

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами

Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ

к.т.н, доц. Михелєв І.Л.

" ____ " _____ 2021 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "магістр"

на тему: Дослідження моделей процесів управління кредитного відділу банку та
розробка інформаційної системи

Виконав: студент 6144 м групи

_____ ***Бабенко Я.А.***

(підпис)

Керівник роботи:

доцент кафедри ІУСТ, к.т.н.,

(посада, науковий ступінь вчене звання)

_____ ***Гайдаєнко О.В.***

(підпис)

м. Миколаїв – 2021 року

Анотація

Предметом роботи є розробка проекту інформаційної системи управління Кредитний відділ банку. Система розроблена для підвищення ефективності робота співробітників кредитного відділу.

Аналіз об'єктів та їх організаційної діяльності, структура для виявлення недоліків у системі управління кредитним відділом, вимоги до формування проекту інформаційної системи управління кредитною інформацією, аналіз підрозділів та існуючих рішень здійснюється в роботі. В міру необхідності розроблено концепцію та програмні рішення для розробки інформаційних систем. Розроблені програмні рішення включають загальносистемні рішення, рішення з інформації, програмне та апаратне забезпечення.

Робота викладена на 59 сторінках надрукованого тексту, містить 20 рисунків та список використаної літератури з 20 найменувань.

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		2

Annotation

The subject of the work is the development of the project of information management system for the Credit Department of the bank. The system is designed to increase the efficiency of the credit department employees' work.

The analysis of objects and their organizational activities, structure for identification of deficiencies in the system of management of the credit division, requirements to formation of the project of information system of management of credit information, analysis of sub-divisions and existing solutions is carried out in the work. To the extent necessary, developed the concept and software solutions for the development of information systems. The developed software solutions include general system solutions, information solutions, software and hardware.

The work is spread over 59 pages of printed text, contains 20 pictures, a list of used literature from 20 items.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		3

ЗМІСТ

ВСТУП	6
РОЗДІЛ І. ДОСЛІДЖЕННЯ МОДЕЛЕЙ УПРАВЛІННЯ КРЕДИТНОГО ВІДДІЛУ БАНКУ	7
1.1 Аналіз існуючих рішень	7
1.1.1 ERP система планування ресурсів підприємства	7
1.1.2 Система електронного документообігу (СЕД)	7
1.1.3 Система управління взаємовідносинами з клієнтами (CRM)	8
1.1.4 Порівняння існуючих рішень	8
1.2 Опис об'єкта дослідження та його організаційна структура	9
1.3 Аналіз предметної області процесів обробки інформації	12
РОЗДІЛ ІІ. РОЗРОБКА КОНЦЕПЦІЇ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ	14
2.1 Формування вимог до проєкту інформаційної системи	14
2.2 Цільове призначення та задача дослідження	16
2.3 Розробка функціональної моделі(блок-схеми) системи	17
2.4 Формування вхідної, довідкової та вихідної інформації бази даних системи.....	19
РОЗДІЛ ІІІ. РОЗРОБКА ТЕХНІЧНОГО ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ КРЕДИТНОГО ВІДДІЛУ БАНКУ	20
3.1 Загальносистемні рішення	20
3.1.1 Діаграма використання	20
3.1.2 Діаграма діяльності	25
3.2 Рішення з інформаційного забезпечення	28
3.2.1 Концептуальна та логічна модель бази даних.....	28
3.2.2 Аналіз нормалізації сутностей бази даних.....	30

3.3	Рішення з програмного забезпечення	31
3.3.1	Вибір інструментальних засобів.....	31
3.3.2	Структура програмного забезпечення.....	33
3.3.3	Фізична модель бази даних	33
3.3.4	Організаційна структура	34
3.3.5	Методи і засоби розробки програмного забезпечення.....	35
3.4	Рішення з технічного забезпечення	36
3.5	Рішення Розробка інтерфейсів.....	37
РОЗДІЛ IV . РЕАЛІЗАЦІЯ ПРОЄКТУ ІНФОРМАЦІОННОЇ СИСТЕМИ		
КРЕДИТНОГО ВІДДІЛУ БАНКУ		39
4.1	Розробка документації системи	39
4.2	Розгортання системи	39
РОЗДІЛ V. ОХОРОНА ПРАЦІ		40
5.1	Характеристика приміщення	40
5.2	Мікрокліматичні умови	41
5.3	Освітлення	41
5.4	Шум та вібрація.....	41
5.5	Випромінювання	42
ВИСНОВОК		43
Список використаних джерел.....		44
ДОДАТОК А.....		47
ДОДАТОК Б		50
ДОДАТОК В.....		51
ДОДАТОК Г		51

ВСТУП

Сьогодні існує багато банків. Комуś вдалося піднятися на високий рівень, а іншим – навпаки і принесли лише втрати. Тут виникає питання: як вирішити цю проблему?

З часом все більшого значення набувають інформаційні системи, які значно покращили та спростували роботу різних компаній. Однією з головних складових сучасного світу є інформація. Процес обробки інформації дуже тривалий і ресурсномісткий процес. Цей процес потрібно прискорити, щоб заощадити час і ресурси.

Щодня банки надають послуги багатьом клієнтам, і не кожен може впоратися з таким потоком інформації.

Одним із шляхів вирішення цієї проблеми є впровадження нової технології обробки інформації. Впровадження нових технологій можна вирішити за допомогою нових інформаційних систем. Тому в цій статті буде розроблено проект інформаційної системи кредитного відділу банку. Основним завданням проекту є розробка інформаційної системи, спрямованої 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;

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		6

РОЗДІЛ І. ДОСЛІДЖЕННЯ МОДЕЛЕЙ УПРАВЛІННЯ КРЕДИТНОГО ВІДДІЛУ БАНКУ

1.1 Аналіз існуючих рішень

Існують різноманітні системи управління банків, якими зараз користуються організації. Усі вони різні але спрямовані для покращення та прискорення робочого процесу. Ці системи підходять для різних стуртур управління. Підприємства самостійно встановлюють більш прийнятну структуру управління. Вибір структури визначається спеціалізацією кредитних організацій стратегічними цілями та планами.

1.1.1 ERP система планування ресурсів підприємства

Ця система призначена для прискорення та автоматизації бізнес процесів. Складається з безліч модулів, тому призначена для обробки великого обсягу важливої інформації. ERP система має єдиний стандарт управління та технології, що допомагає швидко виконувати роботу. Але не дивлячись на велику перевагу також має вагомий недолік. Для бездоганної роботи такої системи необхідно багато грошових та трудових витрат, протягом усього часу користування для підтримки роботи.

1.1.2 Система електронного документообігу (СЕД)

Розрахована на велику кількість користувачів система включає в себе електронний документообіг спрямований на позбавлення паперових ділових документів. Великою перевагою є можливість інтеграції з різноманітними системами для швидкого заповнювання документів. Також, зберігається історія редагування та обробки заяв, дуже швидко та просто генеруються звіти. Але, нажаль, повністю позбавитись паперів не вдалось, та недоліком такої системи є потреба збереженню паперових копій для деяких операцій.

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		7

1.1.3 Система управління взаємовідносинами з клієнтами (CRM)

Прикладне програмне забезпечення спрямоване на спрощення роботи з клієнтами, та покращення роботи з ними завдяки зберіганню історії взаємовідносин та повної інформації про клієнтів. Перевагами такої системи є єдина база клієнтів з повною інформацією та історією. Наступна перевага співробітник може контролювати поетапно кожну угоду та можливість створення статичних звітів на основі зібраної інформації.

1.1.4 Порівняння існуючих рішень

Порівняння існуючих систем представлено у Таблиця 1.1

Вимоги	ERP	СЕД	CRM
Створення місячних звітів	-	+	+
Скорочення паперового документообігу	-	+	+
Автоматизація внутрішніх операцій	+	+	+
Автоматизація розподілу праці	-	-	-
Аналіз стану підприємства	+	-	-

Таблиця 1.1 - Порівняння існуючих

Таким чином порівнюючи усі представлені системи є, які не покривають базові потреби підприємства в більшій або меншій мірі. Також є системи які дуже дорогі для обслуговування.

Тому є доцільним розглянути варіант створення нової системи управління кредитним відділом банку.

1.2 Опис об'єкта дослідження та його організаційна структура

Розглянемо структуру SimpleBank, що є предметом даного дослідження. Банк є фінансовою установою, яка займається різними видами бізнесу з валютами та цінними паперами, а також надає фінансові послуги урядам, юридичним і фізичним особам. Банк є комерційною юридичною особою. Створений з метою отримання прибутку, уповноважений на здійснення банківської діяльності. Операція з виключним правом залучення законних коштів згідно із законом. Фізичні особи з метою подальшого здійснення розрахунків від свого імені, а також для відкриття та ведення банківських рахунків юридичних та фізичних осіб. (Вікіпедія, -)[8]

Функція банку – безпечне зберігання коштів клієнтів. Оскільки в банку багато клієнтів, які вносять в нього гроші, банк може переказувати гроші з одного в інший шляхом зміни запису на банківському рахунку (безготівковий розрахунок). Завдяки системі агентських рахунків безготівкові розрахунки також можуть здійснюватися між клієнтами різних банків. Банк видає кредити. Це фактично створює додаткову грошову масу. Кредитні або кредитні відносини - соціальні відносини руху цінностей, що виникають між суб'єктами економічних відносин. Кредитні відносини можуть виражатися в різних формах кредиту (комерційний кредит, банківський кредит), позик тощо. (Вікіпедія, -) [8] SimpleBank використовує лінійну організаційну структуру, яка являє собою систему, в якій кожен підлеглий належить лише одному керівнику. Кожен підрозділ виконує повний комплекс робіт. Схема організаційної структури представлена на Рисунку 1.1.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		9

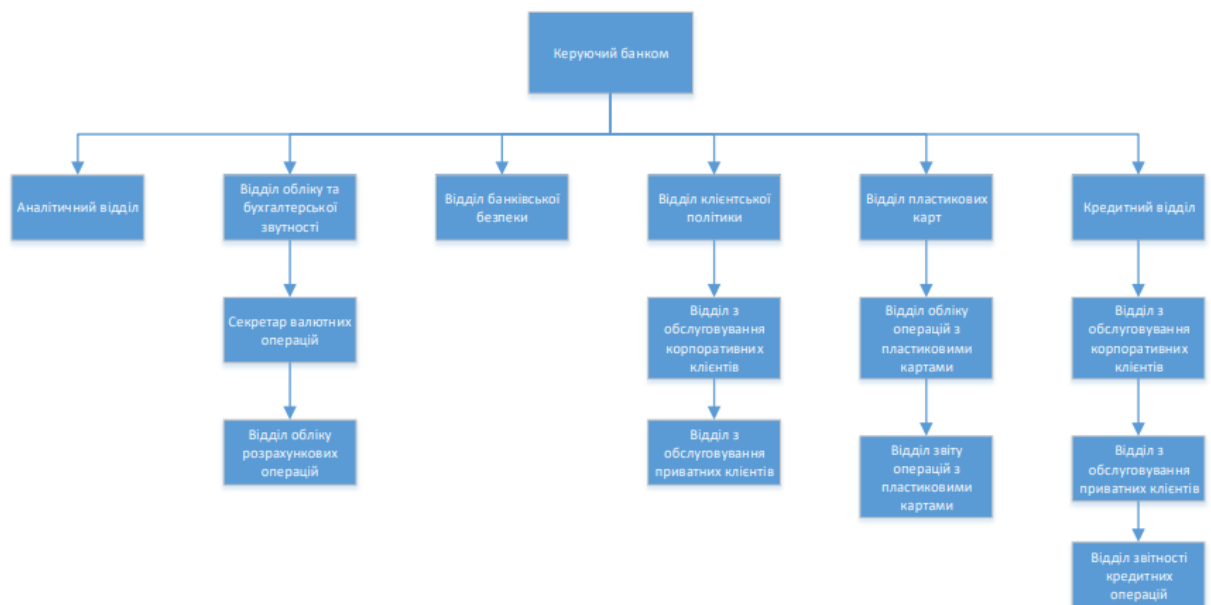


Рисунок 1.1 – Схема організаційної структури «SimpleBank».

Кредитний менеджмент є найважливішим і найбільшим відділом банку. Тому що його основною функцією та основним джерелом доходу є постачання різноманітних позики. (О.В., 2016) [9]

Під появою кредиту як особливої форми вартісного відношення розуміється вартість, випущена суб'єктом господарювання, який протягом певного часу не вступив у новий цикл відтворення. Через позику він ніколи не передає від суб'єкта (кредитора), який не використовує його, іншому суб'єкту (позичальнику), який потребує додаткових коштів.(10, 2013-2014)[10]

Кредит відіграє важливу роль у саморегулюванні обсягу коштів, необхідних для здійснення господарської діяльності. Завдяки цьому кредиту підприємство може отримати кошти, необхідні для нормальної діяльності в будь-який момент. Так само роль кредиту дуже важлива для поповнення оборотних коштів. Потреби кожного підприємства нестабільні і змінюються залежно від умов праці: ринку, природи, клімату, політики тощо. Великий вплив на відтворення основних фондів має кредит. Використовуючи кредит, компанія може покращити та збільшити виробництво швидше, ніж без кредиту.(11, 2013-2014) [11]

Кредит відіграє важливу роль у створенні ефективного механізму фінансування державних витрат, а також в регулюванні ліквідності банківської системи. (11, 2013-2014)[11]

Споживчий кредит - кредит, що видається простим громадянам (домашнім господарствам) для здобуття предметів споживання. Кредити видають не тільки для покупки товарів тривалого користування (квартири, меблі, автомобілі). Але і для різноманітних покупок (мобільні телефони, побутова техніка, продукти харчування). (12Вікіпедія, 2011) [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).

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

- мета кредиту.

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		11

- 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.

Співробітники відділу звітності контролюють платежі, планують виплати та формують щомісячні звіти для керівництва. У кредитному відділі також є менеджер, який відповідає за нагляд за розподілом клієнтів, формування звітів і платежів, а також роботу різних відділів кредитного відділу банку.

1.3 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. У сучасних умовах основним технічним засобом обробки інформації є персональний комп'ютер. Більшість сучасних інформаційних систем перетворюють не інформацію, а дані. Тому їх часто називають системами обробки даних. (Маслов, 2008) [5]

Після аналізу інформаційної системи, яку використовує SimpleBank, було виявлено, що система є неповною та недосконалою. Має багато недоліків, не пов'язує клієнтів із співробітниками та кредитами. Ця система відіграє дуже важливу роль, оскільки дозволяє відстежувати всю поведінку співробітників і клієнтів, а також поведінку клієнтів.

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

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		12

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 всі виявлені недоліки ефективність роботи відділу не дуже висока. З метою підвищення ефективності роботи відділу необхідно замінити інформаційну систему на нову більш функціональну інформаційну систему, яка допоможе зробити роботу кожного співробітника відділу швидшою та зручнішою. Система допоможе швидко відстежувати всі транзакції та будувати роботу всередині підприємства. Це зможе позитивно вплинути на подальшу роботу всього банку та збільшити клієнтську базу.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		13

РОЗДІЛ II. РОЗРОБКА КОНЦЕПЦІЇ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Формування вимог до проєкту інформаційної системи

Відповідно до недоліків системи, яку використовує досліджуваний об'єкт, можна сформулювати вимоги, яким повинна відповідати нова система для підвищення продуктивності та збільшення клієнтської бази.

Спочатку система має бути зрозумілою та легкою для розуміння, щоб кожен співробітник міг нею користуватися. Найкраще рішення – мати веб-сайт, до якого можна отримати доступ з будь-якого місця через Інтернет та комп'ютер. Завдяки простому інтерфейсу немає необхідності витратити багато часу на адаптацію до нової системи, навіть нові співробітники можуть легко завершити роботу.

Перша проблема полягає в тому, що працівники відділу повинні окремо виготовляти та подавати звіти про нових клієнтів та всі кредити та операції. Це незручно і вимагає багато додаткового часу. Тому наступна вимога — генерувати щомісячні звіти за певними стандартами.

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 працівника. Після розгляду працівник банку надсилає звіт керівнику або керівнику агентства. Крім того, враховуючи графік платежів, кожен співробітник формує окремий звіт про оплату кожного клієнта.

Наступна проблема полягає в тому, що менеджеру необхідно вручну запланувати платіж з урахуванням дати позики, суми та терміну позики. Ще одна вимога – розрахувати щомісячну суму погашення та встановити графік

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		14

платежів по кожній позиції відповідно до договору. Це дозволить швидко надати клієнтам графік платежів, згідно з яким будуть здійснюватися платежі.

Важливим є те, що працівники зобов'язані контролювати навантаження (кількість клієнтів). Це дуже незручно і забирає час персоналу. Для вирішення цієї проблеми керівникам необхідно використовувати нову інформаційну систему для призначення нових клієнтів відділам і співробітникам.

Дуже важливо, чи реєстрація клієнтів та кредитів проходить швидко та зручно, адже це вплине на якість та швидкість роботи.

Пошук з новими функціями стане легшим, а співробітники зможуть легко знаходити необхідну інформацію.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		15

2.2 Цільове призначення та задача дослідження

Метою дослідження є підвищення ефективності роботи кредитного відділу банку та збільшення клієнтської бази шляхом створення проекту сучасної інформаційної системи, що керує кредитним відділом банку.

Основним завданням дослідження є розробка концепції проекту інформаційної системи. Для реалізації цієї концепції необхідно провести роботу яка включає 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 відповідних методів, щоб відповідати вимогам інформаційної системи.
- На основі розробленої системної концепції зроблено висновок, наскільки нова система буде добре в порівнянні з існуючою.

2.3 Розробка функціональної моделі(блок-схеми) системи

Проект розвитку системи управління кредитним відділом включає такі процеси:

1. Зареєструватися в системі співробітників. Реєстрація співробітників. У процесі взаємодії з керівником менеджер надає в систему персональні дані співробітника.
2. Зареєструватися в системі клієнта. Реєстрація клієнта. До процесу взаємодії має відношення менеджер, який надає в систему персональні дані клієнтів.
3. Створення кредитної системи. Введіть всі дані за договором. Під час взаємодії співробітник, який надає в систему дані договору, підписує договір із замовником.
4. Формування графіка платежів. Менеджер взаємодіє з процесом, враховуючи всі дані кредитного договору.
5. Формування списку погашення. Співробітник відділу взаємодіє з процесом і враховує всі платежі по кредиту.
6. Формування всіх платежів, які здійснюються клієнтами. Під час взаємодії співробітники, які надають платіжні дані в систему та отримують звіти.
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 2.1.

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		17

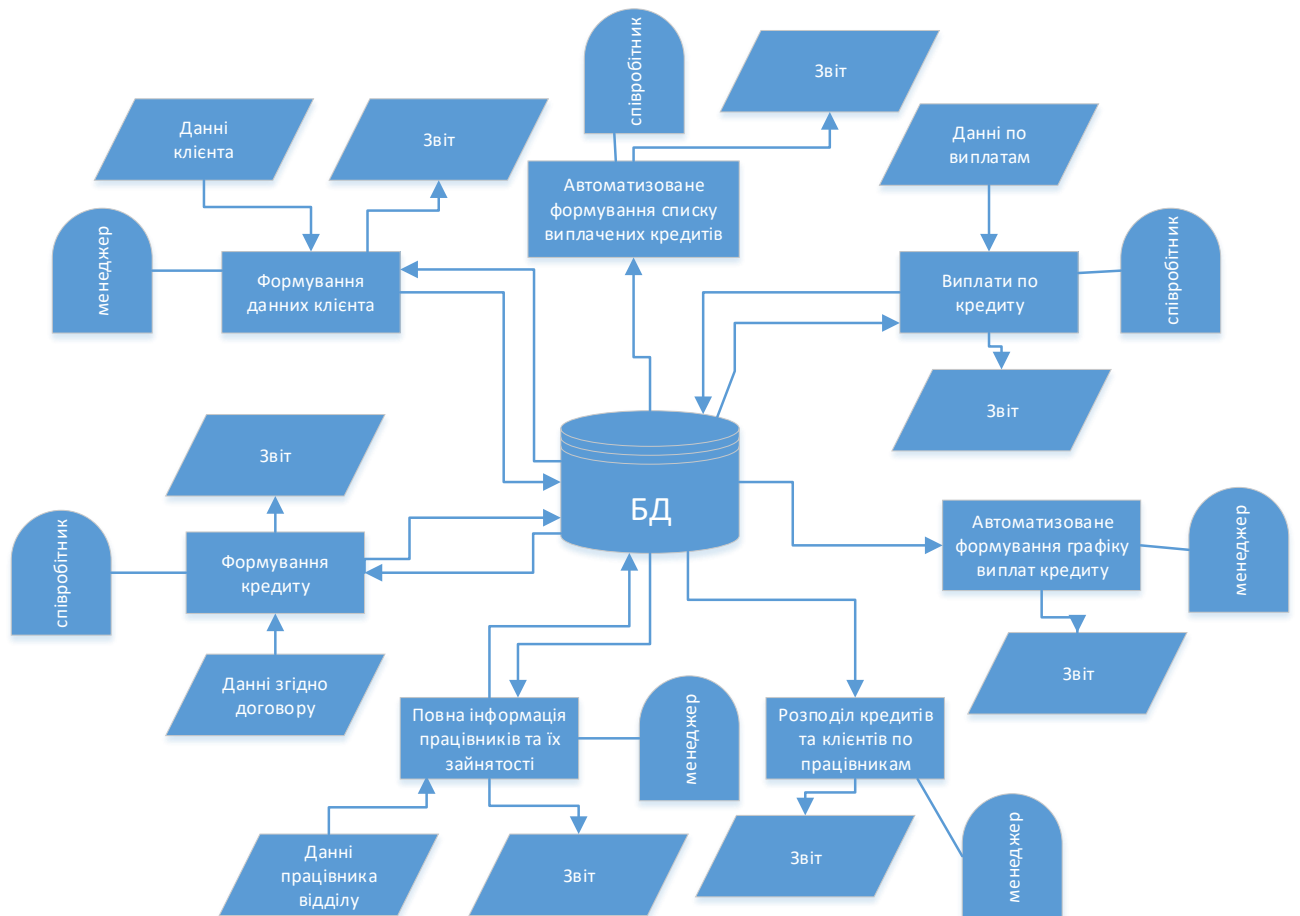


Рисунок 2.1 – Функціональна модель(блок-схема) системи.

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

2.4 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 id 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].

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		19

РОЗДІЛ III. РОЗРОБКА ТЕХНІЧНОГО ПРОЄКТУ ІНФОРМАЦІОННОЇ СИСТЕМИ КРЕДИТНОГО ВІДДІЛУ БАНКУ

3.1 Загальносистемні рішення

3.1.1. 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. (14Вікіпедія, 2007-2020) [14]

Суть діаграми полягає в наступному: Спроектowana система представлена як група сутностей або учасників, які взаємодіють із системою за допомогою так званого використання. Варіанти використання використовуються для опису послуг, які система надає учасникам. Іншими словами, кожен варіант використання визначає набір дій, які виконує система під час розмови з учасниками. При цьому не сказано, як буде реалізовуватися взаємодія між учасниками та системою. (Вендров) [6, 14]

Для визначення сутності та атрибутів ми розробимо схему використання на основі всіх взаємодій, процесів та учасників концепції проекту інформаційної системи.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		20

Діаграма використання процесу “Реєстрація співробітника та взаємодія з ним” представлена на Рисунку 3.1

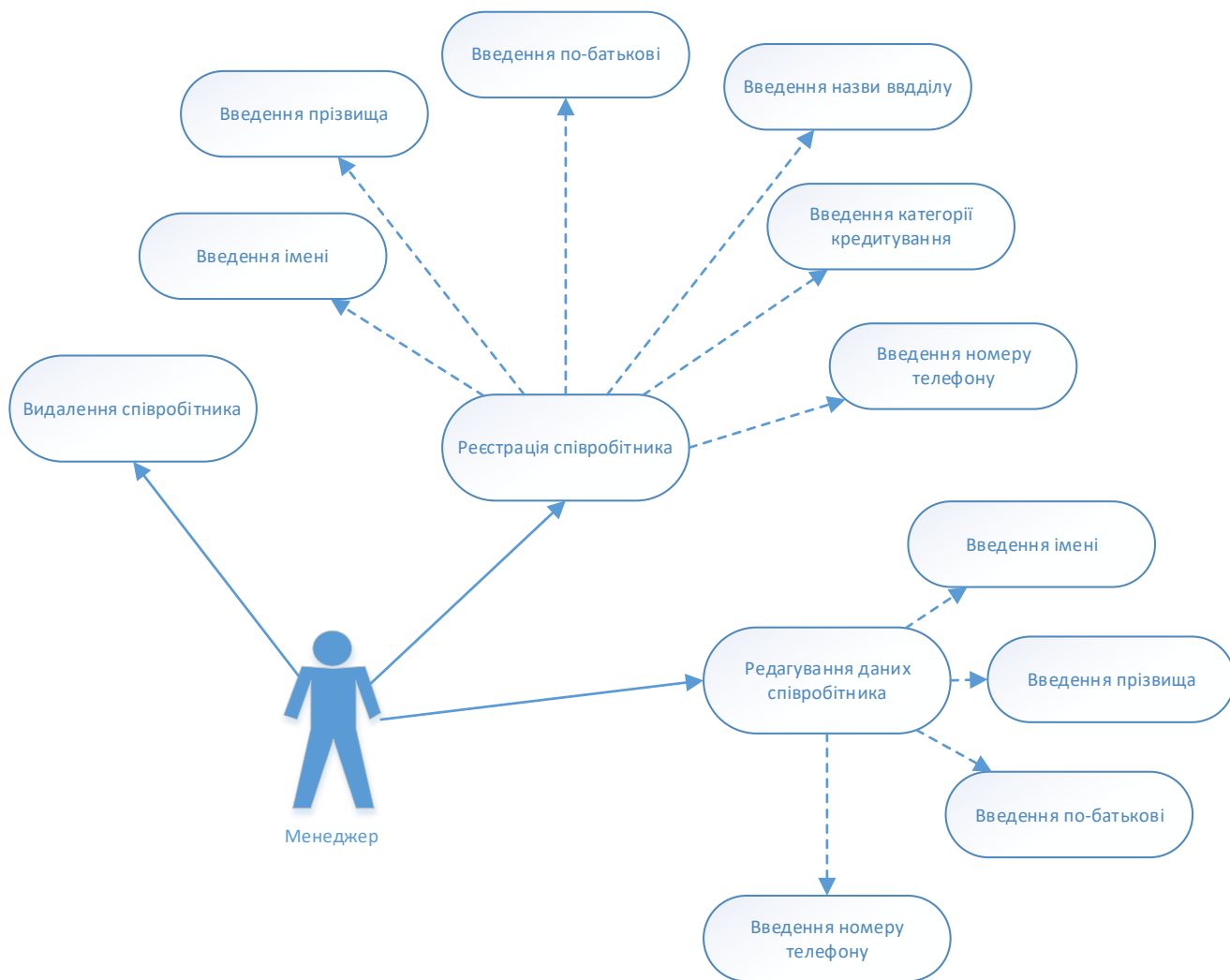


Рисунок 3.1 - Діаграма використання процесу “Реєстрація співробітника та взаємодія з ним”

Діаграма використання процесу “Реєстрація клієнта та взаємодія з ним” представлена на Рисунку 3.2

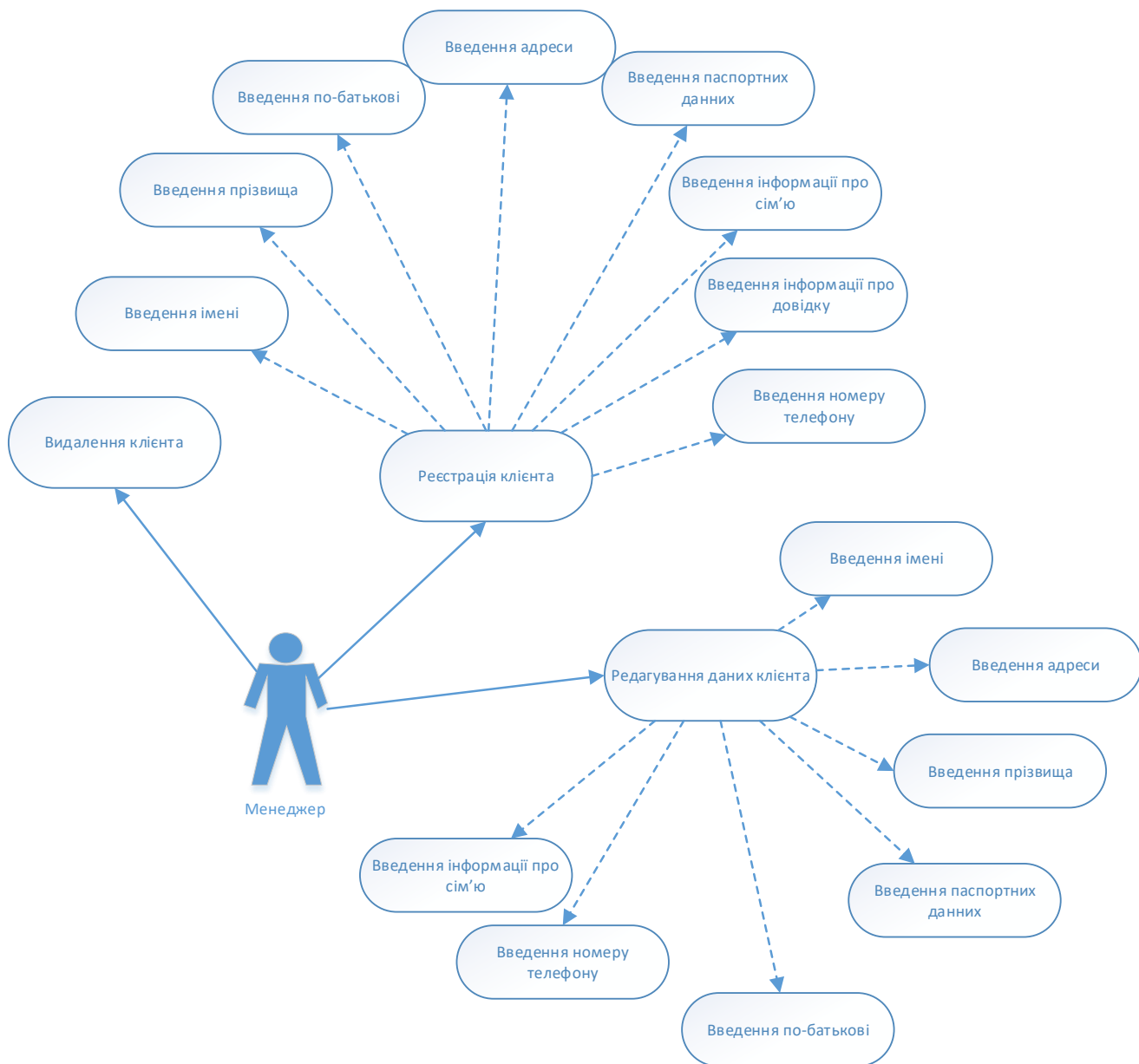


Рисунок 3.2 - Діаграма використання процесу “Реєстрація клієнта та взаємодія з ним”

Діаграма використання процесу “Створення кредиту та взаємодія з ним” представлена на Рисунку 3.3

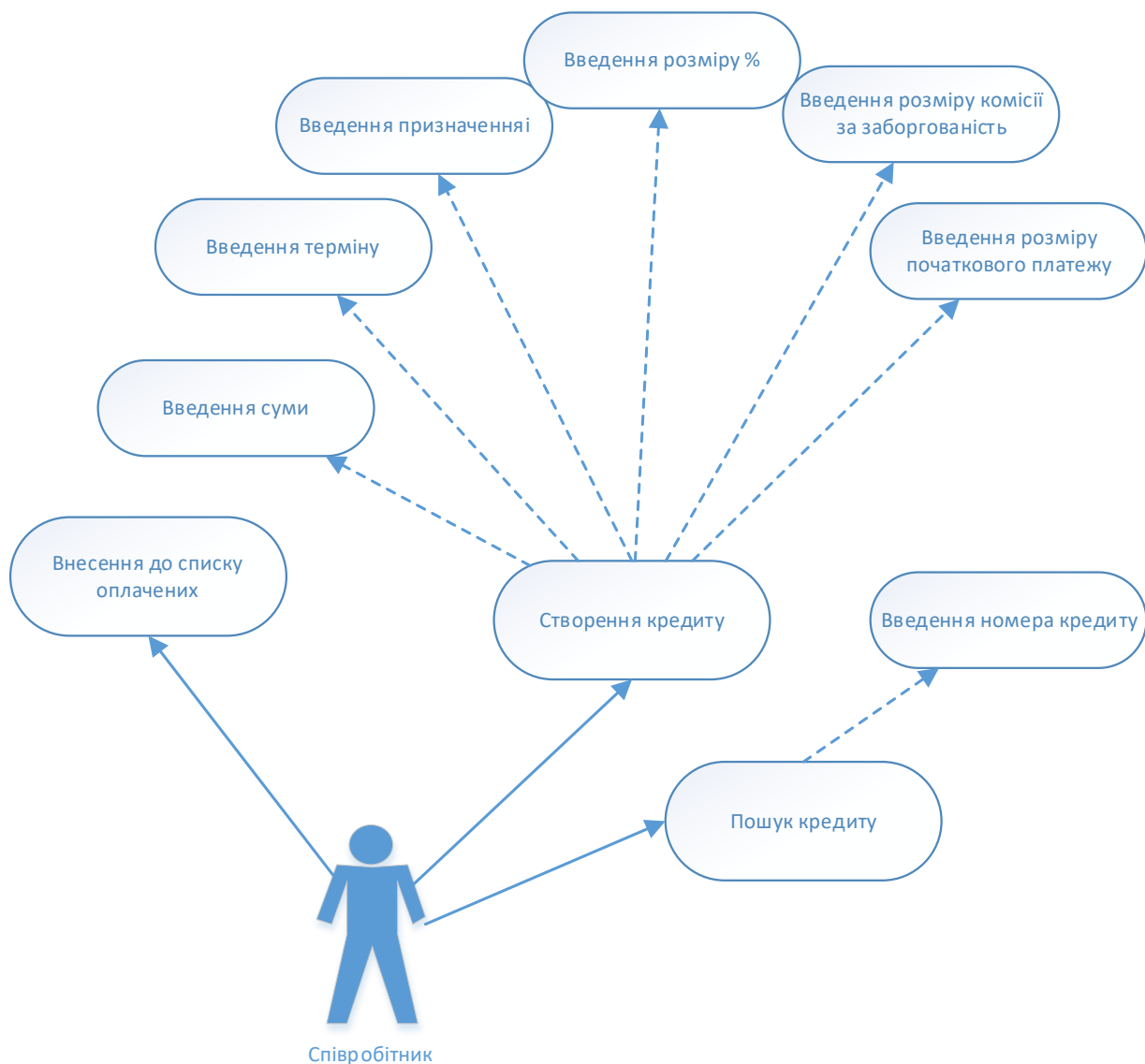


Рисунок 3.3 - Діаграма використання процесу “ Створення кредиту та взаємодія з ним ”

Діаграма використання процесу “Формування графіків” представлена на Рисунок 3.4

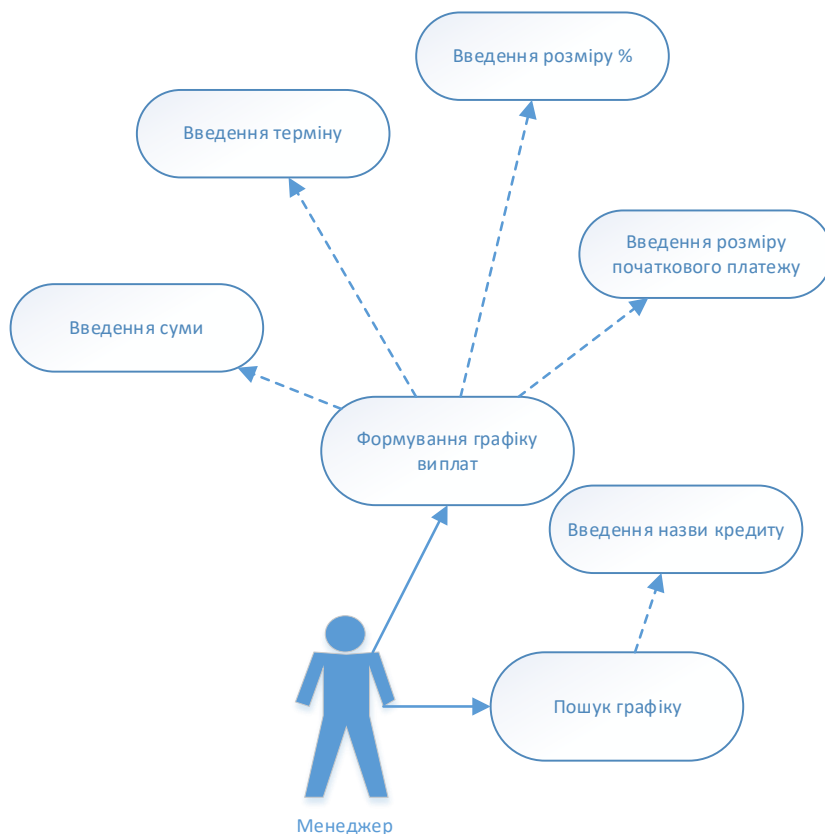


Рисунок 3.4 - Діаграма використання процесу “Формування графіку та взаємодія з ним”

Діаграма використання процесу “Розподіл клієнтів” представлена на Рисунок 3.5

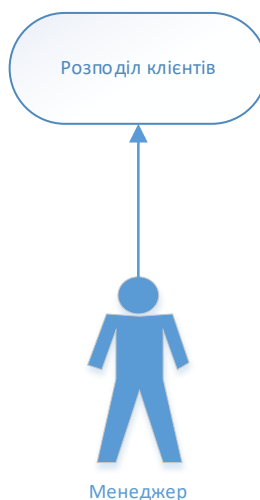


Рисунок 3.5 – Діаграма використання процесу “Розподіл клієнтів”

Діаграма використання процесу “Формування звітів” представлена на Рисунок 3.6

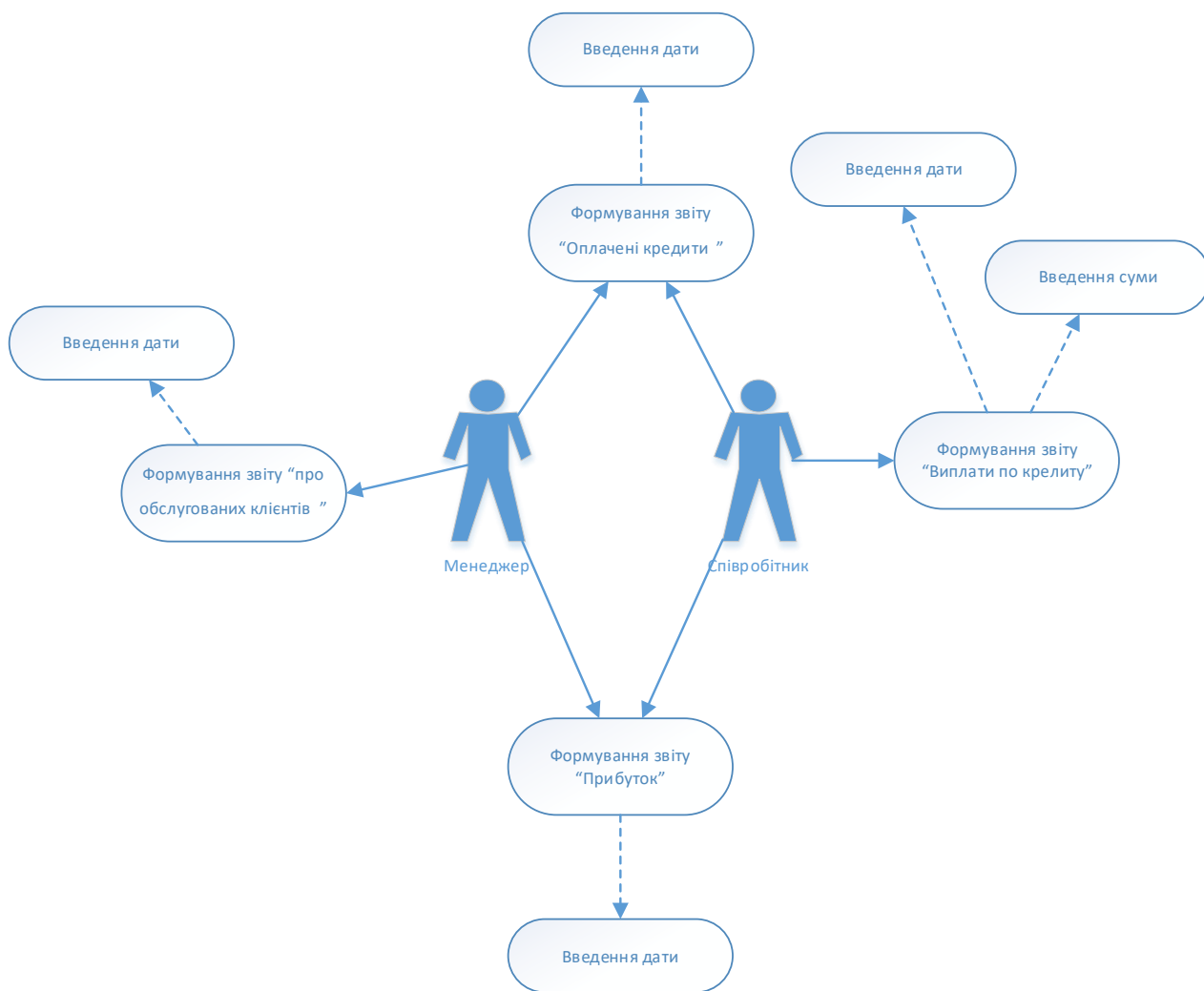


Рисунок 3.6. - Діаграма використання процесу “Формування звітів”

3.1.2. 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]

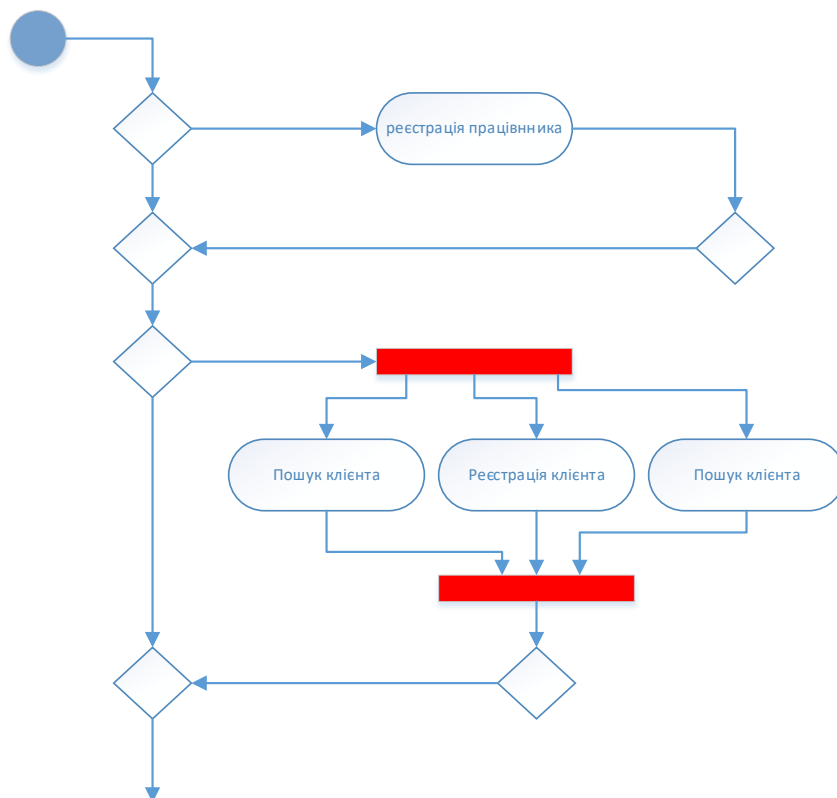
Дія є основною одиницею визначення поведінки в специфікації. Ця дія отримує набір вхідних сигналів і перетворює їх у набір вихідних сигналів. Один або обидва ці набори можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Аналогічно, виконання діяльності – це виконання окремої діяльності, буквально, включаючи виконання дій, що містяться в діяльності. Кожну дію в дії можна виконати один, двічі або кілька разів у дії.

Принаймні, дії мають отримати, трансформувати та перевірити дані, а деякі дії можуть вимагати певного ступеня узгодженості. Специфікації діяльності (на більш високому рівні сумісності) можуть дозволити виконувати декілька (логічних) потоків, а також існування механізмів синхронізації, щоб гарантувати, що операції виконуються в правильному порядку. [15]

Діаграма діяльності складається з обмеженої кількості графіків, з'єднаних стрілками. 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]



1

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		26

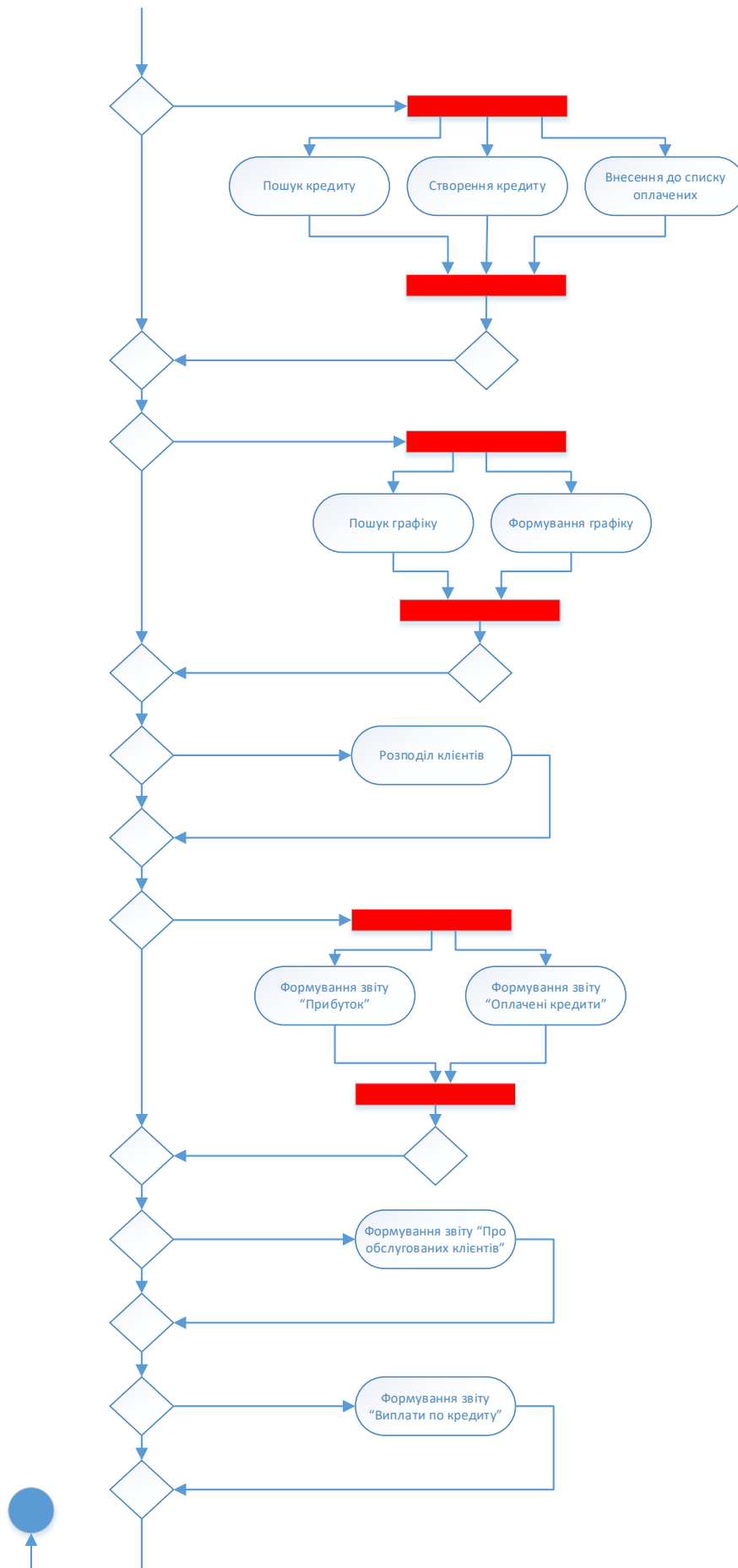


Рисунок 3.7 - Діаграма діяльності

3.2Рішення з інформаційного забезпечення

3.2.1 Концептуальна та логічна модель бази 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 3.8

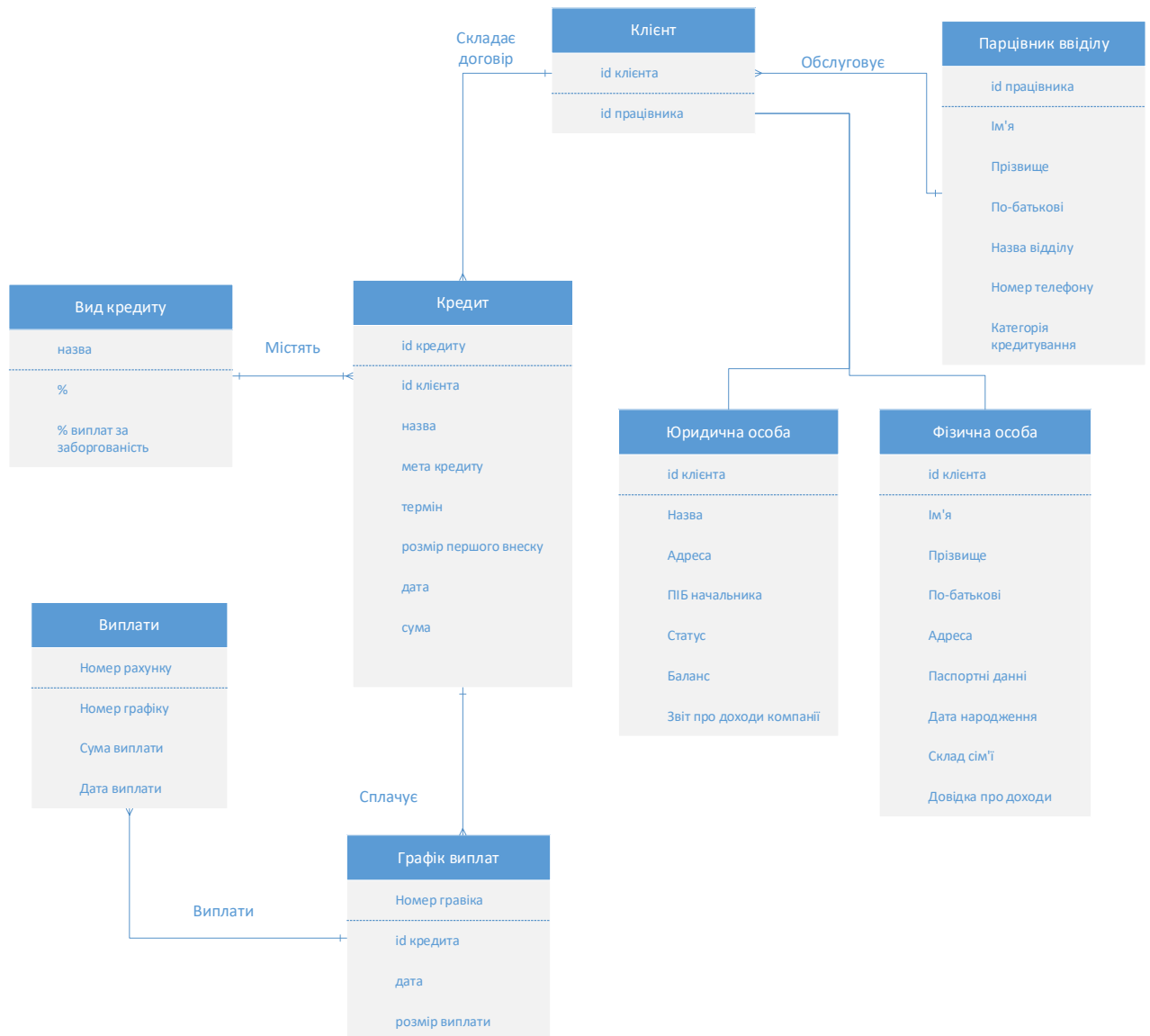


Рисунок 3.8 - Концептуальна модель бази даних

На основі концептуальної моделі було створено логічну модель даних. Модель представлена на Рисунку 3.9

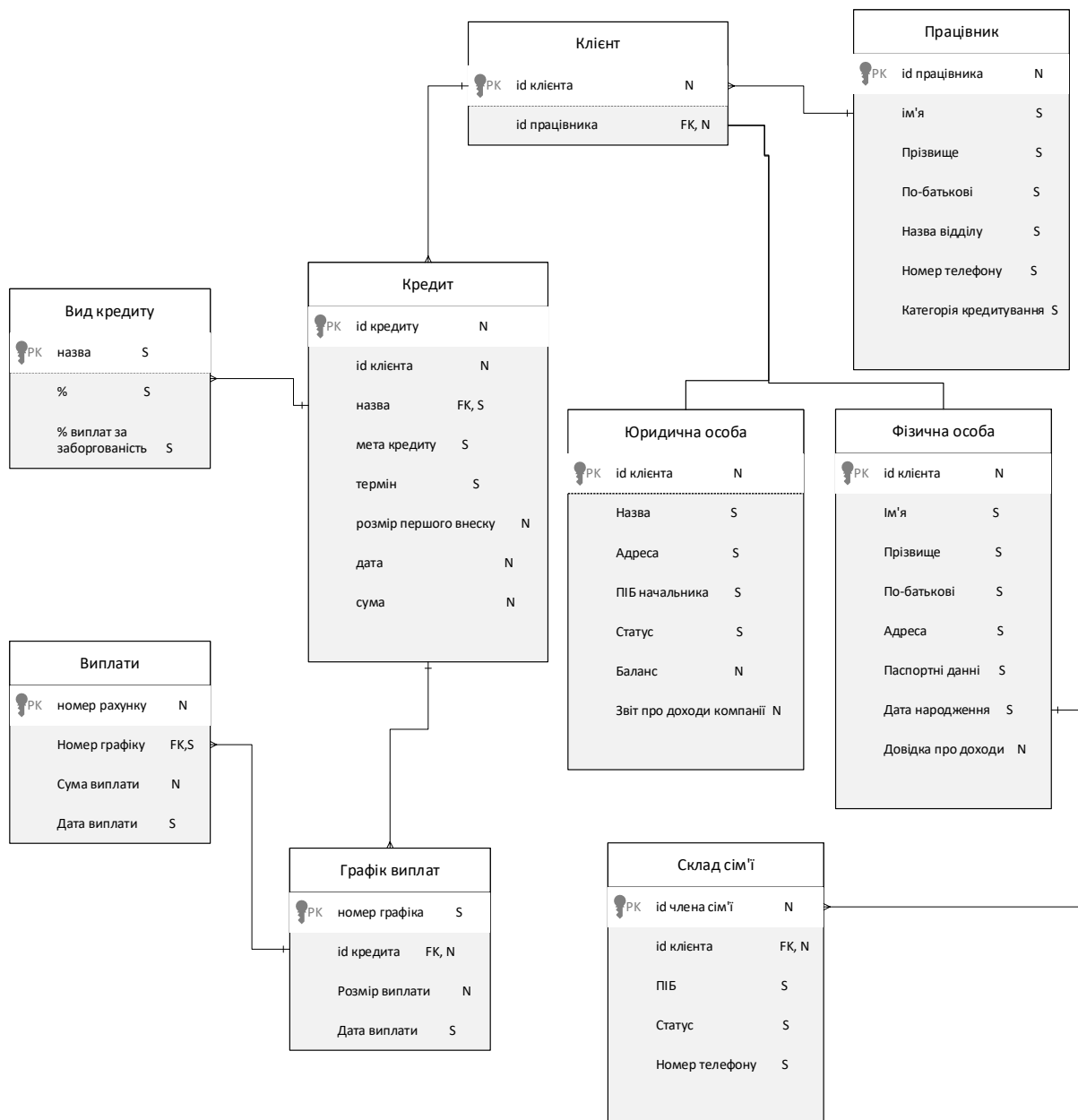


Рисунок 3.9 - Логічна модель бази даних

Згідно моделі, яка була представлена на рисунку 3.9 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.

3.2.2 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 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.

3.3 Рішення з програмного забезпечення

3.3.1 Вибір інструментальних засобів

Операційна система – це базовий комплекс програм, який виконує управління апаратною складовою комп'ютера або віртуальної машини; забезпечує взаємодію з користувачем та управління обчислювальним процесом. (Wiki, 2011) [13]

Вибір операційної системи є важливою частиною розробки концепції, оскільки операційна система залежить від простоти використання, функціональних вимог та якості розробки.

Сьогодні у світі в основному використовуються три операційні системи: Windows, Linux і Mac OS. Кожна система має свої переваги і недоліки.

Серія комерційних операційних систем Windows-Microsoft, зосереджена на використанні графічних інтерфейсів користувача.

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:

- 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.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		31

- Ціна. Після кожного великого релізу доведеться купувати ОС заново. Винятком стала тільки 10 версія з тимчасовим безкоштовним оновленням.

Mac OS — це сімейство операційних систем, вироблених 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.
- Віруси. Для цієї системи значно мала кількість загроз.

Недоліки:

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		32

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

Виходячи з функції організації, необхідно згадати, яке програмне забезпечення має відношення до роботи всієї організації. В основному програмне забезпечення Microsoft. Безпека дуже важлива для таких організацій, і всі програми повинні бути ліцензовані для використання. Тому більш вигідною та зручнішою операційною системою є Windows.

Тому що впроваджена інформаційна система має бути простою, зручною та зрозумілою. Враховуючи всі переваги, для реалізації інформаційної системи було обрано операційну систему Windows.

3.3.2 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

Діаграма основних класів системи представлена на Рисунку 3.10.

Рисунок 3.10 - Діаграма класів

3.3.3 Фізична модель бази даних

Фізична модель даних — це подання дизайну даних як призначеного для реалізації або реалізованого у системі управління базами даних. Завершена фізична модель даних включатиме всі артефакти бази даних, необхідні для досягнення мети продуктивності або для створення відношень між таблицями, як-от індексів, визначень обмежень, зв'язаних і секціонованих таблиць. (Маслов, 2008) [5]

Для роботи с базою даних було використано MySQL. Модель представлена на Рисунку 3.11

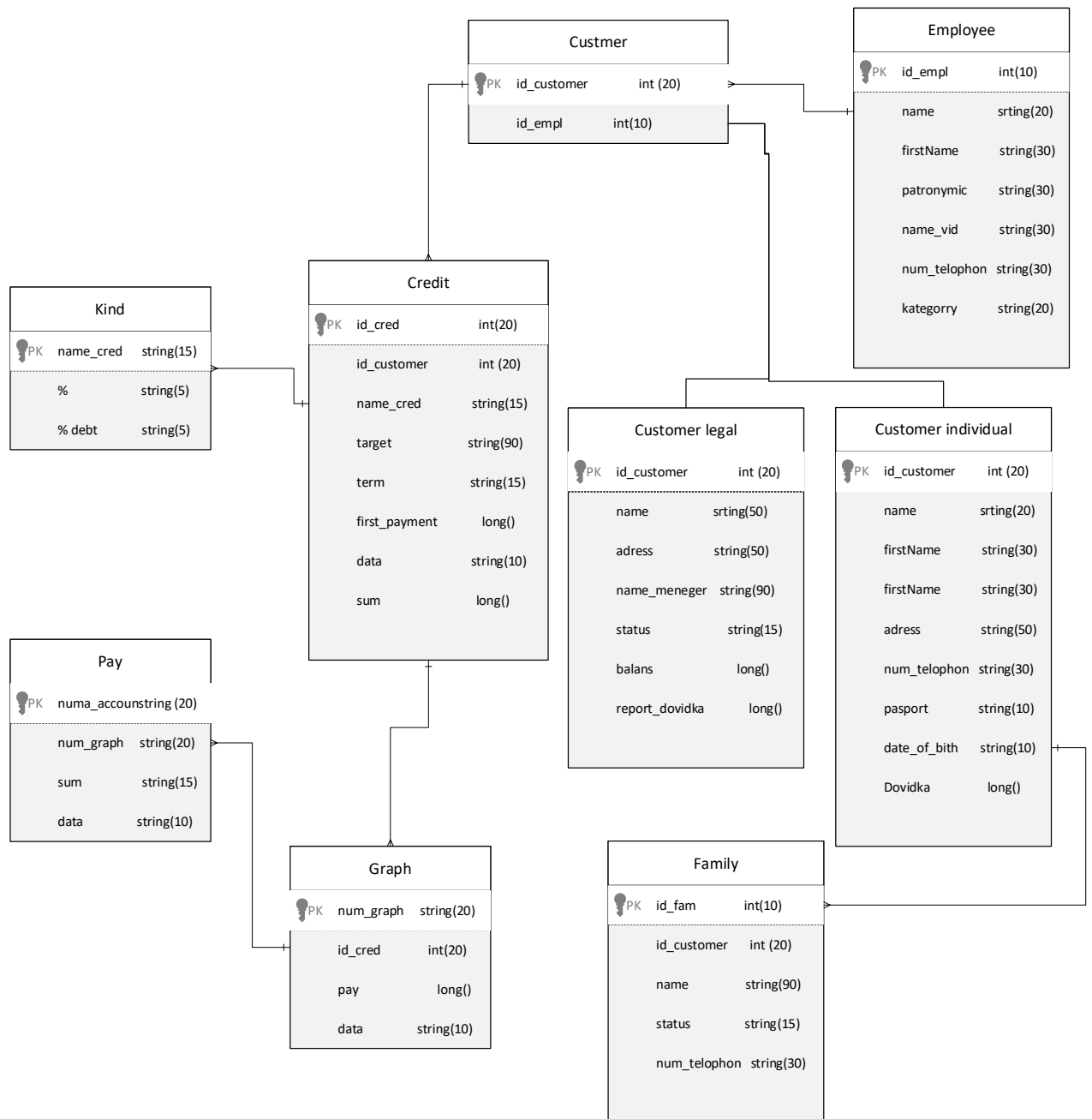


Рисунок 3.11 - Фізична модель бази даних

3.3.4 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 користувачів, які можуть використовувати систему одночасно. Необхідно правильно вибрати структуру, яка містить ідеальний інструмент.

Архітектура клієнт-сервер — це зручна архітектура для інформаційних систем, яка може задовольнити будь-які потреби. Ця архітектура є основною концепцією створення розподілених мережових додатків, що передбачають обмін даними та взаємодію. Вона має такі основні компоненти: набір серверів, які надають доступ до їх інформації або послуг; набір клієнтів, які користуються послугами, наданими сервером; і мережу, яка забезпечує взаємодію між клієнтом і сервером.

Ця архітектура вважається зручною, оскільки для взаємодії потрібні сервер і клієнт, які запитують сервер. Вам не потрібне додаткове програмне забезпечення для архітектури, тому що достатньо мати браузер, який може отримати доступ до сервера і виступати в ролі 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.

3.3.5 Методи і засоби розробки програмного забезпечення

Будь-яка інформаційна система має відповідати обговореним потребам. При розробці проєкту був використан об'єктно-орієнтований підхід, який має такі переваги:

- Класи конструюються з компонентів, що мають прості інструменти.
- Створюється певна сукупність з даних та операцій, усе згруповано та розташовано по певним місцям

- Локалізація коду і даних покращує наочність і зручність супроводу програмного забезпечення.
- від зайвого доступу до даних захищає Інкапсуляція інформації.
- Код без дублювання. Дозволяє зробити структуру коду більш унікальною

Мовою програмування обрали Java. Java - це багатоплатформна мова програмування. Ця мова використовує досить об'ємні бібліотеки класів, вони значно спрощують розробку, надаючи в розпорядження сильні засоби для вирішення різноманітних задач.

Програмним інструментом для реалізації системи стала IntelliJ IDEA. Саме цей інструмент було обрано, тому що IntelliJ IDEA є зручним та має розширений функціонал: лаконічні підказки, тому розробка ведеться швидше, можливість встановлювати різноманітні пакети, які покращують та спростовують розробку та інше.

Клієнтський інтерфейс був створений за допомогою HTML, CSS. HTML є мовою розмітки сторінки та використовується у багатьох веб-сайтах. Інтерфейс зручний для користувача. Швидко та легко є можливість зробити веб сторінку привабливішою, завдяки CSS, який надає стилі, кольори та відображення інформації. Ці мови дають веб-сторінкам стати не тільки цікавими, а ще й привабливими

3.4Рішення 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
- Обсяг дискового простору:16 ГБ для 32-розрядної ОС або 20 ГБ для 64-розрядної ОС

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		36

- 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.

3.5 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 3.12.

Рисунок 3.12 - Схематично сторінка реєстрації

Переглянути інформацію про клієнта або співробітника. Схематично сторінка вибору співпрацівника представлена на Рисунку 3.13

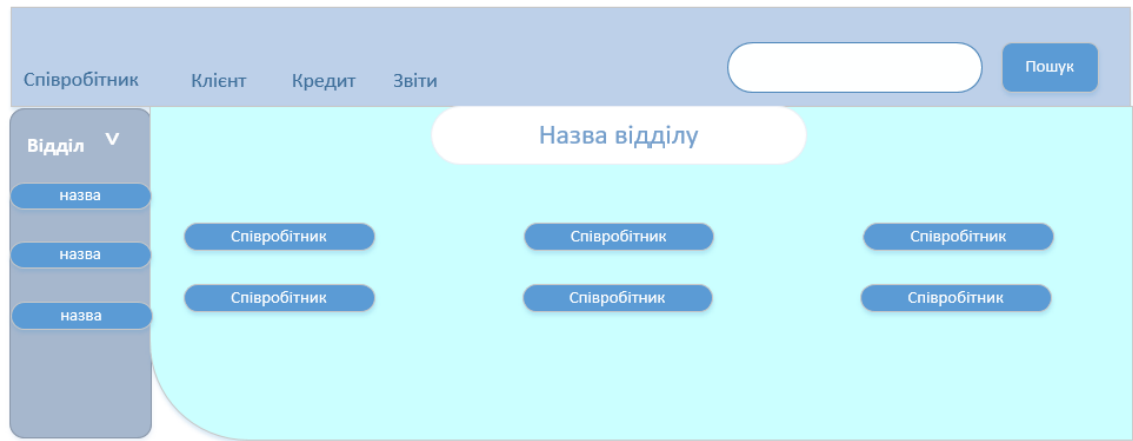


Рисунок 3.13 - Схематично сторінка вибору співробітника

Переглянути інформацію про кредит графік виплат та самі виплати. Схематично сторінка кредиту представлена на Рисунку 3.14

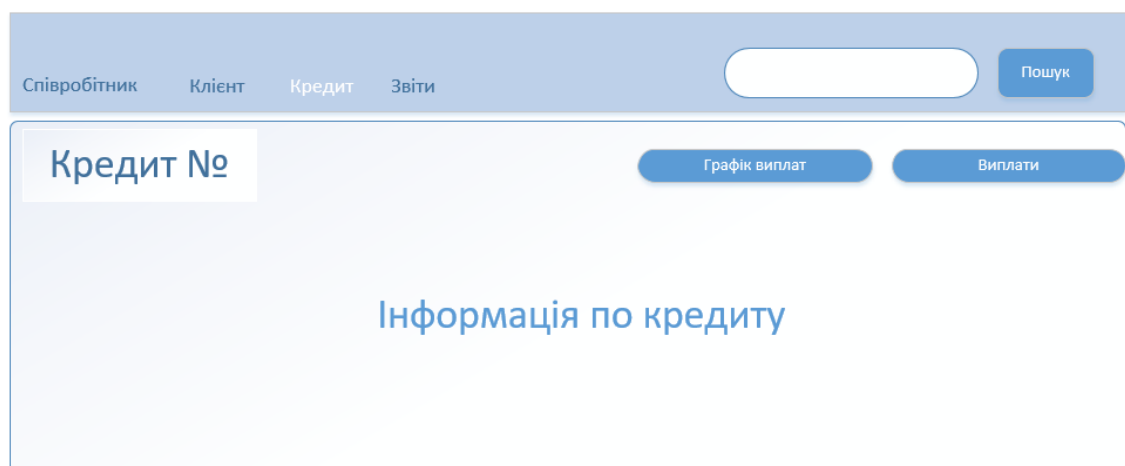


Рисунок 3.14 - Схематично сторінка кредиту

РОЗДІЛ IV. РОЗГОРТАННЯ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ КРЕДИТНОГО ВІДДІЛУ БАНКУ

4.1 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 програми інструменти, які були використані в процесі реалізації представлені у Додатку Г.

4.2 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.

РОЗДІЛ V. 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. [20]

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.

5.1 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. 5.1. 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	225м ³

5.2 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 5.2

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 5.2 – Показники мікроклімату

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

5.3 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.

5.4 Шум та вібрація

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

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.

5.5 Випромінювання

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

					МП.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		42

ВИСНОВОК

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 покращає.

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		43

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. “Моделювання процесів та систем управління”. Навчальний посібник. – 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>
10. Stud me [Електронний ресурс]: <http://studme.com.ua/> Режим доступу: http://studme.com.ua/1347071110546/bankovskoe_delo/kredit_kreditnaya_sistema.tema.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.
18. Санітарні норми мікроклімату виробничих приміщень : ДСН 3.3.6.042-99. – К., 2000.- С.16

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		45

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).
20. Законодавство України [Електронний ресурс]: <https://zakon.rada.gov.ua>
Режим доступу: <https://zakon.rada.gov.ua/laws/show/2694-12#Text>

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		46

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 запитами. 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.
- Формування списків виплачених кредитів.
- Формування усіх виплат які були зроблені клієнтами.

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		47

- 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

Спеціальні вимоги відсутні.

					МР.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		48

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.

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.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		50

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 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.

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		51

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," +
        "date varchar(10) NOT NULL);";
    static final String SQL_sched = "CREATE TABLE schedule" +
```

					MP.126.6144м.01	Аркуш
						52
Зм.	Аркуш	№ документа	Підпис	Дата		

```

        "(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&useLegacyDate
timeCode=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();
}

private void setConnection() throws SQLException {
    connection = DriverManager.getConnection(URL, USERNAME, PASSWORD);
    st = connection.createStatement();
    metadata = connection.getMetaData();
}

```

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		53

```

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("Adress:");
        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();
        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 reference:");
        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.getNuberreference());
        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
        (?, ?, ?, ?, ?, ?, ?)");
        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();
    }
}

```

					MP.126.6144м.01	Аркуш
Зм.	Аркуш	№ документа	Підпис	Дата		56

```

        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();
                }
            }
        }

        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]);
                }
            }
        }

```

					MP.126.6144м.01	Аркуш
						57
Зм.	Аркуш	№ документа	Підпис	Дата		

```

        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);
    }
}
```