрований в системі.

Юридична особа - описує персональні дані клієнта (підприємця).

Фізична особа - описує персональні дані клієнта.

Склад сім'ї – сутність, яка описує сімейний стан кожного клієнта та представляє членів родини.

Працівники – сутність, яка описує співробітника, який працює у відділі. Описують данні працівника.

Кредит – сутність, яка описує створений кредит. Описує усі його критерії за договором.

Вид кредиту – сутність, яка описує вид, які бувають кредити.

Графік виплат – сутність, яка описує, за яким графіком клієнти повинні сплатити заборгованість.

Виплати (сплачено) – сутність, яка описує дійсні виплати усіх клієнтів.

2.2.2 Аналіз нормалізації сутностей бази даних

Нормалізація схеми бази даних – покроковий процес розбиття одного відношення (сутності) відповідно до алгоритму нормалізації на декілька відношень на базі функціональних залежностей.

Нормальна форма – властивість відношення в реляційній моделі даних, що характеризує його з точки зору надмірності, яка потенційно може призвести до логічно-помилкових результатів вибірки або зміни даних. Нормальна форма визначається як сукупність вимог, яким має задовольняти відношення.

Таким чином, схема реляційної бази даних переходить у першу, другу, третю і так далі нормальні форми. Якщо відношення відповідає критеріям нормальної форми n та всіх попередніх нормальних форм, тоді вважається, що це відношення знаходиться у нормальній формі рівня n .

Проводячи аналіз приведених раніше таблиць бази даних автосалону можна сказати наступне:

- 1) Атрибут адреса та ФІО є багатослівним. Це є порушенням першої нормальної форми, але, оскільки, ніяких дій (пошуку, заміни і тд.) з даним атрибутом відбуватися не буде, то таке порушення є допустимим у першій нормальній формі.
- 2) Так як всі відношення у БД знаходяться у першій нормальній формі та кожен не ключовий атрибут повністю залежить від первинного ключа, котрий

					ДР.122.4142.02.ПЗ.Р2	Арк.
						29
Зм.	Арк	№ докум.	Підпис	Дата		

для кожної таблиці є простим(складається з одного атрибуту), то усі відношення автоматично знаходяться у другій нормальній формі.

3) Розглядаючи не ключові атрибути у таблицях БД можна зробити висновок що вони залежать винятково від первинного ключа, а оскільки всі відношення знаходяться у другій нормальній формі то всі вони знаходяться в третій нормальній формі.

Отже всі таблиці представлені раніше знаходяться у третій нормальній формі.

2.3 Рішення з програмного забезпечення

2.3.1 Вибір інструментальних засобів

Операційна система – це базовий комплекс програм, що виконує керування апаратною складовою комп'ютера або віртуальної машини; забезпечує керування обчислювальним процесом і організовує взаємодію з користувачем. [13]

Вибір операційної системи є важливою частиною розробки концепції, бо від операційної системи залежить зручність користування, вимоги до функціоналу та якість розробки.

На сьогоднішній день є три основні операційні системи якими користується світ: Windows, Linux, Mac OS. Кожна з систем має свої переваги та недоліки.

Windows – Сімейство комерційних операційних систем корпорації Microsoft орієнтовані на застосування графічного інтерфейсу при управлінні.

Переваги

- Сумісність. Майже усі програми працюють, або мають аналоги для Windows. Драйвери є для будь-яких пристроїв.
- Підтримка. Компанія Microsoft пропонує гарну підтримку, та оновлення тому програми легко встановлювати.
- Доступність. Дуже багато корисних функцій. Інтерфейс досить дружній, зрозумілий на інтуїтивному рівні, в ньому зможе розібратися кожен.

Недоліки:

					ДР.122.4142.02.ПЗ.Р2	Арк.
						30
Зм.	Арк	№ докум.	Підпис	Дата		

- Віруси. Більшість вірусів призначені для системи Windows.
- Швидкість. Windows потребує дуже багато ресурсів. У системі дуже багато програм та функціоналу який потрібен не кожному користувачу.
- Ціна. Після кожного великого релізу доведеться купувати ОС заново. Винятком стала тільки 10 версія з тимчасовим безкоштовним оновленням.

Mac OS - сімейство операційних систем виробництва корпорації Apple.

Переваги:

- Віруси. Для цієї системи значно мала кількість загроз.
- Надійність. Систем розроблена під конкретні набори конфігурацій комп'ютерів. Тому система функціонує надійно.
- Оптимізація ПО. Усі додатки розроблені лише для Mac OS. Тому функціонування відбувається з невеликою затратою.

Недоліки:

- Ціна. Mac є дорогим на світовому ринку. Ціна зумовлена якістю та безпекою.
- Фізичний комп'ютер. Mac OS розрахована на конкретні конфігурації, які є лише у Apple.
- Сумісність. Кількість програм, написаних під Mac, сильно поступається кількості програм під Windows.

Linux - має багато аналогів та схожих версій. Але з усіх можна виділити наступне:

Переваги:

- Малі технічні вимоги. Системі досить одного ядра процесора і до 512 МБ оперативної пам'яті для роботи.
- Ціна. Багато аналогів безкоштовні. Їх можна встановити та модифікувати систему під себе.
- Різноманітність. Велика кількість версій. Існує сотні різних аналогів. Найпопулярнішими є Ubuntu, Fedora, CentOS і Mint.
- Просте застосування. Деякі аналоги досить прості у використанні, наприклад Ubuntu чи Mint.

					ДР.122.4142.02.ПЗ.Р2	Арк.
						31
Зм.	Арк	№ докум.	Підпис	Дата		

- Віруси. Для цієї системи значно мала кількість загроз.

Недоліки:

- Сумісність. Набагато менше програмного забезпечення реалізовано для цієї системи. Його важко знайти на не завжди легко встановити.

Відштовхуючи від функціоналу установи треба пам'ятати яке програмне забезпечення актуально для роботи усієї установи. Переважно це програмне забезпечення від корпорації Microsoft. Для таких установ важлива безпека та щоб усі програми мали ліцензії на використання. Тому більш вигідною та зручнішою операційною системою є Windows.

Так як реалізована інформаційна система повинна бути простою, зручною та зрозумілою. Переглядаючи усі переваги було обрано для реалізації інформаційної системи операційну систему Windows.

2.3.2 Структура програмного забезпечення

Діаграма класів – статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних та відношення. [16]

Розробимо діаграму класів на основі попередній моделі - функціональної моделі та діаграм діяльності та представлених сутностей.

2.3.3 Фізична модель бази даних

Фізична модель даних — подання дизайну даних як реалізованого чи призначеного для реалізації у системі керування базами даних. Завершена фізична модель даних включатиме всі артефакти бази даних, необхідні для створення відношень між таблицями чи для досягнення мети продуктивності, як-от індексів, визначень обмежень, зв'язаних і секціонованих таблиць. [5]

Для роботи с базою даних було використано MySQL. Модель представлена на Рисунку 2.11

					ДР.122.4142.02.ПЗ.Р2	Арк.
						32
Зм.	Арк	№ докум.	Підпис	Дата		

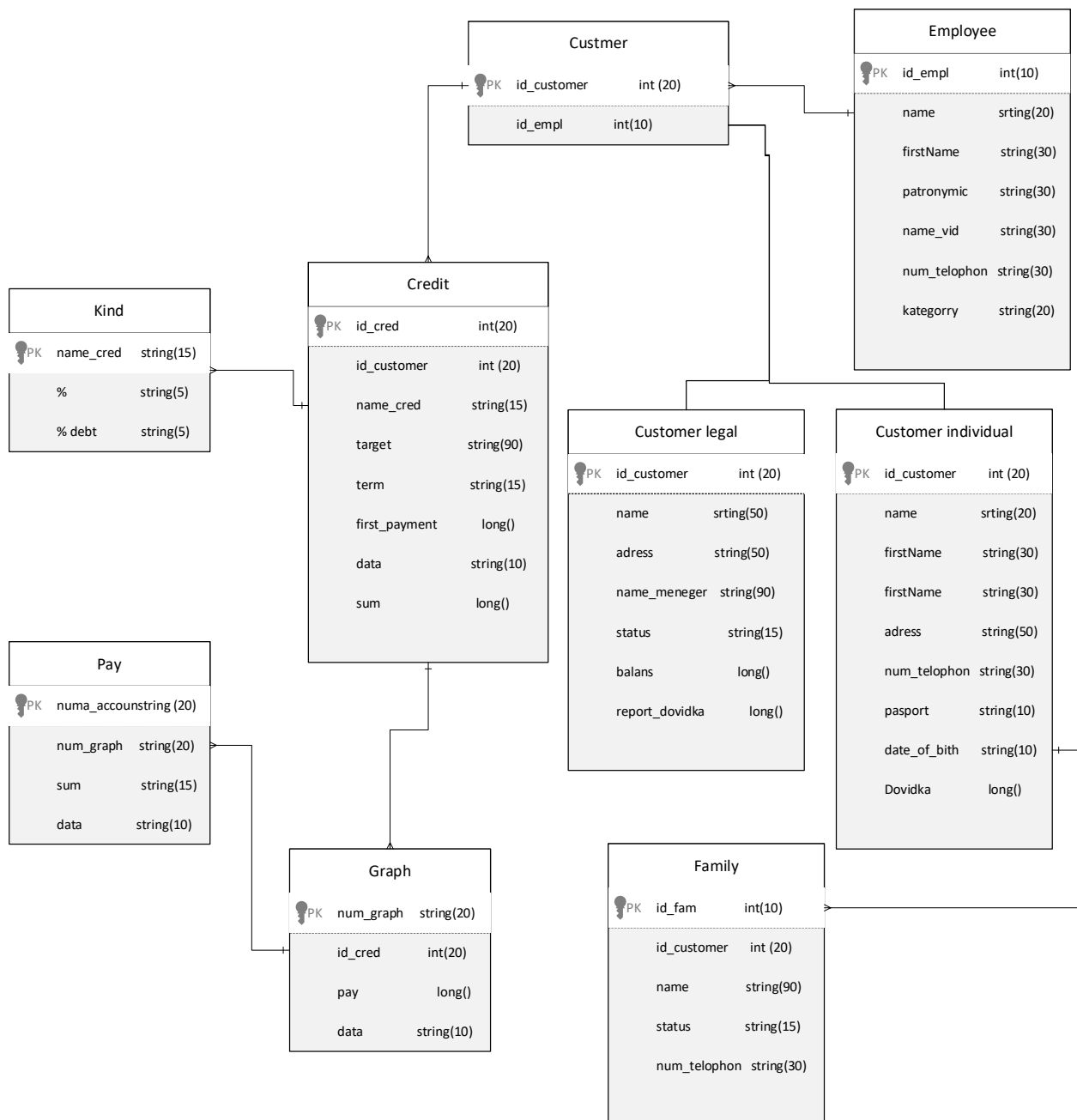


Рисунок 2.11 - Фізична модель бази даних

2.3.4 Організаційна структура

У системі повинен бути комплекс технічних засобів, для обробки даних в проєкту інформаційній системі. До складу цих засобів входять:

- 1) Збір та реєстрація інформації.
- 2) Передача даних.
- 3) Накопичення та збереження даних.
- 4) Контроль збереженої інформації.

5) Розрахунок та видачі результатів на основі отриманої інформації.

Система не повинна обмежувати кількість користувачі, які одночасно використовують систему. Потрібно обирати правильну структуру яка буде містити досконалі засоби.

Клієнт-серверна архітектура є зручною архітектурою для інформаційної системи, яка зможе відповідати усім вимогам. Ця архітектура є домінуючою концепцією у створенні розподілених мережових додатків і передбачає взаємодію та обмін даними. Має ось такі основні компоненти: набір серверів, які надають інформацію чи послуги, які звертаються до них; набір клієнтів, які використовують сервіси, що надаються серверами; мережа, яка забезпечує взаємодію між клієнтами та серверами.

Така архітектура вважаються зручною, бо для взаємодії потрібен сервер та клієнт, який звертається до сервера з проханням. Непотрібно додаткове програмне забезпечення для функціонування архітектури, адже достатньо мати браузер, який буде звертатися до сервера та виконувати роль клієнта.

Методи архітектури:

- Get – отримання даних не змінюючи їх. Легкий та найвідоміший спосіб швидко отримати дані.
- Post – дозволяє додати нові записи. Простий спосіб додати записи.
- Put – дозволяє змінити існуючі записи. Відповідає за редагування.
- Patch – дозволяє змінити ідентифікатор існуючого запису.
- Delete – дозволяю видаляти записи.

2.3.5 Методи і засоби розробки програмного забезпечення

Кожна інформаційна система повинна відповідати поставленим вимогам. При проєктуванні був застосован об'єктно-орієнтований підхід, який має такі переваги:

- Класи дозволяють проводити конструювання з компонентів, що володіють простими інструментами.
- Дані та операції разом утворюють певну сутність, усе згруповано та розташовано у своїх місцях.

					ДР.122.4142.02.ПЗ.Р2	Арк.
						34
Зм.	Арк	№ докум.	Підпис	Дата		

- Локалізація коду і даних покращує наочність і зручність супроводу програмного забезпечення.
- Інкапсуляція інформації захищає данні від зайвого доступу до них.
- Уникнення дублювання коду. Дозволяє зробити структуру коду більш уніфікованою

Було обрано мовою програмування Java. Java - це багатоплатформна мова програмування. Мова Java поставляється з досить об'ємною бібліотекою класів, вони значно спрощують розробку додатків, надаючи в розпорядження потужні засоби вирішення поширених задач..

Програмним інструментом для реалізації системи було обрано IntelliJ IDEA. Було обрано саме цей інструмент, тому що він є зручним та володіє розширеним функціоналом: зручні підказки, через що розробка ведеться швидше, можливість встановлювати різноманітні пакети, які полегшать розробку та інше.

Клієнтський інтерфейс був розроблен за допомогою HTML, CSS.

HTML є мовою розмітки сторінки та використовується у багатьох веб-сайтах. Відображається зручно для користувача. Просто та швидко можна зробити веб сторінку привабливішою, завдяки CSS, який задає стилі, кольори та відображення інформації. Завдяки цим мовам веб-сторінки стають не тільки цікавими, а ще й привабливими.

2.4 Рішення з технічного забезпечення

На основі рішень з програмного забезпечення представимо вимоги до технічного забезпечення.

Оскільки в ролі ОС для реалізації системи було обрано ОС Windows, а саме Windows 10Pro, то мінімальні системні вимоги є

- Процесор: 1 гігагерц (ГГц) або швидший процесор чи SoC
- Оперативна пам'ять: 1 гігабайт (ГБ) для 32-розрядної версії або 2 ГБ для 64-розрядної версії

					ДР.122.4142.02.ПЗ.Р2	Арк.
						35
Зм.	Арк	№ докум.	Підпис	Дата		

- Обсяг дискового простору: 16 ГБ для 32-розрядної ОС або 20 ГБ для 64-розрядної ОС
- Графічна плата: DirectX 9 або новішої версії з драйвером WDDM 1.0
- Дисплей: 800 x 600

Оскільки при реалізації системи будуть використовуватися бібліотеки, які необхідно завантажити в програму, також необхідний доступ до документації по програмних інструментах, то обов'язковою вимогою до технічного забезпечення є наявність інтернет з'єднання.

2.5 Розробка інтерфейсів

На основі моделі прецедентів, яка визначає перелік задач та на основі інформаційних моделей, які визначають перелік сутностей розробимо моделі інтерфейсів.

Перший крок до системи, це є реєстрація. Схематично сторінка реєстрації представлена на рисунку 2.12.

Співробітник Клієнт Кредит Звіти Пошук

Реєстрація співробітника

Форма реєстрації

Реєстрація

Рисунок 2.12 - Схематично сторінка реєстрації

Переглянути інформацію про клієнта або співробітника. Схематично сторінка вибору співпрацівника представлена на Рисунку 2.13

Схематична сторінка вибору співробітника. Вгорі навігаційна панель з вкладками: Співробітник, Клієнт, Кредит, Звіти. Праворуч пошуковий поле з кнопкою Пошук. Ліворуч меню Відділ з розкритою списком назв. Центральна область має поле Назва відділу та сітку з кнопками Співробітник.

Рисунок 2.13 - Схематично сторінка вибору співробітника

Переглянути інформацію про кредит графік виплат та самі виплати. Схематично сторінка кредиту представлена на Рисунку 2.14

Схематична сторінка кредиту. Вгорі навігаційна панель з вкладками: Співробітник, Клієнт, Кредит, Звіти. Праворуч пошуковий поле з кнопкою Пошук. Ліворуч поле Кредит №. Праворуч кнопки Графік виплат та Виплати. Центральна область містить заголовок Інформація по кредиту.

Рисунок 2.14 - Схематично сторінка кредиту

					ДР.122.4142.02.ПЗ.РЗ									
Зм.	Арк.	№ докум.	Підпис	Дата										
Студент		Бабенко Я.А..			Розгортання проєкту інформаційної системи управління кредитного відділу банку				Літ.		Арк.	Аркушів		
Консультант											38	2		
Керівник		Маршак О.І.							НУК імені адмірала Макарова					
Н.контр.														
Зав.кафедрою		Михелєв І.І.												

РОЗДІЛ ІІІ. РОЗГОРТАННЯ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ КРЕДИТНОГО ВІДДІЛУ БАНКУ

3.1 Розробка документації системи

В результаті розробки було реалізовано проєкт інформаційної системи управління кредитного відділу банку. Був розроблений загальний опис системи, для розуміння загального призначення системи, який представлений у Додатку Б.

Детальний опис функціоналу є у інструкції з використання системи, яка була розроблена по функціоналу проєкту інформаційної системи кредитного відділу банку. Інструкція представлена у Додатку В.

Можна ознайомитись і структурою програмного забезпечення. Усі програмні інструменти, які були використані в процесі реалізації представлені у Додатку Г.

3.2 Розгортання системи

Діаграма розгортання – діаграма, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Діаграма розгортання системи зображена.

					ДР.122.4142.02.ПЗ.РЗ	Арк.
						39
Зм.	Арк	№ докум.	Підпис	Дата		

					ДР.122.4142.02.ПЗ.Р4			
Зм.	Арк.	№ докум.	Підпис	Дата	Охорона праці	Літ.	Арк.	Аркушів
Студент		Бабенко Я.А..					40	4
Консультант								
Керівник		Маршак О.І.				НУК імені адмірала Макарова		
Н.контр.								
Зав.Каф.		Михелєв І.Л.						

РОЗДІЛ IV. ОХОРОНА ПРАЦІ

Згідно з Законом України «Про охорону праці», охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

Цей Закон визначає основні положення для реалізації конституційного права працівників на охорону їх життя і здоров'я, на безпечні і здорові умови праці під час трудової діяльності. Закон регулює відносини між роботодавцем і працівником з питань безпеки, гігієни та виробничого середовища.

4.1 Характеристика приміщення

Приміщення відділу має одностороннє природне освітлення і загальне штучне освітлення. Стіни і стеля пофарбовані у світлі кольори, підлога вкрита світлим лінолеумом. У приміщенні відсутні сильні вібрації та шкідливі речовини. Склад повітря в нормі.

У кімнаті знаходяться комп'ютери з 4-ядерним процесором і 23-дюймовим монітором, меблі.

Приміщення має довжину 10 м, ширину 9 м, висоту стелі 2,5 м. Кількість робочих місць – десять. Приміщення знаходиться на першому поверсі 10-поверхового панельного будинку. Площа – 90 м², об'єм – 225 м³. Отримані дані, наведені на Рисунку. 4.1. Нормативні значення згідно з [32]

Параметр	Норма	Дійсні параметри
Площа, S	не менше 6 м ²	90 м ²
Об'єм, V	не менше 15 м ³	225 м ³

Рисунок 4.1 - Фактичні та нормативні значення параметрів приміщення

Усі показники відповідають чинним нормам і вимогам.

					ДР.122.4142.02.ПЗ.Р4	Арк.
						41
Зм.	Арк	№ докум.	Підпис	Дата		

4.2 Мікрокліматичні умови

Згідно з [33] роботу можна віднести до категорії легка 1а. Джерелами тепла в приміщенні є люди, електроустаткування, освітлювальні прилади в темний час доби і система опалювання взимку. Співробітником виділяється до 120 ккал теплової енергії за годину. Оптимальні та фактичні значення параметрів мікроклімату приведені на Рисунку 4.2

Період року	Параметр	Оптимальний	Фактичний
Теплий	Температура	23 – 25 °С	24 °С
	Вологість	40 – 60 %	50 %
	Швидкість повітря	≤ 0.1 м/с	
Холодний	Температура	22 – 24 °С	23 °С
	Вологість	40 – 60 %	55 %
	Швидкість повітря	≤ 0.1 м/с	

Рисунок 4.2 – Показники мікроклімату

Усі показники задовольняють вимогам і є задовільними для здоров'я людини.

4.3 Освітлення

Згідно з [34] робота відноситься до Va розряду зорових робіт. Передбачається використання природного, штучного і змішаного освітлення. Природне освітлення здійснюється за допомогою трьох вікон, площа яких складає $S' = 1,8 \cdot 1,5 \cdot 3 = 8,1 \text{ м}^2$ і є боковим освітленням. У світильниках загального освітлення використовуються лампи розжарювання потужністю 75 Вт із світловим потоком лампи 940 лм. Освітлення задовольняє нормам.

4.4 Шум та вібрація

Джерелом шуму в приміщенні є комп'ютери. Вентилятори системного блоку, процесора, відеокарти є сучасними і мають низький рівень шуму.

					ДР.122.4142.02.ПЗ.Р4	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		42

Згідно з технічною документацією шум, зумовлений вентилятором в блоці складає 25 дБ, вентилятором процесора - 30 дБ, загальний, - 34 дБ. Враховуючи незначний рівень шуму від персональних комп'ютерів і незначний рівень фонового шуму від іншого устаткування, можна стверджувати, що сумарний рівень шумового забруднення приміщення не перевищує максимально допустимий рівень коригованої звукової потужності.

При роботі з персональними комп'ютерами в робочому приміщенні значення характеристик вібрації на робочих місцях не повинна перевищувати допустимих значень.

4.5 Випромінювання

У приміщенні відсутні джерела інфрачервоного, ультрафіолетового і електромагнітного випромінювання, бо монітори комп'ютерів вироблені на основі 88 рідкокристалічної матриці, підсвітка здійснюється неоновю лампою, що не має великого рівня електромагнітного випромінювання і сертифіковані в Україні. Блок живлення є екранованим і не випускає шкідливих видів випромінювання.

					ДР.122.4142.02.ПЗ.Р4	Арк.
						43
Зм.	Арк	№ докум.	Підпис	Дата		

ВИСНОВКИ

В результаті розробки проєкту інформаційної системи управління кредитного відділу банку було проведено аналіз існуючої системи управління та виявлено низку проблем: всім працівникам відділу доводиться постійно подавати окремі звіти по клієнтам та кредитам: менеджеру відділу постійно необхідно переглядати та формувати окремі графіки виплат кожного клієнта письмово; працівники відділу повинні самостійно контролювати кількість навантаження (кількість клієнтів).; менеджеру необхідно в ручному режимі закріплювати клієнтів та кредити за працівниками та інші. Були сформовані вимоги до нової системи управління проектами враховуючи усі недоліки.

Розроблена концепція системи спрямована на покращення робочого процесу співробітників установи, що призведе до підвищення ефективності роботи.

По розробленій концепції було розроблено технічний проєкт системи в якому детально описано вимоги до системи засоби. Описано структуру програмного забезпечення, процес роботи з системою та спроектовані були моделі бази даних.

На основі даних проектування було розроблено програмний продукт, який відповідає всім поставленим вимогам. Цю систему можна впровадити в кредитний відділ банку. Це підвищить продуктивність роботи працівників.

Отже, можна зробити висновок, що було досягнуто поставленої цілі роботи. Після того, як система буде впроваджена у відділ, то робота відділу значно покращас.

					ДР.122.4142.02.ПЗ								
Зм.	Арк.	№ докум.	Підпис	Дата									
Студент		Бабенко Я.А.			Список використаних джерел та додатки				Літ.		Арк.	Аркушів	
Консультант											44	17	
Керівник		Маршак О.І.											
Н.контр.													
Зав.Каф		Михелєв І.Л.											
НУК імені адмірала Макарова													

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ГОСТ 34.601-90. “Інформаційна технологія. Комплекс стандартів на автоматизовані системи. Автоматизовані системи стадії створення”. Дата введення: 01.01.92.
2. ГОСТ 19.003-80. “Схеми алгоритмів та програм. Позначення умовні графічні.”
3. Міжнародний стандарт ISO/IEC 12201-2010 Інформаційна технологія. Системна та програмна інженерія. Процеси життєвого циклу програмних засобів.
4. Методичні вказівки до виконання дипломного проектування зі спеціальності 6.050101 “Комп’ютерні науки” кваліфікації “Бакалавр інформаційних управляючих систем і технологій” / В. Ф. Ажищев, К. В. Кошкін, Г. С. Морозова. – Миколаїв : НУК, 2014. – 57 с.
5. А.В. Маслов. “Проектування інформаційних систем в економіці”. Навчальний посібник/ А.В. Маслов – Томськ. Видавництво Томського політехнічного університету, 2008. – 216 с.
6. А.В. Вендров “CASE-технології. Сучасні методи і засоби проектування інформаційних систем”. с. 123.
7. Ю.Н. Алпатов. “Модельовання процесів та систем управління”. Навчальний посібни. – СПб.: Видавництво “Лань”, 2018. – 140с. (Посібники для вузів. Спеціальна література).
8. Вікіпедія [Електронний ресурс]: <https://uk.wikipedia.org/>. Режим доступу: <https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%BD%D0%BA>
9. Теоретичні засади діяльності комерційних банків [Електронний ресурс] : <http://banking.uabs.sumdu.edu.ua/>. Режим доступу: <http://banking.uabs.sumdu.edu.ua/images/department/banking/discip/gik/bs/Tema13.pdf>

					ДР.122.4142.02.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		45

10. Stud me [Електронний ресурс]: <http://studme.com.ua/> Режим доступу: http://studme.com.ua/1347071110546/bankovskoe_delo/kredit_kreditnaya_sistema.htm
11. Stud me [Електронний ресурс]: <http://studme.com.ua/>. Режим доступу: http://studme.com.ua/1578061510547/bankovskoe_delo/funktsii_kredita.htm
12. Вікіпедія [Електронний ресурс]: <https://uk.wikipedia.org/>. Режим доступу: https://uk.wikipedia.org/wiki/%D0%A1%D0%BF%D0%BE%D0%B6%D0%B8%D0%B2%D1%87%D0%B8%D0%B9_%D0%BA%D1%80%D0%B5%D0%B4%D0%B8%D1%82
13. Fizmat Wiki [Електронний ресурс]: <http://group.fizmat.tnpu.edu.ua/> Режим доступу: http://group.fizmat.tnpu.edu.ua/index.php/%D0%9F%D0%BE%D0%BD%D1%8F%D1%82%D1%82%D1%8F_%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D1%96%D0%B9%D0%BD%D0%BE%D1%97_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B8_%D1%82%D0%B0_%D1%97%D1%97_%D1%81%D0%BA%D0%BB%D0%B0%D0%B4%D0%BE%D0%B2%D1%96
14. Вікіпедія [Електронний ресурс]: <https://uk.wikipedia.org/>. Режим доступу: uk.wikipedia.org/wiki/Діаграма_прецедентів
15. Вікіпедія [Електронний ресурс]: <https://uk.wikipedia.org/>. Режим доступу: uk.wikipedia.org/wiki/Діаграма_діяльності
16. Вікіпедія [Електронний ресурс]: <https://uk.wikipedia.org/>. Режим доступу: https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0_%D0%BA%D0%BB%D0%B0%D1%81%D1%96%D0%B2
17. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин /ДСанПіН 3.3.2.007-98. – К.: Постанова Головного державного санітарного лікаря України, 1998. - № 7.

					ДР.122.4142.02.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		46

18. Санітарні норми мікроклімату виробничих приміщень : ДСН 3.3.6.042-99. – К., 2000.- С.16
19. Природне і штучне освітлення : ДБН В.2.5-28:2006. – К. : Міністерство будівництва, архітектури та житлово-комунального господарства України, 2006. – 68 с. – (Національні стандарти України).

					ДР.122.4142.02.ПЗ	Арк.
						47
Зм.	Арк	№ докум.	Підпис	Дата		

ДОДАТКИ

Додаток А – Технічне завдання на розробку

1. Вступ

1.1. Назва – Проєкт інформаційної системи управління кредитного відділу банку.

1.2. Призначення програми – вдосконалити процеси управління, які відбуваються в кредитному відділі банку під час роботи шляхом створення звітів на основі збережених в системі даних про виплати та кредити, ведення звітності по співробітникам та клієнтам.

1.3. Область застосування - кредитні відділи банків для зручного розподілу праці, систематизації роботи усього відділу.

2. Призначення розробки

Система повинна надавати можливість створення кредитів за усіх пунктів договору, зручно формувати звіти за запитом. Допомогати менеджеру контролювати навантаження співробітників відділів та допомагати розподілити клієнтів. Надавати можливість співробітнику зручно відстежити та знайти інформацію про клієнта.

Усі функції спрямовані на мінімізацію витраченого часу співробітників та підвищення продуктивності роботи адже саме це головне призначення системи.

3. Вимоги до системи

3.1. Вимоги до функціональних характеристик

Система повинна містити наступні функціональні характеристики:

- Реєстрація в системі співробітника. Взяття співробітника на облік установи.
- Реєстрація в системі клієнта. Взяття клієнта на облік.
- Створення в системі кредиту. Внесення усіх даних за договором.
- Формування графіків виплат.

					ДР.122.4142.02.ПЗ	Арк.
						48
Зм.	Арк	№ докум.	Підпис	Дата		

- Формування списків виплачених кредитів.
- Формування усіх виплат які були зроблені клієнтами.
- Розподіл кредитів та клієнтів по співробітникам.

3.2. Вимоги до надійності

3.2.1. Вимоги до забезпечення надійності функціонування програми

Надійне функціонування інтернету на персональних комп'ютерів у офісі. Надійний інтрнет або WiFi з'єднання. Відкритий доступ лице на інтернеті установи.

.

3.3. Умови експлуатації

Системою можуть користуватися працівники кредитного відділу банку: співробітники та менеджер. Для використання системи необхідна наявність інтернет з'єднання. Однією з умов є наявність браузера у користувача, так як система працює у інтернеті

3.4. Вимоги до інформаційної та програмної сумісності

3.4.1. Вимоги до інформаційних структур і методів розв'язання

Вимоги до інформаційних структур і методів розв'язання відсутні.

3.4.2. Вимоги до програмних засобів , які використовуються програмою

Програмні засоби що використовуються мають бути вільно поширювані але мати систему безпеки

3.5. Вимоги до маркування та упаковки

3.5.1. Вимоги до маркування

- Читабельність тексту, чіткість ілюстрацій.
- Спокійні кольори

3.5.2. Вимоги до упаковки

Вимоги до упаковки відсутні.

3.6. Вимоги до транспортування та зберігання

Вимоги до транспортування програми відсутня.

3.7. Спеціальні вимоги

					ДР.122.4142.02.ПЗ	Арк.
						49
Зм.	Арк	№ докум.	Підпис	Дата		

Спеціальні вимоги відсутні.

4. Вимоги до програмної документації

4.1. Попередній склад програмної документації

Попередній склад програмної документації повинен включати в себе:

- технічне завдання;
- устав проекту.

4.2. Склад програмної документації

Програмна документація складається з наступних програмних документів:

- Договір з клієнтом.
- Договір по кредиту.
- Договір про виплати.

5. Техніко-економічні показники

5.1. Економічні переваги розробки

Орієнтовна економічна ефективність не розраховується.

6. Стадії розробки програмного забезпечення

Розробка повинна включати 5 стадій:

1. Технічне завдання.
2. Ескізний проект.
3. Технічний проект.
4. Робочий проект.
5. Впровадження.

					ДР.122.4142.02.ПЗ	Арк.
						50
Зм.	Арк	№ докум.	Підпис	Дата		

Додаток Б – Загальний опис системи

Проект інформаційної системи управління кредитного відділу банку є інструментом, який позбавить менеджера виконувати великі кількості ручних робіт, які включають в себе: формування звітів, розподіл клієнтів, пошук інформації. Основними функціональними характеристиками системи є:

- Створення та редагування клієнтів та співробітників.
- Створення, кредиту та контроль за виплатами .
- Розподіл клієнтів за відділом на співробітником.
- Формування звітів, які необхідні для звітності прибутків.

Система реалізована за допомогою мови програмування Java, HTML CSS з використанням додаткових бібліотек та має зручний та інтуїтивно зрозумілий інтерфейс.

Структура всіх сутностей повністю відповідає фізичній моделі бази даних, яка була розроблена в роботі. Для зручного та швидкого розгортання системи всі сутності в таблиці створювалися міграціями.

					ДР.122.4142.02.ПЗ	Арк.
						51
Зм.	Арк	№ докум.	Підпис	Дата		

Додаток В – Структура програмного забезпечення

В результаті реалізації проєкту інформаційної системи управління кредитним відділом банку було створено додаток з використанням мови програмування Java. З використанням Servlet API, та технологію JSP.

Було обрано мовою програмування Java. Java - це багатоплатформна мова програмування. Мова Java поставляється з досить об'ємною бібліотекою класів, вони значно спрощують розробку додатків, надаючи в розпорядження потужні засоби вирішення поширених задач..

Програмним інструментом для реалізації системи було обрано IntelliJ IDEA. Було обрано саме цей інструмент, тому що він є зручним та володіє розширеним функціоналом: зручні підказки, через що розробка ведеться швидше, можливість встановлювати різноманітні пакети, які полегшить розробку та інше.

Клієнтський інтерфейс був розроблен за допомогою HTML, CSS.

HTML є мовою розмітки сторінки та використовується у багатьох веб-сайтах. Відображається зручно для користувача. Просто та швидко можна зробити веб сторінку привабливішою, завдяки CSS, який задає стилі, кольори та відображення інформації. Завдяки цим мовам веб-сторінки стають не тільки цікавими, а ще й привабливими.

					ДР.122.4142.02.ПЗ	Арк.
						52
Зм.	Арк	№ докум.	Підпис	Дата		

Додаток Г – Тексти програмних модулів системи

Реалізація проєкту інформаційної системи складається з багатьох модулів. У Лістингу представлені окремі частини проєкту.

Лістинг 1 DB.java

```
package com.company;

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
import java.sql.*;
import java.util.Scanner;

public class Main {

    static final String SQL_Cus = "CREATE TABLE cust" +
        "(id_cust int PRIMARY KEY NOT NULL ," +
        "id_employee int NOT NULL );";
    static final String SQL_Cred = "CREATE TABLE credit" +
        "(id_credit int PRIMARY KEY NOT NULL ," +
        "id_cust int NOT NULL," +
        "name varchar(15) NOT NULL," +
        "loanpurpose varchar(50) NOT NULL," +
        "term varchar(10) NOT NULL," +
        "firstpay varchar(20) NOT NULL," +
        "sum int );";
    static final String SQL_Phy = "CREATE TABLE physicalpeson" +
        "(id_cust int PRIMARY KEY NOT NULL," +
        "fullname varchar(80) NOT NULL," +
        "adress varchar(50) NOT NULL," +
        "balance varchar(10) NOT NULL," +
        "pasport varchar(10) NOT NULL," +
        "dateofbith varchar(10) NOT NULL);";
    static final String SQL_Ent = "CREATE TABLE entity" +
        "(id_cust int PRIMARY KEY NOT NULL," +
        "companyname varchar(40) NOT NULL," +
        "chief varchar(80) NOT NULL," +
        "adress varchar(50) NOT NULL," +
        "balance varchar(10) NOT NULL," +
        "numberreferense int );";
    static final String SQL_Emp = "CREATE TABLE employee" +
        "(id_employee int PRIMARY KEY NOT NULL," +
        "fullname varchar(80) NOT NULL," +
        "department varchar(30) NOT NULL);";
    static final String SQL_Fam = "CREATE TABLE family" +
        "(id_person int PRIMARY KEY NOT NULL," +
        "id_cust int NOT NULL," +
        "fullname varchar(80) NOT NULL," +
        "statut_person varchar(20) NOT NULL);";
    static final String SQL_Kind = "CREATE TABLE kindcredit" +
        "(name varchar(15) PRIMARY KEY NOT NULL," +
        "percent varchar(5) NOT NULL," +
        "interestondebt varchar(5) NOT NULL);";
    static final String SQL_Paid = "CREATE TABLE paidof" +
        "(number_account int PRIMARY KEY NOT NULL," +
        "number_sch varchar(10) NOT NULL," +
        "sum_paid int," +
```

					ДР.122.4142.02.ПЗ	Арк.
						53
Зм.	Арк	№ докум.	Підпис	Дата		

```

        "date varchar(10) NOT NULL);"");
static final String SQL_sched = "CREATE TABLE schedule" +
    "(number_sch varchar(10) PRIMARY KEY NOT NULL," +
    "id_credit int," +
    "sum int," +
    "month_pay varchar(10) NOT NULL);"");
private DatabaseMetaData metadata;
private Connection connection;
private ResultSet resultSet;
private PreparedStatement pt;
private Statement st;
Scanner in = new Scanner(System.in);
int iCus = 11;
int iEn = 506;
int iCred = 2016;

private static final String URL =
"jdbc:mysql://localhost:3306/cust?autoReconnect=true&useSSL=false&useLegacyDatetim
eCode=false&serverTimezone=UTC";
private static final String USERNAME = "root";
private static final String PASSWORD = "";

public static void main(String[] args) throws SQLException{
    new Main().run();
}

public void run() throws SQLException{
    setConnection();
    String[][] cust = new String[100][];
    String[][] physical = new String[100][];
    String[][] entity = new String[100][];
    String[][] employee = new String[100][];
    String[][] credit = new String[100][];
    String[][] kindcred = new String[100][];
    String[][] family = new String[100][];
    String[][] sched = new String[100][];
    String[][] paidof = new String[100][];
    readFile("cust.txt", cust);
    readFile("physical.txt", physical);
    readFile("entity.txt", entity);
    readFile("employee.txt", employee);
    readFile("credit.txt", credit);
    readFile("kindcredit.txt", kindcred);
    readFile("family.txt", family);
    readFile("schedule.txt", sched);
    readFile("paidof.txt", paidof);
    putInToEnt(entity);
    putInToEm(employee);
    putInToPhy(physical);
    putInToCust(cust);
    putInToKind(kindcred);
    putInToCred(credit);
    putInToFam(family);
    putInToSch(sched);
    putInToPaid(paidof);
    menu();
    st.close();
    pt.close();
    connection.close();
}

```

					ДР.122.4142.02.ПЗ	Арк.
						54
Зм.	Арк	№ докум.	Підпис	Дата		

```

private void setConnection() throws SQLException {
    connection = DriverManager.getConnection(URL, USERNAME, PASSWORD);
    st = connection.createStatement();
    metadata = connection.getMetaData();
    resultSet = metadata.getTables(null, null, "credit", null);
    if(resultSet.next()) {
        st.executeUpdate(SQL_Cus);
        st.executeUpdate(SQL_Cred);
        st.executeUpdate(SQL_Phy);
        st.executeUpdate(SQL_Ent);
        st.executeUpdate(SQL_Emp);
        st.executeUpdate(SQL_Fam);
        st.executeUpdate(SQL_Kind);
        st.executeUpdate(SQL_Paid);
        st.executeUpdate(SQL_sched);
        st.executeUpdate("ALTER TABLE cust ADD FOREIGN KEY (id_employee)
REFERENCES employee (id_employee);");
        st.executeUpdate("ALTER TABLE credit ADD FOREIGN KEY (id_cust)
REFERENCES cust (id_cust);");
        st.executeUpdate("ALTER TABLE credit ADD FOREIGN KEY (name) REFERENCES
kindcredit (name);");
        st.executeUpdate("ALTER TABLE schedule ADD FOREIGN KEY (id_credit)
REFERENCES credit (id_credit);");
        st.executeUpdate("ALTER TABLE paidof ADD FOREIGN KEY (number_sch)
REFERENCES schedule (number_sch);");
        st.executeUpdate("ALTER TABLE family ADD FOREIGN KEY (id_cust)
REFERENCES physicalpeson (id_cust);");
    }else {st.executeUpdate(SQL_Cus);
        st.executeUpdate(SQL_Cred);
        st.executeUpdate(SQL_Phy);
        st.executeUpdate(SQL_Ent);
        st.executeUpdate(SQL_Emp);
        st.executeUpdate(SQL_Fam);
        st.executeUpdate(SQL_Kind);
        st.executeUpdate(SQL_Paid);
        st.executeUpdate(SQL_sched);
        st.executeUpdate("ALTER TABLE cust ADD FOREIGN KEY (id_employee)
REFERENCES employee (id_employee);");
        st.executeUpdate("ALTER TABLE credit ADD FOREIGN KEY (id_cust)
REFERENCES cust (id_cust);");
        st.executeUpdate("ALTER TABLE credit ADD FOREIGN KEY (name) REFERENCES
kindcredit (name);");
        st.executeUpdate("ALTER TABLE schedule ADD FOREIGN KEY (id_credit)
REFERENCES credit (id_credit);");
        st.executeUpdate("ALTER TABLE paidof ADD FOREIGN KEY (number_sch)
REFERENCES schedule (number_sch);");
        st.executeUpdate("ALTER TABLE family ADD FOREIGN KEY (id_cust)
REFERENCES physicalpeson (id_cust);");
    }
}

private void readFile(String filename, String[][] objects) {
    String str;
    try (BufferedReader bufferedReader = new BufferedReader(new
FileReader(filename))) {
        int i = 0;
        while ((str = bufferedReader.readLine()) != null){
            objects[i] = str.split(";");
            i++;
        }
    } catch (IOException ex) {
        System.err.println("Error reading file");
    }
}

```

```

    }
}

private void putInToCust(String[][] cust) throws SQLException {
    pt = connection.prepareStatement("INSERT INTO cust VALUE (?,?)");
    for (String[] str : cust) {
        if(str!=null){
            pt.setInt(1,Integer.parseInt(str[0]));
            pt.setInt(2,Integer.parseInt(str[1]));
            pt.execute();
        }
    }
}

private void addPhy(int i) throws SQLException {
    System.out.print("");
    String m = in.nextLine();
    System.out.print("Full name:");
    String fname = in.nextLine();
    System.out.print("Address:");
    String adress = in.nextLine();
    System.out.print("balance:");
    String balance = in.nextLine();
    System.out.print("Pasport:");
    String pas = in.nextLine();
    System.out.print("Date of Birth (dd.mm.yyyy):");
    String db = in.nextLine();

    Physical physical = new Physical(i, fname, adress, balance, pas, db);
    addPhyDb(physical);
}

private void addPhyDb(Physical physical) throws SQLException {
    pt = connection.prepareStatement("INSERT INTO physicalpeson VALUE
(?,?,?,?,?,?,?)");
    pt.setInt(1,physical.getId_cust());
    pt.setString(2, physical.getFullname());
    pt.setString(3,physical.getAdress());
    pt.setString(4,physical.getBalance());
    pt.setString(5,physical.getPasport());
    pt.setString(6,physical.getDateofbirth());
    pt.execute();
}

private void putInToPhy(String[][] phy) throws SQLException {
    pt = connection.prepareStatement("INSERT INTO physicalpeson VALUE
(?,?,?,?,?,?,?)");
    for (String[] p : phy) {
        if(p!=null){
            pt.setInt(1,Integer.parseInt(p[0]));
            pt.setString(2,p[1]);
            pt.setString(3,p[2]);
            pt.setString(4,p[3]);
            pt.setString(5,p[4]);
            pt.setString(6,p[5]);
            pt.execute();
        }
    }
}

private void addEn(int i) throws SQLException {
    System.out.print("");
    String m = in.nextLine();

```

					ДР.122.4142.02.ПЗ	Арк.
						56
Зм.	Арк	№ докум.	Підпис	Дата		

```

        System.out.print("Company name:");
        String fname = in.nextLine();
        System.out.print("Full name chief:");
        String chief = in.nextLine();
        System.out.print("Adress:");
        String adress = in.nextLine();
        System.out.print("Balance:");
        String balance = in.nextLine();
        System.out.print("Number referense:");
        int db = in.nextInt();

        Entity entity = new Entity(i, fname, chief, adress, balance, db);
        addenDb(entity);
    }

    private void addenDb(Entity entity) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO entity VALUE
        (?, ?, ?, ?, ?, ?)");
        pt.setInt(1, entity.getId_cust());
        pt.setString(2, entity.getCompanyname());
        pt.setString(3, entity.getChief());
        pt.setString(4, entity.getAdress());
        pt.setString(5, entity.getBalance());
        pt.setInt(6, entity.getNuberreferense());
        pt.execute();
    }

    private void putInToEnt(String[][] ent) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO entity VALUE
        (?, ?, ?, ?, ?, ?)");
        for (String[] str : ent) {
            if(str!=null) {
                pt.setInt(1, Integer.parseInt(str[0]));
                pt.setString(2, str[1]);
                pt.setString(3, str[2]);
                pt.setString(4, str[3]);
                pt.setString(5, str[4]);
                pt.setInt(6, Integer.parseInt(str[5]));
                pt.execute();
            }
        }
    }

    private void addCred(int i) throws SQLException {
        System.out.print("Customer number:");
        int num = in.nextInt();
        System.out.print("");
        String m = in.nextLine();
        System.out.print("Name credit:");
        String nam = in.nextLine();
        System.out.print("Loan purpose:");
        String loan = in.nextLine();
        System.out.print("Term:");
        String term = in.nextLine();
        System.out.print("First pay:");
        String fpay = in.nextLine();
        System.out.print("Sum:");
        int sum = in.nextInt();

        Credit credit = new Credit(i, num, nam, loan, term, fpay, sum);
        addCrdb(credit);
    }

    private void addCrdb(Credit credit) throws SQLException{
        pt = connection.prepareStatement("INSERT INTO credit VALUE

```

					ДР.122.4142.02.ПЗ	Арк.
						57
Зм.	Арк	№ докум.	Підпис	Дата		

```

        (?, ?, ?, ?, ?, ?, ?)");
        pt.setInt(1, credit.getId_credit());
        pt.setInt(2, credit.getId_cust());
        pt.setString(3, credit.getName());
        pt.setString(4, credit.getLoanpurpose());
        pt.setString(5, credit.getTerm());
        pt.setString(6, credit.getFirstpay());
        pt.setInt(7, credit.getSum());
        pt.execute();
    }

    private void putInToCred(String[][] cred) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO credit VALUE
        (?, ?, ?, ?, ?, ?, ?, ?)");
        for (String[] str : cred) {
            if (str != null) {
                pt.setInt(1, Integer.parseInt(str[0]));
                pt.setInt(2, Integer.parseInt(str[1]));
                pt.setString(3, str[2]);
                pt.setString(4, str[3]);
                pt.setString(5, str[4]);
                pt.setString(6, str[5]);
                pt.setInt(7, Integer.parseInt(str[6]));
                pt.execute();
            }
        }
    }

    private void putInToEm(String[][] empl) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO employee VALUE (?, ?, ?)");
        for (String[] str : empl) {
            if (str != null) {
                pt.setInt(1, Integer.parseInt(str[0]));
                pt.setString(2, str[1]);
                pt.setString(3, str[2]);
                pt.execute();
            }
        }
    }

    private void putInToFam(String[][] fam) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO family VALUE (?, ?, ?, ?)");
        for (String[] str : fam) {
            if (str != null) {
                pt.setInt(1, Integer.parseInt(str[0]));
                pt.setInt(2, Integer.parseInt(str[1]));
                pt.setString(3, str[2]);
                pt.setString(4, str[3]);
                pt.execute();
            }
        }
    }

    private void putInToKind(String[][] kind) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO kindcredit VALUE (?, ?, ?)");
        for (String[] str : kind) {
            if (str != null) {
                pt.setString(1, str[0]);
                pt.setString(2, str[1]);
                pt.setString(3, str[2]);
                pt.execute();
            }
        }
    }

```

					ДР.122.4142.02.ПЗ	Арк.
						58
Зм.	Арк	№ докум.	Підпис	Дата		

```

    }

    private void putInToSch(String[][] sch) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO schedule VALUE (?, ?, ?, ?)");
        for (String[] str : sch) {
            if(str!=null){
                pt.setString(1, str[0]);
                pt.setInt(2, Integer.parseInt(str[1]));
                pt.setInt(3, Integer.parseInt(str[2]));
                pt.setString(4, str[3]);
                pt.execute();
            }
        }
    }

    private void putInToPaid(String[][] paid) throws SQLException {
        pt = connection.prepareStatement("INSERT INTO paidof VALUE (?, ?, ?, ?)");
        for (String[] str : paid) {
            if(str!=null){
                pt.setInt(1, Integer.parseInt(str[0]));
                pt.setString(2, str[1]);
                pt.setInt(3, Integer.parseInt(str[2]));
                pt.setString(4, str[3]);
                pt.execute();
            }
        }
    }
}

```

Лістинг 2. Index1.html

```

<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <title>Bank</title>
    <link rel="stylesheet" type="text/css" href="style.css"/>
</head>
<body>
<div id = "main">
    <div id="head">
        <h1>Simple Bank</h1>
        <table>
            <tbody>
                <tr>
                    <td>
                        <a href="index.html">Главная </a>
                        <a href="page.html"> Страница </a>
                        <a href="feedback.html"> Обратная связь</a>
                    </td>
                </tr>
            </tbody>
        </table>
    </div>
    <div id="basic" class="box">
        <div>
            <h2>Информация</h2>
            <p>

```

```

        </p>
    </div>
</div>
<div id="tel">
    <p> Контакты: 456-234-567 |   Почта: yrewt@gmail.com</p>
</div>
</div>
</body>
</html>

```

Лістинг 3. Mains.java

```

package server;

import logic.Calc;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.IOException;

@WebServlet(name = "Mains", urlPatterns = {"*.html"})
public class Mains extends HttpServlet{
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

    }
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        double start = Double.parseDouble(request.getParameter("start"));
        double end = Double.parseDouble(request.getParameter("end"));
        double step = Double.parseDouble(request.getParameter("step"));
        Calc calculator = new Calc(start,end,step);
        request.setAttribute("calc", calculator.getPoints());
        request.getRequestDispatcher("/tab.jsp").forward(request,response);
    }
}

```

					ДР.122.4142.02.ПЗ	Арк.
						60
Зм.	Арк	№ докум.	Підпис	Дата		