В ході виконання першого етапу курсової роботи мною були визначені основні підгрупи та ролі користувачів інформаційної системи. Також було вказано, якими правами володіють відповідні користувачі системи та визначено їхні повноваження.

В підсумку отримали декілька підгруп, які можна назвати наступним чином:

Авторизовані Користувачі, Неавторизовані користувачі та Адміністратори ІС.

В свою чергу, Авторизовані Користувачі діляться на дві ролі: Клієнт та Водій. Водій – це категорія Авторизованих Користувачів з розширеними правами (зокрема правами доступу до доступних замовлень та прийому їх). Клієнт – звичайний користувач ІС, що належить до категорії Авторизованих Користувачів і має розширені права (на відміну від неавторизованих користувачів), такі як отримання знижки на замовлення та залишення відгуку про дану машину чи водія.

Підгрупа Адміністраторів поділяється на Адміністраторів Даних та Системних Адміністраторів, які мають власні права та обов’язки.

В ході виконання другого етапу курсової роботи було побудовано Use Case діаграми для Інформаційної Системи «Виклик таксі». Даний вид діаграми дозволяє розробнику бачити візуалізовані права користувачів та їхні ролі, що є дуже важливим при розробці програмного продукту. Також ці діаграми дозволяють досить непогано візуалізувати послідовності переходу користувача між інтерфейсами кінцевої системи, тобто дозволяють визначити права певних ролей користувачів.

Діаграми було фактично створено для чотирьох типів користувачів: неавторизований користувач, авторизований «клієнт» користувач, авторизований «водій» користувач та адміністратор даної системи.

В ході виконання третього етапу курсової роботи було побудовано діаграму послідовностей (sequence diagram) для акторів даної Інформамційної Системи. Результати роботи можна описати наступним чином: .

* Для водія передбачено тільки робота із системою після реєстрації. Незареєстровані водії повинні пройти процедуру реєстрації. Для аутентифікації потрібно ввести коректний логін-пароль по легуванні. Після цього водієві будуть доступні опції шукати замовлення, а його місцезнаходження буде передано системі пошуку клієнта. Система пропонує замовлення водієві, а він може прийняти та відхилити його. При відмові від замовлення повідомлення передається адміністратору, а він в свою чергу надсилає протокол відміни на блок пошуку таксі, щоб система шукала іншого водія для клієнта.
* Для клієнта існують дві ролі: незареєстрований користувач та зареєстрований, які мають власні переваги та особливості. Зокрема, для кожного з них доступна можливість перегляду відгуків, замовлення таксі та оскарження замовлення (за потребою). Проте, для зареєстрованого користувача існують такі привілеї, як оформлення знижки, залишення коментарів, перегляд історії відгуків, перегляд та редагування власного акаунту.

За допомогою загальноприйнятих операторів для фреймів взаємодії описана досить логічно. Зокрема, для користувачів передбачений оператор альтернативи, який розбиває всю систему на зареєстрованих та незареєстрованих користувачів. Для перевірки правильності введення даних при реєстрації та логіні були передбачені відповідні цикли.

Дані діаграми зможуть допомогти розробнику розібратися у послідовності виконання різних процесів різними ролями користувачів.

В ході виконання четвертого етапу курсової роботи було побудовано діаграму ієрархії основних процесів, а також діаграми стандарту IDEF3 (діаграми нотації станів OSTN та діаграми опису послідовностей етапів процесу PFDD) для клієнтів системи «Виклик таксі». Опис даних діаграм наведений нижче.

На Рисунку 1.1 наведено діаграму ієрархії процесів, на якій чітко видно на які саме процеси розбивається основний процес. Декомпозиція першого рівня процесу «Виклик таксі» складається з наступних процесів: «Аутентифікація», «Введення основних параметрів», «Робота з коментарями», «Підрахунок ціни», «Показання параметрів таксі», «Пошук клієнта». Дана діаграма призначена одночасно і для водія, і для клієнта, але при продукту мають бути передбачені області видимості цих ролей. Для клієнта не будуть доступні деякі опції, такі як «Пошук клієнта», для водія не буде доступна опція «Коментування поїздки», для незареєстрованого користувача не буде доступу до своєї сторінки, залишення коментарів та отримання знижки.

Декомпозиція другого рівня складається з декомпонованих процесів першого рівня і містить відповідно наступні процеси:реєстрація, видалення користувача, логування, введення форми запиту та інших параметрів виклику, перегляд, залишення та видалення коментарів, підрахунок ціни та кілометрів, перегляд доступних машин, водіїв та інших різних процесів, пов’язаних із водієм.

На Рисунку 2.1 зображена діаграма, що описує процеси користувача. Отже, користувач заходить в систему і зразу стає перед вибором (J1): пройти процес аутентифікації чи ввести свої дані для пошуку авто і залишитись не залогованим користувачем. При виборі останньому користувач обмежений в своїх діях і може тільки викликати таксі по параметрам, що він ввів чи (J2) переглянути коментарі про водіїв. При вході на перехресті (J1) як аутентифікований користувач, клієнт має право на (J3) виклик таксі по параметрам, що вказані в нього на сторінці, роботу з коментарями, що ділиться (J5) на перегляд та залишення коментарів та роботу зі знижками, що ділиться (J4) на перегляд доступних знижок та підрахунок вартості поїздки зі знижкою.

На рисунку 2.2 зображена діаграма процесу виклику таксі. На вхід подаються дані, що користувач ввів раніше, а далі йде розгалуження (J1), що показує опції: система може підрахувати ціну, знаходити машину та показати користувачу доступні райони. Далі, на (J2) відбувається об’єднання процесів знаходження машини та перегляду доступних районів та по завершенню цих процесів відбувається контакт з водієм.

На рисунку 2.3 зображений процес аутентифікації. Коли починається користувацька сесія і він вибрав опцію аутентифікації, в нього є два вибора: (J1) ввести власні пароль та логін і зайти в систему чи пройти реєстрацію і зайти в систему. Після завершення сесії дані користувача видаляються.

На рисунку 2.4 зображено процес пошуку клієнта водієм. У водія доступні 2 опції (J2): він може почати приймати системні пропозиції чи шукати користувача власноруч. При виборі першого варіанту, до нього приходить заявка, яку він може (J5) прийняти або відхилити. Якщо він приймає заявку, він починає обслуговувати клієнта, а якщо відхилить – пошук продовжиться. Якщо ж було обрано пошук власноруч, то водій може (J6) продивитися райони і доступних клієнтів та, власне шукати клієнта. Якщо він знайде потрібного клієнта, то може запропонувати йому свої послуги.

На рисунку 3.1 зображено діаграму опису стану об’єкту «замовлення». Як видно, Замовлення переходить зі стану не створеного у стан створеного методом процесу System Preposition. Тобто як тільки система знаходить клієнта і передає його водію. Потім об’єкт переходить у стан прийнятого чи відхиленого відповідними процесами, якщо водій прийняв чи відхилив замовлення. При прийнятому замовленні, він може перейти у стан виконаного через процес обслуговування клієнта водієм. Далі об’єкт переходить у стан дропнутого через процес видалення замовлення.

На рисунку 3.2 зображено OSTN діаграму для об’єкту користувач при аутентифікації. До системи користувач потрапляє неаутентифікований. В нього є 3 варіанти дій: залишитися анонімним (тоді буде створено temp\_user), зареєструватися в системі процесом реєстрації чи зайти до сайту через свої логін і пароль по процесу логінування. Причому, якщо користувач зайшов до системи під власним акаунтом, то він має право на внесення змін до своєї сторінки. Далі користувач буде вилучений з системи через процес Del User.

На рисунку 3.3 описано діаграму станів об’єкту водій. Спочатку водій заходить до системи неаутентифікованим. Далі він вже за відомим процесом аутентифікується і стає готовим до прийому і виконання замовлень через процес пошуку клієнтів. Далі, у водія є дві можливості: Він може шукати клієнта вручну або за допомогою системи. При пошуку за допомогою системи водій може відмовитися від запропонованого клієнта (decline preposition) і повернутися у стан готовності. При прийнятті замовлення, водій переходить у стан зайнятого. Після виконання замовлення він може повернутися до стану готового до виконання або стати вилученим.

В ході виконання 5 етапу курсової роботи було визначено життєві цикли для процесів системи.

В табл. 1.1 вказано основні підпроцеси головного процесу системи та обрана модель життєвого циклу для їхньої розробки.

Для процесу Аутентифікації передбачено каскадну модель життєвого циклу, оскільки даний процес повинен повинен містити найвищий рівень безпеки, оскільки включає в себе введення персональних даних користувача, таких як логін, пароль, електронна пошта, телефон, особиста адреса, дані профілю.

Для процесу Введення параметрів поїздки передбачено каскадну модель, оскільки вхідними даними для даного процесу слугуватимуть такі дані користувача, як поточна адреса, телефон, ім’я та власні параметри поїздки, тому тут потрібно забезпечити високий рівень безпеки.

Для процесу Робота з коментарями передбачено спіральну модель життєвого циклу, оскільки даний функціонал буде повністю засновуватись на записах в базу даних, яка при розробці за спіральною моделлю розробляється першочергово. Також від даного процесу не вимагається налаштувань приватності, оскільки коментарі будуть в публічному доступі і зможуть бути переглянуті навіть гостями системи.

Для процесу Підрахування ціни передбачено каскадну модель, оскільки дана система є розрахунковою і повинна чітко і правильно підраховувати вартість поїздки.

Для процесу Перегляд параметрів таксі передбачено спіральну модель життєвого циклу, оскільки дана система не має високих вимог до безпеки, проте повинна мати змогу на швидке редагування.

Для процесу Пошуку клієнта передбачено каскадну модель життєвого циклу, оскільки це ключовий процес системи і розробник повинен забезпечити безперебійну, точну, безпечну роботу процесу. Тим більше, цей процес є критичним по часу, тому що потрібно забезпечити пошук клієнта водію та пошук водія клієнту в найкоротші терміни.

Для процесу роботи користувача зі своєю сторінкою була обрана каскадна модель, оскільки вона повинна забезпечувати високий рівень безпеки при роботі користувача зі своїми даними (зміна паролю, введення особистих даних, тощо).

В ході виконання шостого етапу курсової роботи було побудовано DFD діаграми, описано основні процеси та накопичувачі даних.

Згідно створених діаграм потоків даних згодом буде легко уявити процеси та їх функціонування, а також створити цілісну, правильно функціонуючу Інформаційну Систему

В ході виконання сьомого етапу курсової роботи було побудовано ERD діаграми, що описують зв’язки між п’ятьма сутностями системи «Виклик Таксі».

Дані діаграми описують, як мають взаємодіяти таблиці створюваної системи.

В ході роботи над восьмим етапом курсової роботи було обрано та представлено архітектуру, згідно якої будуть функціонувати блоки системи. За допомогою заданих архітектур можна уявити, як буде функціонувати система в цілому.