МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

факультет програмної інженерії та бізнесу

кафедра інженерії програмного забезпечення

**Лабораторна робота № 7**

з дисципліни «Основи программної інженерії»

*назва дисципліни*

на тему: «АНАЛІЗ БІЗНЕС-ПРОЦЕСІВ З ВИКОРИСТАННЯМ IDEF0 з використанням онлайн сервісу Draw.io»

Виконав: студент 1 курсу групи № 611п освітньої програми 121 інженерія програмного забезпечення

(шифр і назва ОП)

\_\_\_\_\_\_\_\_\_\_\_\_Глухов М.О\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище й ініціали студента)

Прийняв: к.е.н., доцент каф. 603

Дем’яненко О.С.

(посада, науковий ступінь, прізвище й ініціали)

Кількість балів:

Харків 2024

**Зміст**

[**Зміст** 2](#_Toc182757800)

[**Постановка задач** 3](#_Toc182757801)

[**Зміст звіту** 3](#_Toc182757802)

[**Теоретичні відомості** 3](#_Toc182757803)

[**Виконання** 11](#_Toc182757804)

[**Висновки** 15](#_Toc182757805)

**Постановка задач**

Мета роботи: ознайомитися з функціональною методикою проектування IDEF0 на прикладі опису процесу в середовищі DrawIO. Викласти роботу для зберігання в систему контроля версій Github.

**Прорядок виконання роботи**

1. Створити новий проект в середовищі DrawIO. , (Також рекомендується встановити альбомну орієнтацію сторінки).
2. Відповідно до варіанта завдання табл. 2 розробити IDEF0 діаграму з рівнем декомпозиції не менше трьох (див. приклад).
3. Розмістити та зберігти роботу на Github. Зробити скрін шоти та вставити їх в звіт.
4. Скласти звіт.

# **Зміст звіту**

1. Постановка задачі.
2. Словесний опис бізнес-процесу відповідно до варіанта.
3. IDEF0 діаграми.
4. Викладення скриншотів .
5. Короткий опис кожної з діаграм.
6. Висновки.

**Теоретичні відомості**

Для моделювання складних систем існують добре обкатані методології та стандарти. До них відносяться, зокрема, методології сімейства IDEF, за допомогою яких можна ефективно представляти і аналізувати моделі діяльності широкого спектру складних систем в різних розрізах. При цьому глибина дослідження процесів у системі визначається самим розробником, що дозволяє не перевантажувати створювану модель зайвими даними.

Методологію IDEF0 можна вважати кінцевим етапом розвитку добре відомої графічної мови опису функціональних систем SADT (Structured Analysis and Design Teqnique).

Стандарт IDEF0 був розроблений в 1981 році в рамках великої програми автоматизації промислових підприємств ICAM (Integrated Computer Aided Manufacturing), запропонованої департаментом Військово-Повітряних Сил США. Сімейство стандартів IDEF успадкувало своє позначення від назви цієї програми (IDEF ICAM DEFinition).

Під час реалізації програми виникла необхідність розробити нові методи аналізу процесів взаємодії в промислових системах. Крім вдосконаленого набору функцій для опису бізнес-процесів, однією з вимог стало наявність ефективної методології взаємодії в рамках "аналітик-фахівець". Новий метод повинен був забезпечити групову роботу над створенням моделі, з безпосередньою участю всіх аналітиків і фахівців,зайнятих в рамках проекту. Так і виникла методологія функціонального моделювання IDEF0. З 1981 року стандарт IDEF0 зазнав кілька незначних змін, в основному обмеження характеру. Остання його редакція була випущена в грудні 1993 року Національним Інститутом Стандартів і Технологій США (NIST).

**Функціональний блок**

Графічна мова IDEF0 дивно проста і гармонійна. В основі методології лежать чотири основних поняття, перше з яких - поняття функціонального блока (Activity Box). Функціональний блок графічно зображується у вигляді прямокутника (рис. 4), і, за вимогами стандарту, його назва повинна містити дієслівну форму ( "виробляти послуги", а не "виробництво послуг").

Кожна з чотирьох сторін функціонального блока має своє значення: верхня сторона - "Керування" (Control), ліва сторона - "Вхід" (Input), права -"Вихід" (Output), нижня сторона - "Механізм" (Mechanism). Кожному функціональному блоку в рамках системи привласнюється унікальний ідентифікаційний номер.

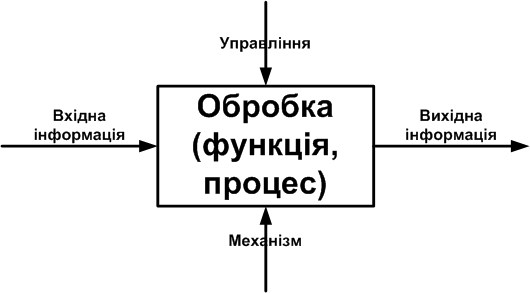


Рис. 4. Функціональний блок

За допомогою інтерфейсних дуг відображаються різні об'єкти, в тій чи іншій мірі визначають процеси, що відбуваються в системі. Такими об'єктами можуть бути елементи реального світу (деталі, вагони, співробітники і т. д.), або потоки даних й інформації (документи, дані, інструкції і т. д.). Залежно від того, куди підходить інтерфейсна дуга, ії називають "вхідною", "вихідною" або "керуючою". "Джерелом" (початком) кожної дуги може бути тільки вихідна сторона функціонального блока, а "приймачем" (кінцем) – будь-яка з трьох інших сторін.

На рис. 5 зображений функціональний блок "написання дипломної роботи". У реальному процесі є методичні рекомендації для написання дипломної роботи . Може здатися, що і диплом, і документ з рекомендаціями є вхідними об'єктами, але це не так. Насправді, диплом обробляється за правилами, які відображені в методичних рекомендаціях, які зображуються керуючою інтерфейсною дугою.

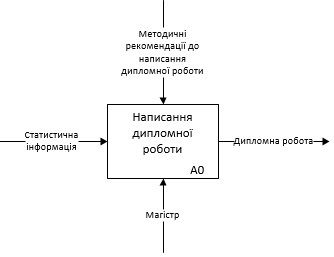


Рис.5. Приклад опису процесу написання диплому.

Однією з найбільш важливих особливостей методології IDEF-SADT є поступове введення все більш детальних уявлень моделі системи у міру розробки окремих діаграм. Побудова моделі IDEF-SADT починається з отримання всієї системи у вигляді найпростішої діаграми, що складається з одного блоку процесу і стрілок ICOM, службовців для зображення основних видів взаємодії з об'єктами поза системою. Оскільки вихідний процес представляє всю систему як єдине ціле, дане подання є найбільш загальним і підлягає подальшій декомпозиції.

Декомпозиція є умовним прийомом, що дозволяє уявити систему у вигляді, зручному для сприйняття, і оцінити її складність. В результаті декомпозиції підсистеми за певними ознаками виділяються окремі структурні елементи та зв'язки між ними. Декомпозиція служить засобом, що дозволяє уникнути труднощів у розумінні системи. Глибина декомпозиції визначається складністю і розмірністю системи, а також цілями моделювання.

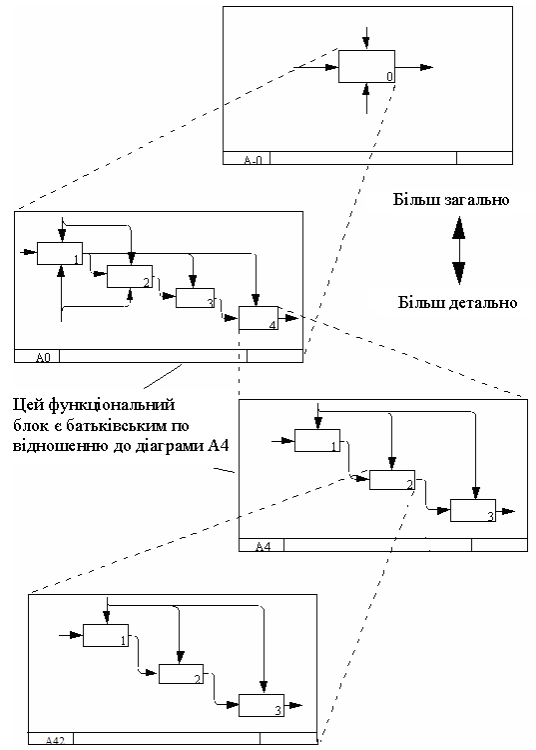


Рис. 6. Декомпозиція функціональних блоків

Слід звернути увагу на взаємозв'язок нумерації функціональних блоків і діаграм: кожен блок має свій унікальний порядковий номер на діаграмі (цифра в правому нижньому кутку прямокутника), а позначення під правим кутом вказує на номер дочірньої для цього блока діаграми. Відсутність такого позначення свідчить про те, що декомпозиції для цього блока не існує.

Часто окремі інтерфейсні дуги не варто розглядати в дочірніх діаграмах нижче або вище певного рівня. Наприклад, інтерфейсну дугу, яка зображує "деталь" на вході у функціональний блок "Обробити на токарному верстаті", немає сенсу відображати на діаграмах більш високих рівнів - це буде тільки перевантажувати їх і робити складними для сприйняття. Також буває необхідно позбутися від окремих "концептуальних" інтерфейсних дуг і не деталізувати їх глибше деякого рівня.

Для вирішення подібних завдань в стандарті IDEF0 передбачено поняття тунелювання. Символ "тунелю" (Arrow Tunnel) – дві круглі дужки навколо початку інтерфейсної дуги – позначає, що дуга була успадкована від функціонального батьківського блока і з'явилася тільки на цій діаграмі. "Тунель" навколо кінця (стрілки) інтерфейсної дуги в безпосередній близькості від блока-приймача означає, що в дочірньої по відношенню до цього блока діаграмі ця дуга відображатися та розглядатися не буде. Як правило, окремі об'єкти і відповідні їм інтерфейсні дуги "прибираються" на проміжних рівнях ієрархії. У цьому випадку вони спочатку "занурюються в тунель", а потім "повертаються з тунелю".

Обмеження складності IDEF0-діаграм

Як правило, IDEF0-моделі містять складну і концентровану інформацію. Щоб зменшити їх перевантаженість і зробити легким для читання, відповідний стандарт рекомендує розміщувати:

– від трьох до шести функціональних блоків на діаграмі. Верхня межа (шість) змушує розробника використовувати ієрархії при описі складних предметів, а нижня межа (три) гарантує, що на відповідній діаграмі досить деталей, щоб виправдати її створення;

– чотири інтерфейсні дуги, відповідні до одного функціонального блока (або виходять з нього).

Суворо дотримуватися цих обмежень необов'язково, але, як показує досвід, вони дуже корисні в реальній роботі.

Зазвичай IDEF0-моделі несуть в собі складну і концентровану інформацію, і для того, щоб обмежити їх перевантаженість і зробити легким для читання, у відповідному стандарті прийняті відповідні обмеження складності:

- Обмеження кількості функціональних блоків на діаграмі трьома-шістьма. Верхня межа (шість) змушує розробника використовувати ієрархії при описі складних предметів, а нижня межа (три) гарантує, що на відповідній діаграмі досить деталей, щоб виправдати її створення;

- Обмеження кількості відповідних до одного функціонального блоку (що виходять з одного функціонального блоку) інтерфейсних дуг чотирма.

Приклад IDEF0 діаграм для процесу «Виконання лабораторної роботи» відображено на рис. 7– 10.

ПРИКЛАД. Рис.7. Модель бізнес-процесів медичної інформаційної системи підтримки прийняття рішень кардіологічного відділення (IDEF0)

Рис.8. Перший рівень декомпозиції (IDEF0)

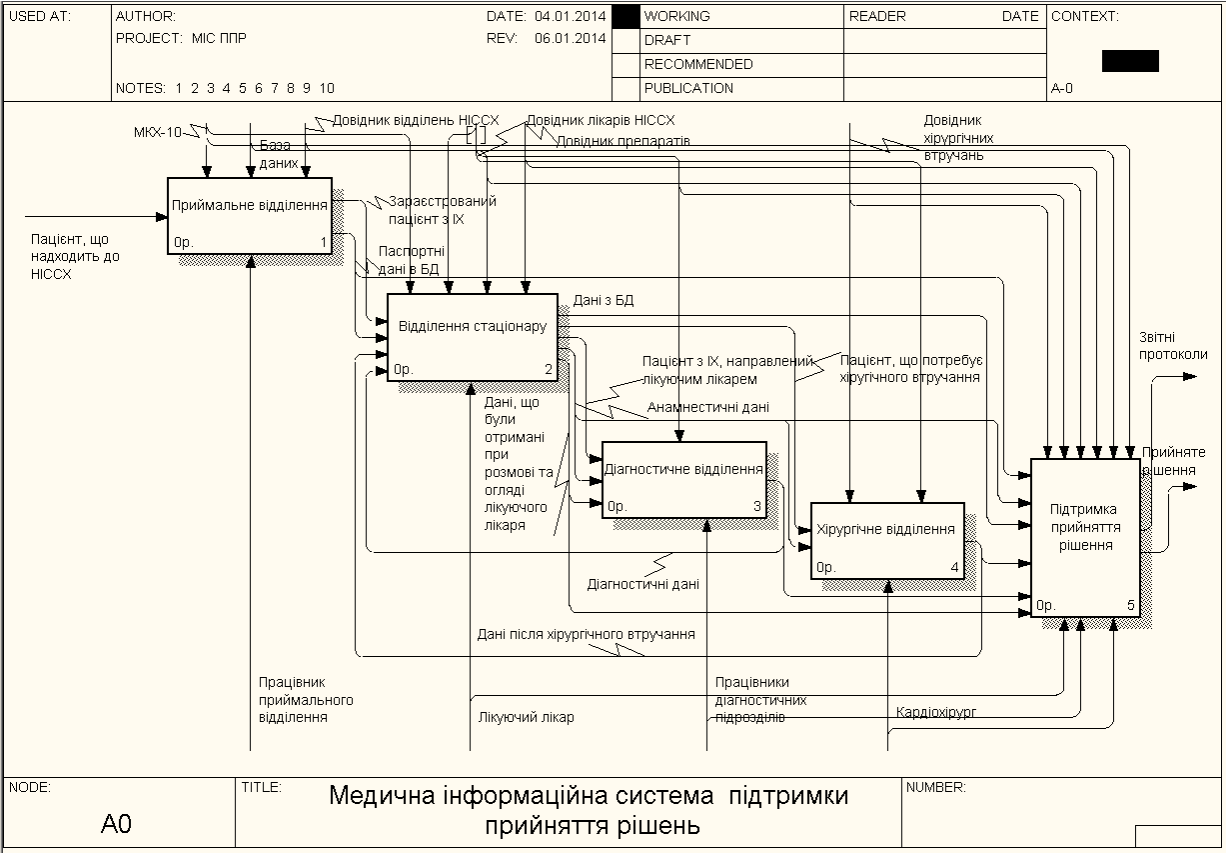
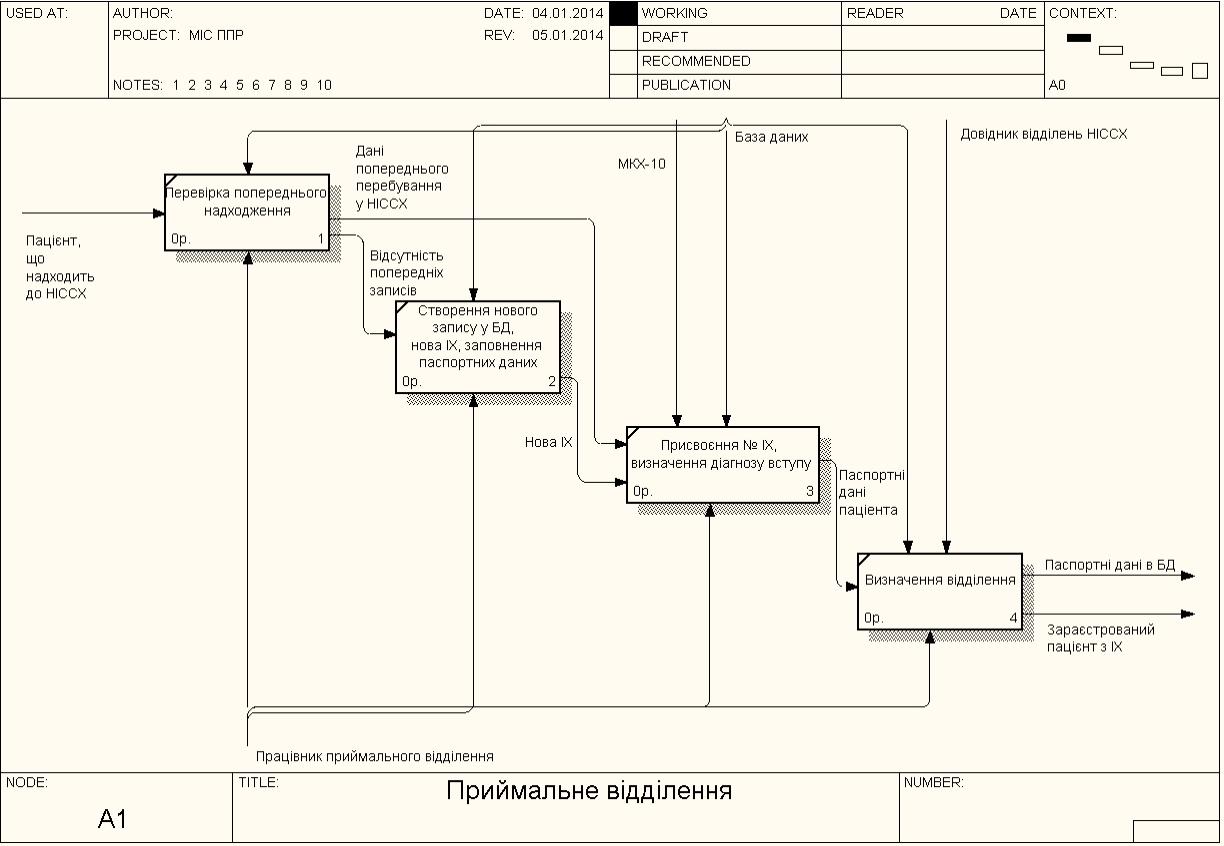
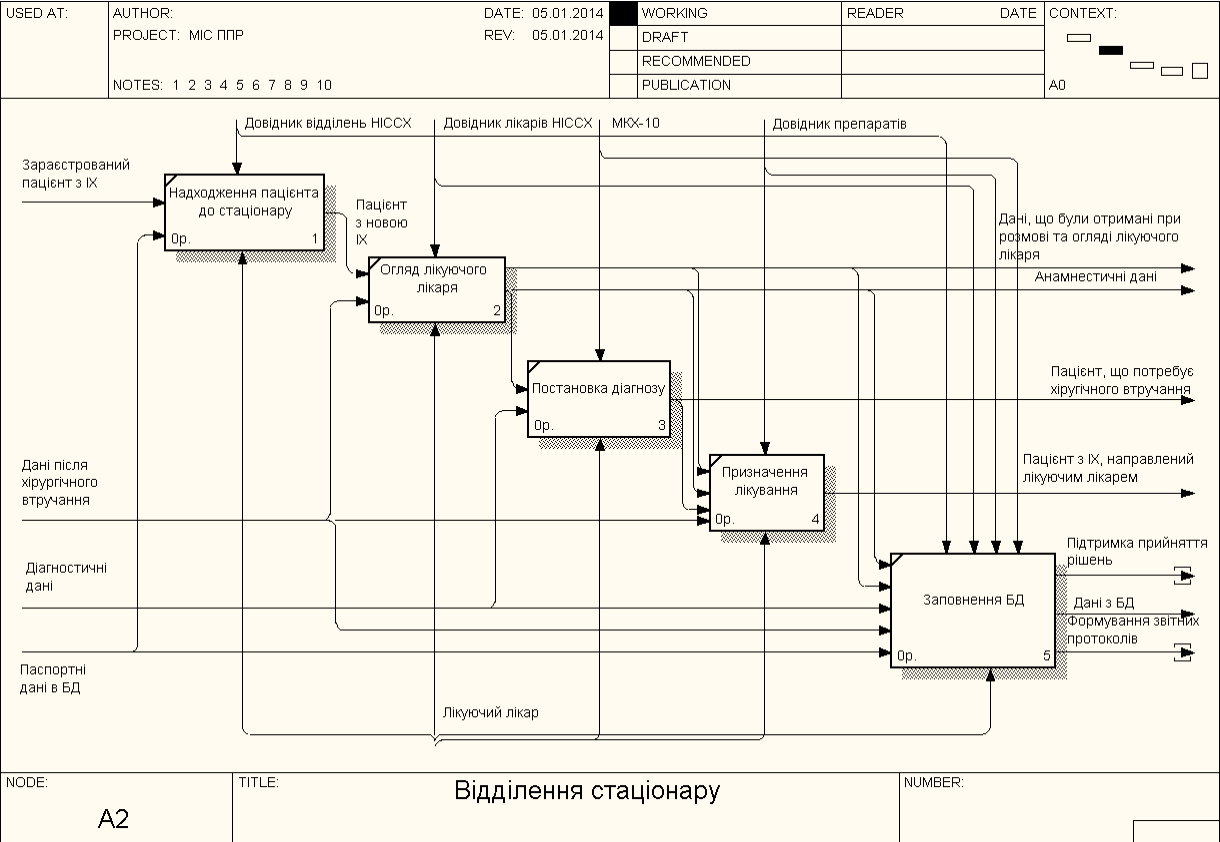


Рис.9. Діаграма декомпозиції IDEF0 (аналіз бізнес-процесів при впровадженні МІСППР)



Рис.10. Діаграма декомпозиції IDEF0 (аналіз бізнес-процесів при впровадженні МІСППР)

**Виконання**

|  |  |
| --- | --- |
| № варіанта | Процес що вимагає опису |
| 7 | Організація відеопрокату |

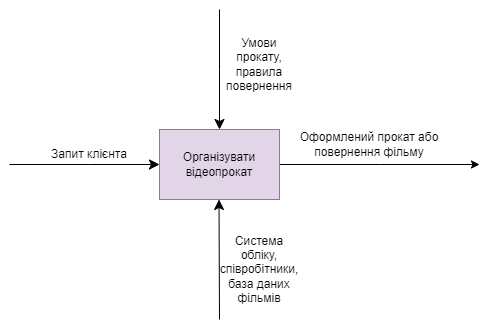


Рисунок 11– Діаграма А-0

**1. Опис діаграми A-0**

Діаграма A-0 представляє загальний процес "Організувати відеопрокат". Це найвищий рівень вашої моделі, який охоплює основний процес без розбиття на деталі.

**Input**:Запит клієнта: інформація про бажаний фільм для прокату.

**Output** :Оформлений прокат або повернення фільму: результат процесу, що включає як оформлення прокату, так і повернення фільму клієнтом.

**Control** :Умови прокату, правила повернення: визначають правила, за якими клієнт може орендувати фільм або повернути його.

**Mechanism**:Співробітники, база даних, система обліку: необхідні ресурси для реалізації процесу відеопрокату.

Ця діаграма служить як загальний огляд для процесу організації відеопрокату, включаючи основні етапи взаємодії з клієнтом і механізми для виконання цих етапів.

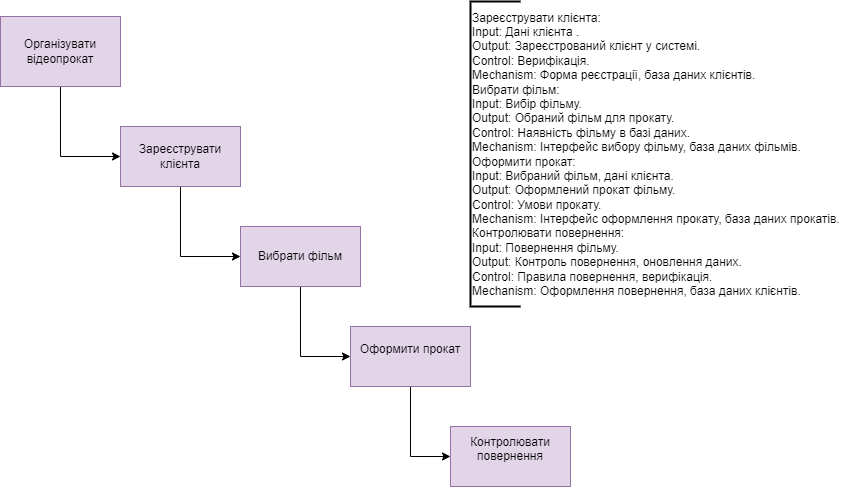


Рисунок 12– Діаграма А-1

2. Опис діаграми A-1

Діаграма A-1 — це декомпозиція процесу "Організувати відеопрокат" на підпроцеси. Цей рівень деталізує ключові етапи процесу, де кожен підпроцес має свої вхідні і вихідні дані, а також механізми і контроль.

* Підпроцес "Зареєструвати клієнта":

Input: Дані клієнта.

Output: Зареєстрований клієнт.

Control: Умови реєстрації.

Mechanism: База даних клієнтів.

* Підпроцес "Вибрати фільм":

Input: Запит клієнта.

Output: Вибраний фільм.

Control: Наявність фільму в базі.

Mechanism: Інтерфейс вибору фільмів.

* Підпроцес "Оформити прокат":

Input: Дані клієнта і вибраний фільм.

Output: Оформлений прокат.

Control: Умови прокату.

Mechanism: Система оформлення прокату.

* Підпроцес "Контролювати повернення":

Input: Прокатний договір.

Output: Повернення фільму або штраф.

Control: Умови повернення.

Mechanism: База даних, система обліку.

Ця діаграма показує, як загальний процес "Організувати відеопрокат" розбивається на менші підпроцеси, кожен з яких відповідає за конкретний етап взаємодії з клієнтом.

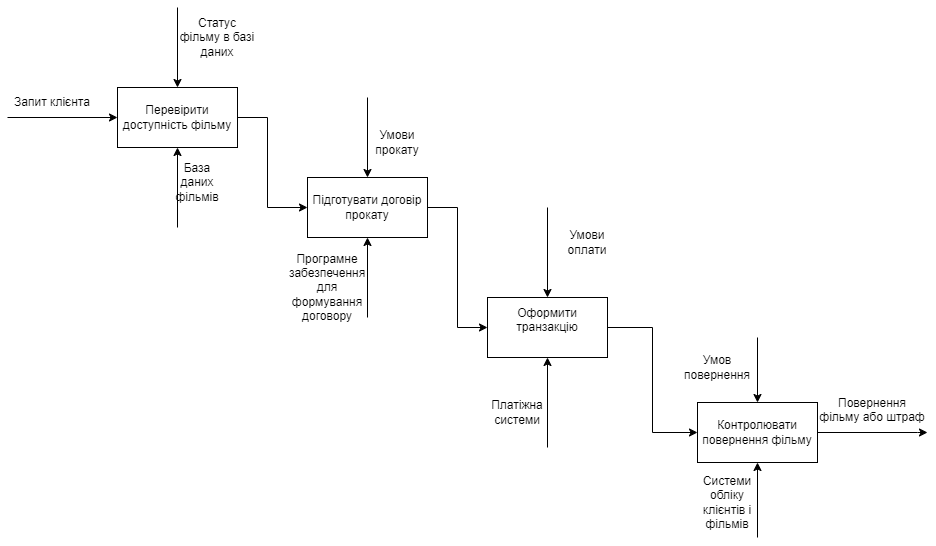


Рисунок 13– Діаграма А-2

3. Опис діаграми A-2

Діаграма A-2 деталізує один із підпроцесів "Оформити прокат" на ще менші етап

* Підпроцес "Перевірити доступність фільму":

Input: Запит клієнта (вибраний фільм).

Output: Підтверджена наявність або відмова.

Control: Статус фільму в базі даних.

Mechanism: База даних фільмів.

* Підпроцес "Підготувати договір прокату":

Input: Дані клієнта і вибраний фільм.

Output: Сформований договір.

Control: Умови прокату.

Mechanism: Програмне забезпечення для формування договору.

* Підпроцес "Оформити транзакцію":

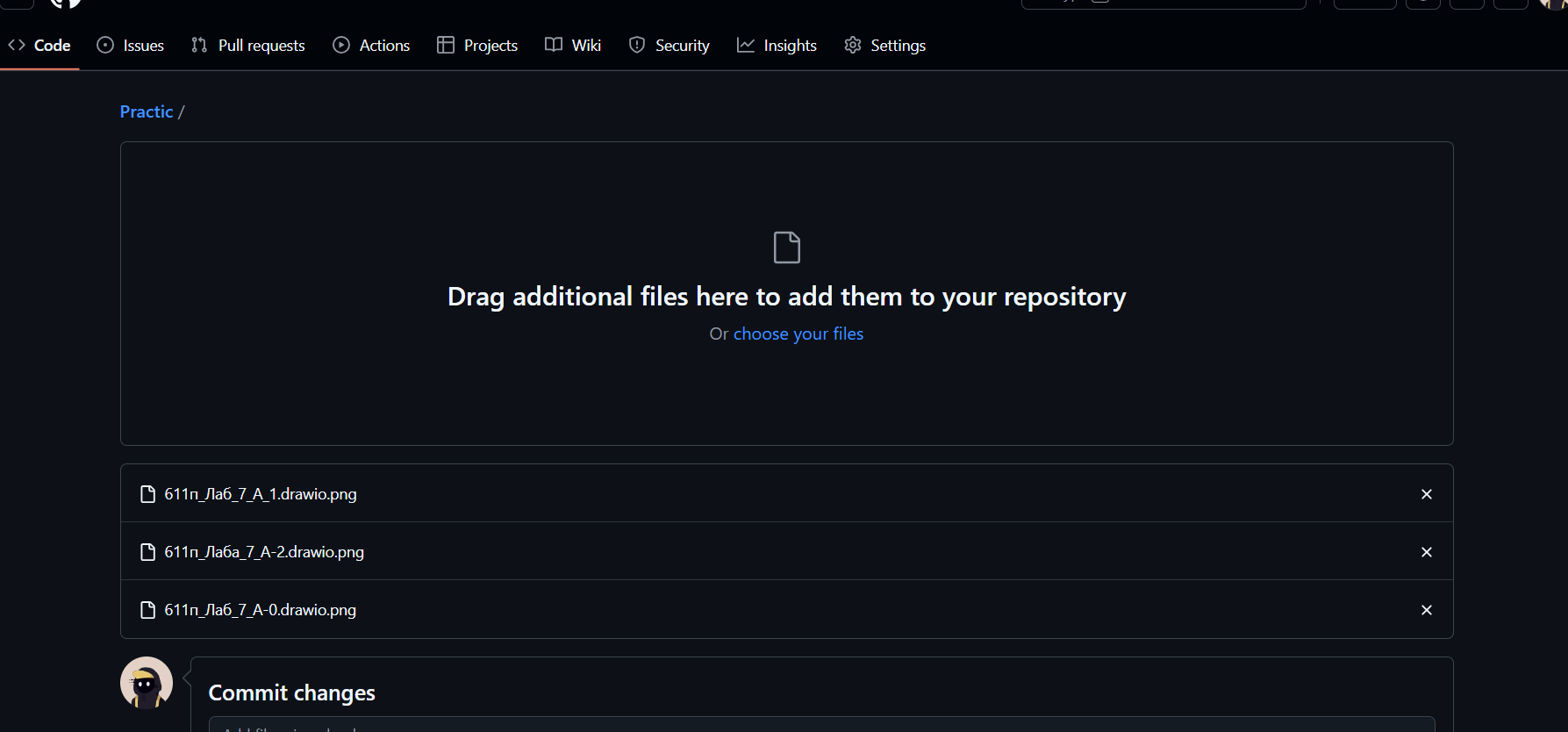
Input: Договір прокату і платіж.

Output: Завершений прокат.

Control: Умови оплати.

Mechanism: Платіжна система.

Ця діаграма показує, як процес "Оформити прокат" можна розбити на дрібніші етапи, де кожен етап виконується окремо, а потоки даних чітко визначаються для кожного етапу.

Завантаження файлів в репозиторій : 

Посилання на репозитрій:<https://github.com/Maks-obs/Practic/>

**Висновки**

У ході виконання лабораторної роботи було змоделено процес "Організація відеопрокату" за методологією IDEF0. На рівні A-0 представлено загальний процес, на рівні A-1 — основні підпроцеси (реєстрація клієнта, вибір фільму, оформлення прокату, контроль повернення), а на рівні A-2 було детально розглянуто підпроцес "Оформити прокат".Методологія IDEF0 допомогла чітко структуризувати процеси, визначити їхні взаємозв’язки та ресурси, що дає можливість оптимізувати систему відеопрокату.