

ВИСНОВКИ

Отже у ході виконання першого етапу курсової роботи були розглянуті всі можливі типи користувачів:

- неавторизовані;
- авторизовані;
- адміністратори даних;
- адміністратори інформаційної системи.

Розглянуті їх області видимості та можливості в даній інформаційній системі.

Отже у ході виконання другого етапу курсової роботи було розглянуто дві основні групи користувачів та складено Use Case діаграму для кожного. Можна помітити, що в неавторизованих користувачів набагато менше можливостей, а саме:

- 1) Перегляд відкритої(основної) інформації;
- 2) Авторизація/Реєстрація.

У авторизованих:

- 1) Перегляд відкритої(основної) інформації;
- 2) Вихід із профіля;
- 3) Завантаження роботи;
- 4) Перегляд робіт що вже були завантажені;
- 5) Перегляд особистого профіля.

У адміністраторів можливостей набагато більше чим у когось іншого, наприклад до всіх вище вказаних додаються наступні:

- 1) Перевірка робіт інших користувачів;
- 2) Коментування робіт інших користувачів;
- 3) Реєстрування користувачів;
- 4) Видалення профілей інших користувачів;
- 5) Перегляд особистого профіля;

б) Перегляд робіт що вже були завантажені.

Отже у ході виконання третього етапу курсової роботи було побудовано діаграми послідовності для основних видів користувачів, а саме:

- неавторизовані користувачі;
- авторизовані користувачі;
- користувачі адміністратори.

Як видно, що неавторизовані користувачі можуть лише:

- зареєструватись;
- ввійти під своїм персональним логіном;
- переглянути відкриту інформацію;
- написати у службу підтримки.

У авторизованого користувача до вище вказаних функцій додаються наступні:

- відредагувати особистий профіль;
- завантажити роботу;
- переглянути завантажені роботи.

Користувач типу адміністратор має всі вище вказані можливості і також наступні:

- перевірка робіт;
- блокування користувачів;
- надання користувачам додаткових можливостей.

Під додатковими можливостями розуміється розширення області видимості звичайних зареєстрованих користувачів.

Ієрархічне дерево яке зображено на рисунку 2.2.1 має три рівні ієрархії. На першому рівні знаходиться саме процес «Visit site» на другому знаходяться три процеси які місять в собі інші.

- 1) «Create work» - цей процес містить у собі такі процеси як:

- 1.1) create;
 - 1.2) edit;
 - 1.3) delete.
- 2) «Auth» - цей процес містить у собі такі процеси як:
- 2.1) registration;
 - 2.2) log in;
 - 2.3) temp user.
- 3.) «V» - цей процес містить у собі такі процеси як:
- 3.1) view;
 - 3.2) comment;
 - 3.3) edit comment.

Для кожного із головних процесів було складено IDEF(3) та OSTN діаграми які можна спостерігати на рисунках у третьому та четвертому розділах.

Розглянемо наприклад діаграму потоків процесу «авторизація» який зображено на рисунку 2.3.1 . Дана діаграма має п'ять процесів, але також включає в себе діаграму потоків процесів створення роботи або коментаря. Між процесами існують розгалуження. В даному прикладі використовується розгалуження «XOR» - виключне або, та &. Перше має на увазі, що виходящий процес розпочнеться тільки тоді як один із входящих закінчиться. Так наприклад при підключенні до сервісу користувач має зареєструватися, а потім або виконати «log in» .

Розглянемо тепер OSTN діаграму для об'єкта «user». При підключенні до даного процесу користувач зможе або виконати «log in» або «registration» або знаходитись у стані «temp user». Після реєстрації, входження в систему під зареєстрованим логіном чи при знаходженні у стані «temp user» користувач може бути видалений, тобто його життєвий цикл у даній інформаційній системі буде закінчений, виконується автоматично.

Аналогічно було побудовано діаграми для інших об'єктів та процесів.

В процесі виконання п'ятого етапу курсової роботи було виявлено які з підпроцесів якою моделлю життєвого циклу будуть розроблятися і обгрунтовано чому.

Отже маємо наступне, підпроцеси процесу Auth будуть розроблятися за допомогою каскадної або каскадної з поверненням моделлю життєвого циклу, а всі інші за допомогою спіральної моделі життєвого циклу.

У ході виконання етапу курсової роботи було побудовано DFD діаграму інформаційної системи «Антиплагіат робіт студентів». Головний процес називається «Ordering» який поділяється на наступні:

- 1) Work with user;
- 2) Work with work;
- 3) Create work;
- 4) View work.

Які у свою чергу поділяються на:

- 1.1) Login– для «Work with user»;
- 1.2) Edit profile – для «Work with user»;
- 1.3) Registration – для «Work with user»;
- 1.4) View profile – для «Work with user»;
- 1.5) ADD_work – для «Work with work»;
- 1.6) VIEW_work– для «Work with work»;
- 1.7) Edit work – для «Work with work»;
- 1.8) Create work – для « Create work»;
- 1.9) Check – для « Create work»;
- 1.10) View work – для « View work»;

Використовуються такі накопичувачі:

- 1) Created work;
- 2) Viewed work;
- 3) work;
- 4) works;
- 5) users.

Концептуальну ERD діаграму можна побачити на рисунку А.1.

На ній зображено наступні сутності:

- 1) Users;
- 2) Comment;
- 3) Work.

Сутність Users має такі атрибути:

- 1) Record book;
- 2) Group;
- 3) Name pk;
- 4) Password;
- 5) Role;

Сутність Comment має такі атрибути:

- 1) Id comment pk
- 2) Name work fk
- 3) Name fk
- 4) Record book fk.

Сутність Work має такі атрибути:

- 1) Body
- 2) Time
- 3) Name fk
- 4) Id comment fk
- 5) Name work pk.

Сутність «User» та сутності «Comment» мають зв'язок один до багатьох, тобто декілька коментарів можуть бути залишені одним користувачем та необов'язково що коментарі будуть залишені користувачем взагалі.

Сутність «User» та сутності «Work» мають зв'язок один до багатьох, тобто декілька робіт можуть бути залишені одним користувачем та необов'язково що користувач матиме роботу.

Сутність «Work» зв'язана з собою ж, тобто з сутністю «Comment» зв'язком один до багатьох тобто до роботи може бути прикріплено декілька коментарів або коментарів може не бути взагалі.

У ході виконання восьмого етапу курсової роботи було проаналізовано чотири основних стани use case діаграми і підібрано для кожного архітектуру клієнт-серверних інформаційних систем. І як результат для всіх окрім другого стану - розподілене представлення даних. Тому що в цих станах немає функцій які б «спілкувались» із сервером часто, на сторінках цих процесів проходить валідація яка міститься на стороні клієнта. А для другого використовуємо розподілений застосунок, тому що, дані про користувача та його активність у системі дістаються з серверу, та за умови їх зміни знову іде робота з сервером що потребує додаткової захищеності.

У ході виконання даного етапу курсової роботи було реалізовано графічний інтерфейс користувача інформаційної системи і детально розглянуто такі сторінки як:

- 1) Головна;
- 2) Реєстрація;
- 3) Завантаження роботи;
- 4) Перегляд завантажених робіт;

Для кожного поля на сторінках вище реалізовано перевірки на ввід даних.