ЗМІСТ

[1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ 2](#_Toc470450044)

[2 USE CASE UML ДІАГРАМИ 5](#_Toc470450045)

[3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ 6](#_Toc470450046)

[4 IDEF3 11](#_Toc470450047)

[5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ 14](#_Toc470450048)

[6 DFD 15](#_Toc470450049)

[7 ERD 16](#_Toc470450050)

[8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС 17](#_Toc470450051)

# 1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ

Області видимості об'єктів - це група об'єктів інформаційної системи, яка може бути вивчена або редагована окремою групою користувачів.

Для неавторизованих користувачів в область видимості буде входити:

1. інформації різного роду в інформаційній системі;
2. поле реєстрація в ІС;
3. поле авторизація в ІС;
4. рейтингу клінінгових компаній;

Та процеси:

1. оформлення заявки(з мінімальною кількістю полів для заповнення) на прибирання без подальшої оплати;
2. реєстрація;
3. авторизація;

Для авторизованих користувачів:

1. активні та виконані прибирання(історія замовлень);
2. особистий чорний список;
3. особистий кабінет;
4. рейтингу клінінгових компаній;
5. замовлення;
6. відповіть сервісу інформаційної системи(результат замовлення);

Та процеси:

1. налаштування особистого кабінету;
2. налаштування інформації звязаної з оплатою замовлення;
3. перегляд інформації в інформаційній системи;
4. оформлення заявки з розширеною кількістю полів на прибирання;
5. оплата заявки;
6. отримання результату заявки(з точною інформацією пр надання послуг);
7. можливість залишити відгук про ту чи інфу клінінгову компанію;
8. редагування замовлення;

Для авторизованих користувачів за областю видимості знаходяться такі об'єкти, як перегляд інформації про інших користувачів, детальна інформація та сам процес оплати послуги, детальні відомості про тих, хто надає послуги.

Для адміністраторів інформаційної системи:

1. налаштування системи;
2. резервне копіювання даних;
3. робота на сервері;

Для адміністраторів даних:

1. інформація про клінінгові компанії, які надают послуги;
2. інформація про замовлення користувача;
3. інформація про оплату;
4. рейтингу клінінгових компаній;
5. інформація про користувача;

Та процеси для послуги:

1. додати;
2. видалити;
3. редагувати.

Для користувача:

1. зареєструвати;
2. забанити;
3. надіслати листа;
4. змінити інформацію про користувача;
5. опублікувати відгук, залишений користувачем;
6. редагувати історію послуг для користувача.

Також існує головний адміністратор даних, який створений розробником. В його права також входить зміна та надання більших прав для інших користувачів.

# 2 USE CASE UML ДІАГРАМИ

Для кожної групи користувачів було створено UML – діаграму, яка ілюструє операції, що виконують різні ролі користувачів. Визначено основні характеристики та функції системи, які виконують різні групи користувачів. Основною функцією системи є збереження та обробка запитів користувачів. На даному етапі визначені класи користувачів та побудована UML – діаграма як для користувача, так і для адміністратора.

# 3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ

В данному розділі буде описний порядок взаємодії між об’єктами для кожної діаграми.

Діаграма 1.1.

Дана діаграма описує взаємодію між головними акторами - авторизованим користувачем(Auth\_user) та адміністратором(Admin), а також такими процесами, як Profile\_settings, General\_pages, Application, Basket, Black\_list, Comments, Ready\_order, Payment.

Користувач обмінюється з об’єктами (процесами) повідомленнями. Після авторизації, користувач може змінити свої налаштування профілю та зберегти їх, або вийти з особистого кабінету(розлогінитись). Зі сторони адміністратора, в налаштуваннях користувача може бути змінений доступ користувача до контенту ІС.

Також користувач може переглядати контент ІС, тобто різні інформаційні сторінки, для цього він просто звертається до системи з проханням перегляду сторінок, адміністратор зі своєї сторони може змінювати інформацію з контенту ІС, додавати нові послуги і т.д.

Для отримання замовлення, користувач повинен заповнити всі поля заявки та внести це замовлення у кошик, де звідти зможе оплатити замовлення. Після того, як користувач оплатив замовлення, адміністратор перевіряє його та зв’язує це замовлення з відповідною компанією, яка займається прибиранням офісів.

Ще одним об’єктом являються коментарі(Comments), які пишуть користувачі. Але для їх публікації, вони повинні пройти перевірку на ряд факторів, цим займається адміністратор. Після перевірки вони публікуються.

Користувач може заносити компанії, які доставляли їм сервіс у чорний список(Black\_list).

Після перегляду корзини(Basket) користувач може оплатити замовлення, надіславши повідомлення обєкту Payment. Далі, після оплати, готове замовлення передається адміністратору і він визначає яка клінінгова компанія підходить користувачеві, після цього, користувач може дізнатись всю інформацію про замовлення перейшовши до обєкту Ready order. Також дане замовлення переходить в історію замвлень, що є в обєкті Profile\_Settings.

Діаграма 1.2.

Дана діаграма описує взаємодію між головними акторами - неавторизованим користувачем(Unauth\_user) та адміністратором(Admin), а також такими процесами, як Auth, General\_pages, Application, Basket.

Користувач обмінюється з об’єктами (процесами) повідомленнями. Для повноъ функціональності користувач повинен залогінитись у ІС або зареєструватись(для цього адміністратор створює нового користувача). Це все відбувається з обєктом Auth.

Також користувач може переглядати контент ІС, тобто різні інформаційні сторінки, для цього він просто звертається до системи з проханням перегляду сторінок, адміністратор зі своєї сторони може змінювати інформацію з контенту ІС, додавати нові послуги і т.д.

Для отримання замовлення, користувач повинен заповнити всі поля заявки(Application) та внести це замовлення у кошик (Basket), де звідти зможе оплатити замовлення. Для того, щоб перейти до оплати потрібно залогінитись(від обєкта Basket прийде повідомлення, що необхідно залогінитись).

# 4 IDEF3

а) Діаграма ієрархії процесів, що зображена на рисунку 2.1, має три рівні ієрархії. На першому рівні знаходиться головний процес - Прибирання офісів, на даному етапі зображено три рівні ієрархії.

На другому рівні цей процес розбивається на чотири: «View info about», «Manipulation with application», «Management process», «Authorization». Третій рівень представляє собою декомпозицію другого. Процес «Manipulation with application», декомпонується на такі процеси: «Filling», «Application in basket», «Payment», «Receive info about order». Процес «View in» декомпонується на такі процеси: «Rate of cleaning companies», «Price-list», «Rules» , «Info about agreements» , «Contacts». Процес «Authorization» декомпонується на такі процеси: «Anonim», «Registration». «Management process» декомпонується на такі процеси: «Receive payment order», «Check the right», «Connect with company», «Send info to client». Діаграма виконана за допомогою програмного забезпечення PowerDesigner.

1) Розглянемо діаграму нотації станів об’єкта «користувач» (зображена на рисунку 3.1). Користувач, який зайшов в систему може авторизуватись, якщо користувач ще не зареєстрований в системі, він може це зробити, або функціонувати як анонімний користувач. Для авторизованих користувачів існує функція редагування даних, яка також може привести до видалення користувача з системи, ця функція існує для анонімних користувачів, які видаляються з системи одразу після виходу з неї.

2) Розглянемо діаграму нотації станів об’єкта «заява» (зображена на рисунку 3.2) та діаграму потоків процесу «Manipulation with application» (зображену на рисунку 4.2). Користувачеві системи надається право заповнювати заяву, після заповнення вона переходить у статус заповненої заяви, яка після підтвердження може бути додана у кошик, або стати готовою до оплати. З кожного статусу існує можливість редагування заяви з подальшим видаленням у разі необхідності. Далі заява може бути оплачена та перейти в статус “Оплачена заява”.

3) Розглянемо діаграму нотації станів об’єкта «замовлення» (зображена на рисунку 3.3) та діаграму потоків процесу «Management process» (зображену на рисунку 4.1). Майже всі процеси управляються адміністратором даних. Для початку замовлення перевіряють на правильність введених даних, далі визначаються дата та назва клінінгової компанії. Після цього замовлення надсилають клієнту та клінінговій компанії, яка буде займатись цим клієнтом.

4)Розглянемо діаграму потоків процесу «View info about» (зображену на рисунку 4.3). Починаючи з процесу перегляду сторінок, користувач може вибрати один із пунктів меню, після переходу до одного з пунктів, користувач обов’язково побачить інформацію про договори з компаніями та рейтинг компаній.

# 5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ

В даному етапі було визначено моделі життєвого циклу для кожного підпроцесу головного процесу. Вибір зупинився на всіх трьох моделях. Для більш простих процесів буде використана задачна модель, так як функціонал у підпроцесів споріднений, і документація для всіх однакова. Це процеси Contacts, View info about, Info about agreement, Price – list, Rules.

Payment, Filling application, Put application to basket, Receive info about order – для цих процесів обрано спіральну модель, тому що для цих процесів буде додаватись новий функціонал.

Каскадну модель для всіх інших підпроцесів було обрано тому що саме ця модель забезпечує високий рівень розробки, адже для процесів Receive paid order(adm.), Check the right(adm.), Connect with company(adm.), Send to client info about order(adm.), Login, Registration це необхідно.

# 6 DFD

Було розглянуто процес “Office cleaning”. Для більш зрозумілої логіки роботи цього процесу, його було розбито на підпроцеси, які відповідають більш елементарним діям.

Робота основного процесу складається з процесів, пов’язаних з роботою користувачем(тобто реєстрація, авторизація та зміна даних облікового запису), роботою клінінгових компаній, процесом замовлення послеги та її оплатою. Сутність Client може виступати у ролі авторизованого, неавторизованого користувача, та користувача з розширеними правами – адміністратором.

На нульовому рівні DFD було описано взаємодію цього процеса з трьома сутностями. Далі на першому рівні DFD сутності взаємодіяли з чотирма підпроцесами і так роблячи декомпозицію кожного процесу можна детально дослідити роботу головного процесу.

Процес №1 “ Office cleaning ” – головний процес задачі. Відповідає за всі головні підзадачі ІС. На вході отримує дані від клієнта(сутність Client), дані огранізації, яка відповідає за прибирання офісів (сутність Cleaning service organization) на виході – лот, який буде виставлятись на аукціоні та дані про його проведення.

Алгоритм:

1. Заповнення заявки клієнта;
2. Оплата зі сторони клієнта;
3. Визначення умов надання послуги;
4. Організація інформування про послугу для всіх сторін процесу прибирання.

Процес №1.1 “ View info about IS ” – відповідає за те, щоб клієнт міг бути проінформований про ІС та послуги, які вона надає. Тобто перегляд сторінок ІС, таких як угоди з клінінговими компаніями, рейтинг компаній, контаки, прайс ліст та інше. На вході отримує дані про варіант вибору сторінки, на виході – саму сторінку з потрібною інформацією.

Алгоритм

1. Процес вибору сторінка;
2. Процес перегляду сторінки;

Процес № 1.2 “Authorization” – відповідає за авторизацію користувачів в ІС. На вході отримує дані про користувача(логін та пароль), на виході - рекламне оголошення, в статус перебування в ІС.

Алгоритм:

1. Перебувати в ІС в якості аноніму;
2. При бажанні зареєструватись в ІС, якщо облікового запису ще немає;
3. Залогінитись в ІС.

Процес №1.3 “Management process” – відповідає за вирішення організаційних питань, управлінням готовим замовленням, на вході отримує оплачене замовлення від клієнта, на виході інформацію з наданням послуги.

Алгоритм:

1. Отримання оплаченного замовлення;
2. Перевірка на правильність введених даних;
3. Процес узгодження місця та часу від клінінгових компаній;
4. Відсилання інформації про майбутню послугу клієнтам.

Процес №1.4 “Manipulation with application”- відповідає за те, щоб всі заявки на подання товару на аукціон, від продавців були написані та оплачені. На вході отримує дані заповнену заявку від користувача, на виході оплачену заявку.

Алгоритм:

1. Написання заявки на надання послуги;
2. Відправлення заявки в кошик
3. Оплата заявки;
4. Отримання готового замовлення з часом і місцем .

Процес №1.1.1 “ Rate of cleaning comp. ” – процес перегляду сторінки з інформацією про рейтинги клінінгових компаній.

Процес №1.1.2 “ View contacts ” – процес перегляду сторінки з інформацією про рейтинги контакти ІС.

Процес №1.1.3 “ View info about agreements ” – процес перегляду сторінки з інформацією про угоди ІС з клінінговими компаніями.

Процес №1.1.4 “ View price list ” – процес перегляду сторінки з інформацією про ціни, які надає ІС.

Процес №1.1.5 “ View rules of IS ” – процес перегляду сторінки з інформацією про правила користування ІС.

Процес №1.2.1 “ Login ” – процес, при якому користувач входить в ІС під своїм логіном та паролем. Вхідні дані це логін та пароль клієнта, вихідні – статус “Authorized user”.

Процес №1.2.2 “ Registration ” – процес, при якому користувач може створити власний обліковий запис для користування в ІС. Вхідні дані - це інформація користувача, вихідні дані – інформація про власний обліковий запис в ІС.

Процес №1.2.3 “ As temp\_usr ” – відповідає за існування користувача в системі в якості аноніму.

Процес №1.3.1 “ Check the right ” - процес, який відповідає за перевірку оплаченої заявки. Вхідні дані – оплачена заявка, вихідні – результат перевірки.

Процес №1.3.2 “ Connect with company ” – відповідає за визначення часу та місця надяння послуги. Вхідні дані – оплачена заявка без часу та місця надання послуги, вихідні - оплачена заявка без часу та місця надання послуги.

Процес №1.3.3 “ Send to a client ” – відповідає за інформування клієнта про умови надання послуги. Вхідні дані - замовлення, яке ще не отримав клієнт, вихідні - замовлення зі статусом, який говорить про те, що клієнт був інформований про замовлення.

Процес №1.4.1 “ Filling ” – процес, який відповідає за заповнення заявки на надання послуги зі сторони клієнта. Вхідні дані – інформація користувача та незапонена заявка, вихідні дані - заявка с заповненою інформацією про послугу та інформацією.

Процес №1.4.2 “ Payment ” – процес, який відповідає за оплату заявки. Вхідні дані – це дані про оплату клієнта, вихідні – замовлення.

Процес №1.4.3 “ Put into the basket ” – відправлення заявки у кошик. Вхідні дані це дані з накопичувача Application, вихідні – заявка, яка може бути оплачена.

Процес №1.4.4 “ Receive info about order ” – процес отримання інформації про замовлення. Вхідні дані - замовлення, яке ще не отримав клієнт, вихідні - замовлення зі статусом, який говорить про те, що клієнт був інформований про замовлення.

# 7 ERD

На данному етапі була визначена основна логіка для зв’язків сутностей інформаційної системи. Були розглянуті п’ять основних та одна з’єднувальна сутності. ERD є нормалізованою для правильного та коректного функціонування бази даних інформаційної системи.

На діаграмі можна побачити чотири головні сутності: User, Service, Order, Order\_items.

Між кожними сутностями існують зв’язки.  
Для сутності User є зв’язок “Has” із сутністю Order, причому кожному User відповідає багато Order, тобто зв’язок один до багатьох.

Між Order та Order\_items існує зв'язок однин до багатьох, з назвою “Contain” причому по залежності за ключем Order\_id\_fk.

Також між Order\_items існує залежний звязок “Exist” із сутністю Service по ключу Service\_name та підказує який сервіс замовив користувач.

# 8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС

Було розглянуто процес “Office cleaning” та для основних процесів була визначена архітектура.

Для деяких функцій була обрана архітектура “Розподілений застосунок”. Головною причиною вибору був той факт, що доступ до даних в розподілених застосунках можливий з клієнтського ПЗ, а це є дуже зручним для адміністраторів ІС. Валідація вводимих даних, інтерфейс в залежності від ролі реалізується в клієнтській частині, що значно зменшує навантаження на сервер, за рахунок виконання функціоналу у клієнта. Збільшений рівень безпеки, так як на стороні клієнта існує функціонал захисту та перевірки вводимих даних.

Для процесу View information of IS була обрана архітектура “Віддалена БД”, так як цей процес відповідає за відображеня інформації в ІС, а отже, функціонал полягає у користувацькому інтерфейсі та отримання інформації з бази даних. Ще однією причиною вибору даної архітектури є те, що адмініструвати дану БД не є необхідністю.

Для деяких процесів була обрана архітектура “Віддалене представлення даних” , так як основний функціонал даних процесів полягає у маніпулюванні інформацією та роботою з базою даних, користувач клієнтської частини використовує всі функції сервера на пряму і сам формує результат, а зі сторони серверу проходить вставка, видалення або редагування інформації.

Процес “Payment” організовується за архітектурою Представлення даних, тому що весь функціонал розміщується на стороні серверу банку, а від користувача вимагається лише ввід даних.