Міністерство освіти і науки, молоді та спорту України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

ЗВІТ

про виконання модульної контрольної роботи

з дисципліни

“ Технології розробки корпоративних застосувань ”

**Тема:** «Система замовлення букетів»

|  |  |  |
| --- | --- | --- |
| Прийняв: |  | Виконав: |
| Подрубайло О.О. |  | Студентка 3-го курсу  гр. ІП-41 ФІОТ  Охотнік Тетяна |

Київ 2016

1) анотація;

2) вступ;

3) загальні положення;

4) вимоги до системи;

5) опис видів забезпечення системи;

6) опис системи;

7) висновки;

8) перелік посилань.

# Анотація

Пояснювальна записка до модульної роботи: 20 сторінок, 24 рисунки, 4 посилання.

Об’єкт дослідження: реалізація системи для замовлення букетів.

Мета роботи: дослідження методів JavaEE для реалізації системи, створення програмного забезпечення для вирішення поставленої задачі.

# **ВСТУП**

Модульна робота присв’ячена розробці програмного забезпечення для надання користувачу можливості в онлайн режимі обрати букет та замовити його на вказану адресу.

# **РОЗДІЛ 1 ЗАГАЛЬНІ ПОЛОЖЕННЯ**

Головною ідеєю розробки програмного додатку є автоматизація дій користувача системи. Об’єктом автоматизації є система замовлення букетів. Головна мета цієї системи – надати користувачу можливість придбати необхідний товар у online режимі.

Система повинна мати можливість адміністрування довіреною людиною.

У відповідності до загальних вимог можна виділити основні бізнес процеси, які повинні бути у розроблювальній системі:

1) Вхід в систему

Мета цього бізнес-процесу полягає у відокремленні груп користувачів за їх правами доступу та надати необхідні для кожної групи ресурси. Звичайний користувач, адміністратор та неавторизований користувач мають різний ступінь доступу до інформації та для роботи з нею.

2) Реєстрація

Бізнес-процес надає користувачу можливість створити власний аккаунт з його особистою інформацію та певним рівнем доступу до ресурсів сайту.

3) Купівля товару

Включає в себе два бізнес-процеси: додавання замовлення у корзину та підтвердження замовлення товарів з корзини. Купівля товару є головним бізнес-процесом системи і надає користувачу можливість маніпулювати товарами у корзині – впродовж сесії видаляти, додавати товари та підтвердити замовлення.

5) Адміністрування сайту

Адміністратор повинен слідкувати за каталогом товарів, наприклад додавати товар у базу, оновлювати його або видаляти. За допомогою додатку адміністратор має маніпулювати з інформацією сайту.

Наведемо словесні описи вказаних бізнес-процесів.

Бізнес-процес №1. Вхід у систему

|  |  |
| --- | --- |
| ID | UС001 |
| Назва | Вхід у систему |
| Учасники | Користувач, Система |
| Передумови | Користувач зареєстрований |
| Результат | Користувач потрапив до свого аккаунту |
| Основний сценарій | |  |  |  | | --- | --- | --- | |  | Дії користувача | Дії системи | | 1 | Користувач натискає кнопку “Вхід” |  | | 2 |  | Система надає користувачу форму авторизації з наступними полями: “Пароль”, “E-mail” | | 3 | Користувач заповнює поля “Пароль” та “E-mail ”. |  | | 4 | Користувач натискає кнопку “Увійти”. |  | | 5 |  | Система оброблює запит користувача, перевіряє валідність даних та переадресує його на відповідну його правам сторінку | | 6 | Користувач завершує дію. |  | |
| Виключні ситуації | Відсутні |
| Сутності | Поля “Пароль”, “E-mail ”.  Користувач. |

Бізнес-процес №2. Реєстрація в системі

|  |  |
| --- | --- |
| ID | UС002 |
| Назва | Реєстрація у системі |
| Учасники | Користувач, Система |
| Передумови | Користувач не зареєстрований |
| Результат | Користувач зареєстрований |
| Основний сценарій | |  |  |  | | --- | --- | --- | |  | Дії користувача | Дії системи | | 1 | Користувач натискає кнопку “Зареєструватися” |  | | 2 |  | Система надає користувачу форму реєстрації з наступними полями: “Ім’я”, “Прізвище”, “Пароль”, “Підтвердження паролю”, “E-mail”, “Телефон”. | | 3 | Користувач заповнює поля “Ім’я”, “Прізвище”, “Пароль”, “Підтвердження паролю”, “E-mail”, “Телефон”. |  | | 4 | Користувач натискає кнопку “Зареєструватися”. |  | | 5 |  | Система оброблює запит користувача, перевіряє валідність даних та реєструє його у системі. | | 6 |  | Система повідомляє користувача про успішну реєстрацію. | | 7 | Користувач завершує дію. |  | |
| Виключні ситуації | Користувач з обраним e-mail вже існує. |
| Сутності | Поля “Ім’я”, “Прізвище”, “Пароль”, “Підтвердження паролю”, “E-mail”, “Телефон”.  Користувач. |

Бізнес-процес №3. Підтвердження замовлення товарів з корзини

|  |  |
| --- | --- |
| ID | UС003 |
| Назва | Підтвердження замовлення товарів з корзини |
| Учасники | Користувач, Система |
| Передумови | Користувач авторизований  У корзину додано мінімум один товар |
| Результат | Корзина пуста, товари замовлено |
| Основний сценарій | |  |  |  | | --- | --- | --- | |  | Дії користувача | Дії системи | | 1 | Користувач натискає кнопку “Корзина” |  | | 2 |  | Система надає користувачу список доданих у корзину букетів, виводить загальну суму покупки, виводить форму з полями: “Адреса доставки”, “Ім’я одержувача”, “Дата доставки”. | | 3 | Користувач заповнює поля “Адреса доставки”, “Ім’я одержувача”, “Дата доставки”. |  | | 4 | Користувач натискає кнопку “Замовити”. |  | | 5 |  | Система оброблює запит користувача, перевіряє валідність даних та додає у базу. | | 6 |  | Система повідомляє користувача про успішне замовлення та очищує корзину. | | 7 | Користувач завершує дію. |  | |
| Виключні ситуації | Товари у каталогу відсутні |
| Сутності | Поля “Адреса доставки”, “Ім’я одержувача”, “Дата доставки”, товари корзини, сумарна ціна  Користувач. |

Бізнес-процес №4. Додавання замовлення у корзину

|  |  |
| --- | --- |
| ID | UС004 |
| Назва | Додавання замовлення у корзину |
| Учасники | Користувач, Система |
| Передумови | Користувач авторизований |
| Результат | Товари додану у корзину |
| Основний сценарій | |  |  |  | | --- | --- | --- | |  | Дії користувача | Дії системи | | 1 | Користувач натискає кнопку “Каталог” |  | | 2 |  | Система виводить список наявних у базі товарів | | 3 | Користувач натискає кнопку «Додати». |  | | 4 |  | Система зберігає обраний товар, додає у корзину. | | 5 | Користувач завершує дію. |  | |
| Виключні ситуації | Товари у каталогу відсутні |
| Сутності | Товар, корзина, користувач. |

Бізнес-процес №5. Додавання товару у базу

|  |  |
| --- | --- |
| ID | UС005 |
| Назва | Додавання товару у базу |
| Учасники | Адміністратор, Система |
| Передумови | Адміністратор авторизований |
| Результат | У каталог додано новий товар |
| Основний сценарій | |  |  |  | | --- | --- | --- | |  | Дії користувача | Дії системи | | 1 | Адміністратор натискає пункт меню “Управління” |  | | 2 |  | Система надає користувачу список дій: «Додати товар»,  «Видалити товар»,  «Редагувати товар» | | 3 | Адміністратор натискає кнопку “Додати товар”. |  | | 4 |  | Система виводить форму з наступними полями: «Назва товару», «Опис», «Артикул», «Ціна», «Наявність», «Фото». | | 5 | Адміністратор заповнює поля:  «Назва товару», «Опис», «Артикул», «Ціна», «Наявність», «Фото». |  | | 6 | Адміністратор натискає кнопку «Підтвердити». |  | | 7 |  | Система оброблює запит адміністратора, перевіряє валідність даних та додає у базу. | | 8 |  | Система виводить повідомлення по успішне додавання товару у базу | | 9 | Адміністратор завершує взаємодію |  | |
| Виключні ситуації | Користувач з обраним логіном вже існує. |
| Сутності | Поля Логін”, “Пароль”, “E-mail”, “ПІБ”, “Телефон”, “Адреса”.  Користувач. |

Розроблена система призначена для автоматизації роботи магазину квітів, залучення нових клієнтів та зменшення витрат магазину.

Головна задача розроблювальної системи – надати зручний доступ до даних і маніпулювання ними.

В результаті впровадження системи планується вирішити такі задачі:

* Розподілений доступ до системи
* Захист даних
* Управління системою адміністратором
* Доступний інтерфейс користувача

**РОЗДІЛ 2**

**ВИМОГИ ДО СИСТЕМИ**

Аналізуючи попередній розділ, виділимо основні вимоги до системи:

* Купівля букетів
* Захист даних
* Адміністрування системи
* Доступний інтерфейс користувача

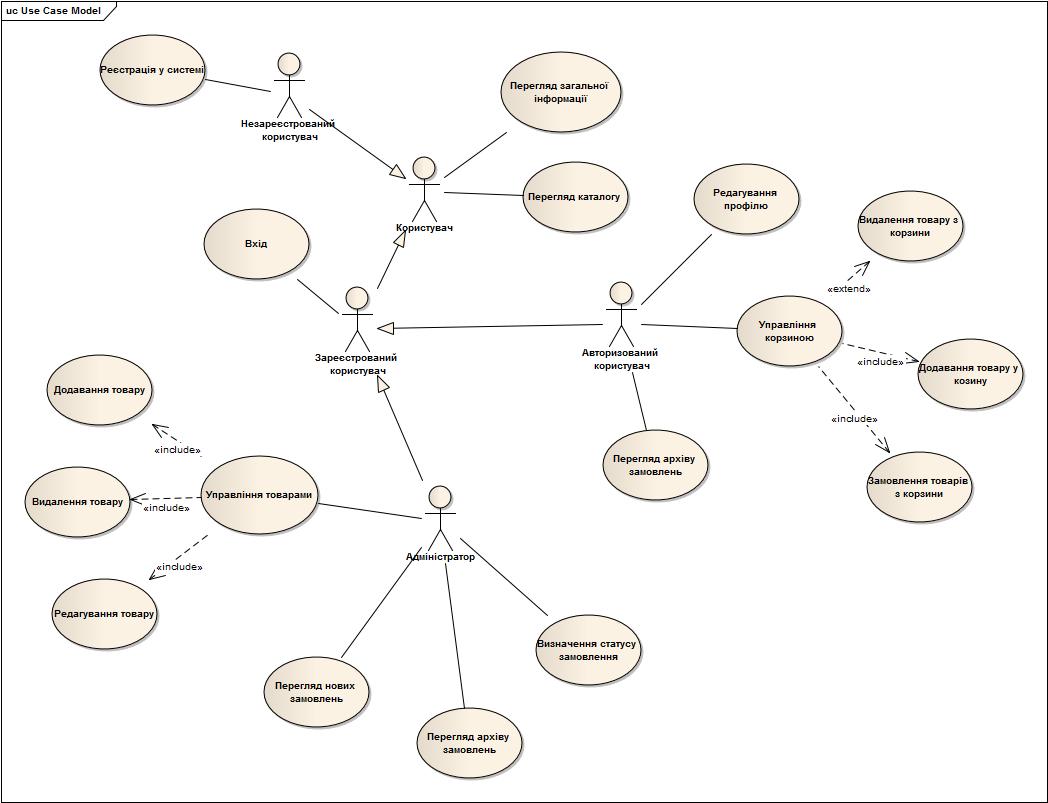
Побудуємо загальну схему прецедентів, яка для ролі користувача показує можливі послідовності дій акторів. На діаграмі зображені наступні актори:

1. Адміністратор – менеджер, який слідкує за базою товарів та керує новими замовленнями;
2. Авторизований користувач – клієнт, який має за мету придбати один або декілька з представлених магазином товарів;
3. Узагальнені актори: користувач, незареєстрований користувач, авторизований користувач.

Розглянемо можливості, які повинна задовольняти автоматизована система:

1. Користувач має можливість перегляду основної інформації та каталогу;
2. Незареєстрований користувач наслідує можливості користувача та має можливість реєстрації у системі;
3. Зареєстрований користувач наслідує можливості користувача та має можливості входу в систему;
4. Авторизований користувач використовує систему для перегляду наданих магазином товарів, а також для додавання товару у корзину або його видалення. Клієнт надається можливість переглядати архів своїх замовлень та редагувати свої контактні дані (для роботи кур’єрської служби). Наслідує можливості зареєстрованого користувача;
5. Адміністратор використовує систему для виконання нових замовлень та для управління товарами. Наслідує можливості зареєстрованого користувача;

На діаграмі використання зображено дії, які може виконувати кожний тип користувачів системи.



Основними вимогами до програмного забезпечення є:

- використання платформи Java EE;

- документованість коду;

- оптимальне використання фізичних ресурсів;

- використанні об’єктно-реляційного відображення для взаємодії програмного комплексу із джерелом даних;

- забезпечення безпечного використання даних.

**РОЗДІЛ 3**

**ОПИС ВИДІВ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ**

**3.1 Моделі та засоби проектування**

Під час проектування значну увагу було звернено на аналіз вимог та аналіз бізнес-процесів системи. Так, було розроблено табличний опис бізнес- процесів.

За допомогою моделі прецедентів, були визначені основні вимоги до функціональності системи. Для створення моделі прецедентів використано USE-CASE діаграму.

**3.2 Технології та засоби розробки системи**

Основною технологією, що використовувалась для розробки системи, є технологія Java EE. У її рамках було використано наступні технології:

- JSF – дану технологію було використано для моделювання інтерфейсу користувача, відображення моделі та взаємодії між інтерфейсом та моделлю системи;

- JPA – дану технологію було використано для доступу до бази даних, взаємодії між моделлю даних та бізнес-логікою системи. У рамках цієї технології було використано

- JPQL – дану технологію, точніше специфіковану мову було використано для реалізації запитів JPA.

- JMS – дану технологію було використано для обміну повідомлення між компонентами розроблювальної системи. У рамках цієї технології було використано технологію MDB.

- JAAS – дану технологію було розроблено для забезпечення авторизації та розділення доступу до ресурсів нашої системи.

- EJB – дану технологію було використано для забезпечення взаємодії із БД та безпеки даних, а також для побудови основної бізнес-логіки додатку.

- PrimeFaces та Ajax використано для оновлення даних без перезавантаження сторінки

Окрім Java EE ми використовували наступні технології:

- XHTML – дана технологія була використана для створення інтерфейсу користувача;

- CSS – дана технологія була використана для створення стилів інтерфейсу користувача

Засобами розробки були наступні програмні застосунки:

- Payara – даний застосунок було використано для розгортання системи, створення пулу та ресурсу для роботи з базою даних, налаштування реалму безпеки та JMS ресурсів.

- MySQL Denwer – локальний сервер даних, виеористаний для зберігання даних;

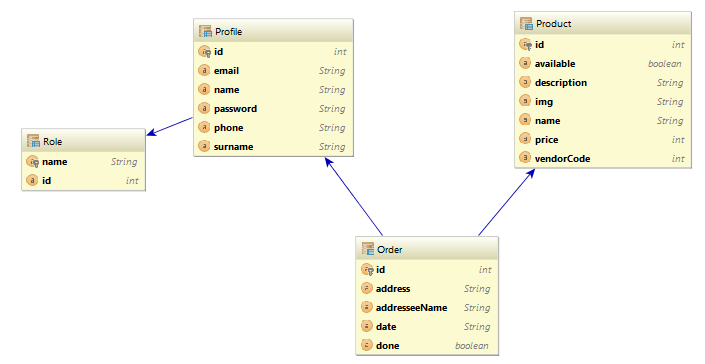
- Intellij IDEA – даний за стосунок було як середу розробки програмного додатку та для розгортання системи на сервері.

**3.3 Інформаційне забезпечення системи**

Для зберігання даних було використано реляційну базу даних MySQL. Окрім цього, для взаємодії програмного комплексу із джерелом даних було використано технології JPA та JPQL.

Для допоміжних ресурсів - дескриптору розгортання, опису залежностей, опису використання даних, опису даних реалму безпуки і т.д. було використано технологію XML.

Нижче відображена структура бази даних:



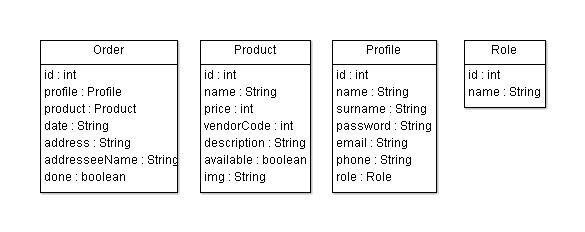
Для обміну данини між контролерами та моделлю система використовує технологію JPA.

**3.4 Програмне забезпечення**

При розробці системи ми використовували архітектурний шаблон MVC. Як основний засів для побудови системи, що повинна базуватися на даній архітектурі, ми використали технологію JSF.

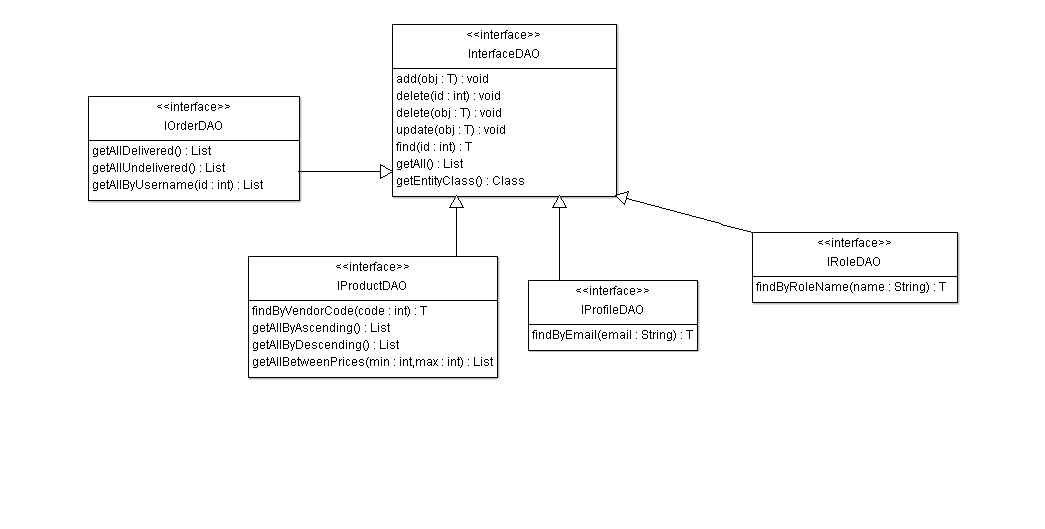
Нижче наведено детальну специфікацію класів системи.

Наведемо специфікацію класів для представлення бізнес-сутностей:

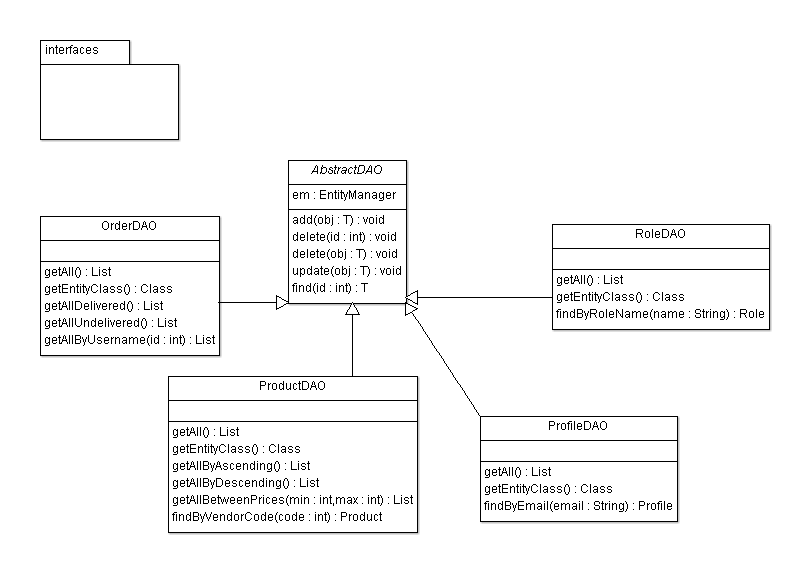


Кожний клас має get, set методи для кожного поля і реалізує equals, hashCode методи.

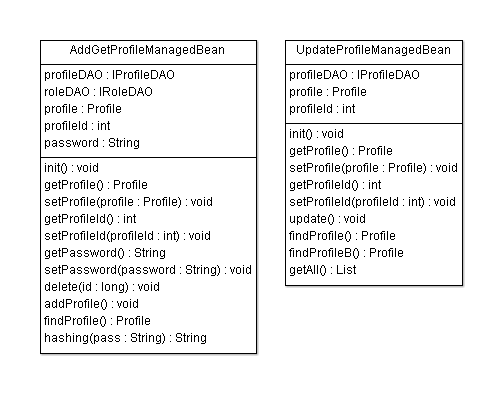
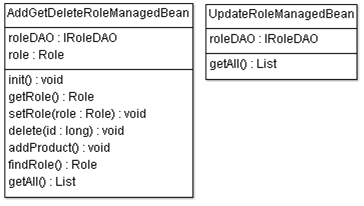
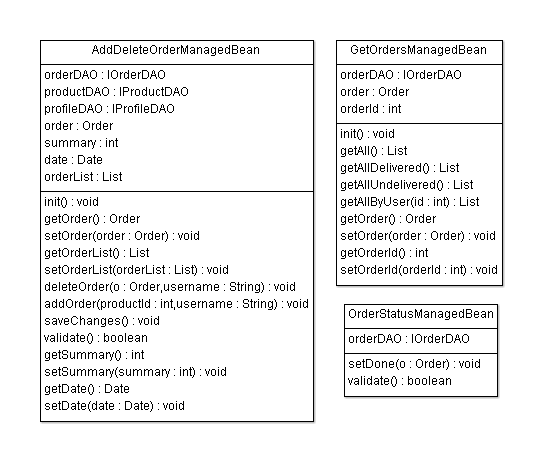
На діаграмі нижче зображено специфікацію інтерфейсів для об’єктів доступу до даних. Маємо узагальнений інтерфейс, який містить загальні дії з сутностями та інтерфейси з індивідуальним набором методів, необхідних саме для цієї сутності.



На діаграмі нижче зображено специфікацію класів, які реалізують інтерфейси доступу до даних. Маємо абстрактний клас з generic матодами, який реалізує загальний інтерфейс роботи з сутностями. Цей абстрактний клас наслідують усі загальні методи для роботи з сутностями. Крім цього, кожне DAO реалізує інтерфейс з методами, специфічними для кожної сутності.



Наведемо специфікацію класів (Manage Bean), що відповідають за бізнес-логіку системи





**РОЗДІЛ 4**

**ОПИС СИСТЕМИ**

**4.1 Опис структури системи**

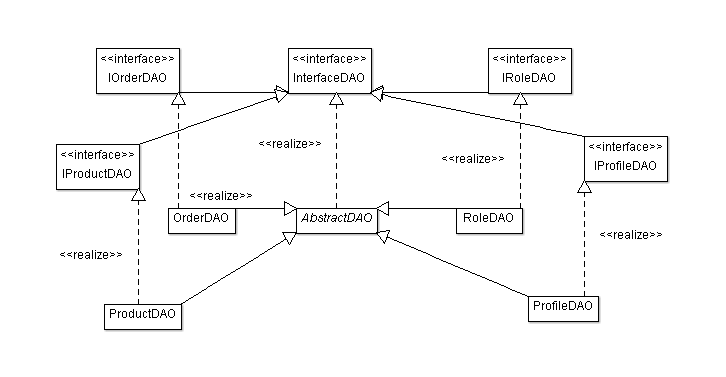
Розроблювальна система базується на архітектурі MVC, отже можна виділити три основні підсистеми:

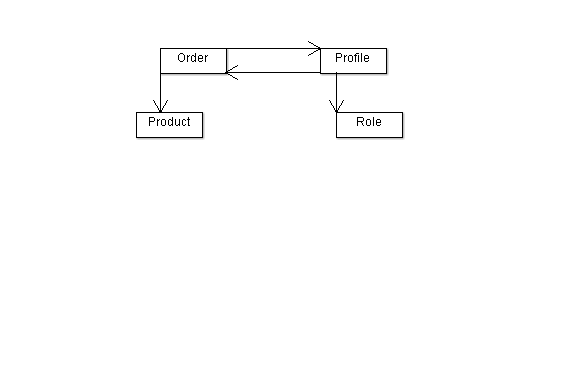
1. Підсистема “Робота з даними”;

2. Підсистема “Взаємодія користувача з системою”;

3. Підсистема “Інтерфейс користувача”.

Підсистема «Робота з даними» (схема нижче) забезпечує роботу системи з базою даних, виконує роботу по зміні та пошуку даних. До підсистеми входять сутності Role, Profile, Product, Order. Кожній сутності відповідає клас DAO, який взаємодіє з базою даних. DAO використовуються як EJB компоненти.





Підсистема “Взаємодія користувача з системою” відповідає за взаємодію користувача з системою для доступу до даних, інакше кажучи, вона в необхідний момент активує підсистему “Робота з даними” та робить вказані користувачем дії. Підсистема включає в себе Manage Beans, які включаються у JSF сторінку та забезпечують її роботу. Manage Bean має у собі головну бізнес-логіку системи.

До цієї системи також можна віднести jQuery валідацію даних на стороні клієнта.

Підсистема “Інтерфейс користувача”.

Дана підсистема відповідає за представлення даних користувачу, та містить файли, що відповідають за відображення отриманих даних - XHTML сторінки, CSS, JS файли.

**4.2 Опис функціонування системи**

Для кожної з вказаних вище підсистеми визначимо функції, які вона надає і, тим самим, автоматизує:

Підсистема “Робота з даними”:

* Вибірка даних із БД
* Видалення даних із БД
* Додавання даних у БД
* Редагування даних у БД

Підсистема “Взаємодія користувача з системою”:

* Виконання запитів користувача (видалена, додавання, редагування)
* Формування відповіді
* Отримання даних, наданих користувачем

Підсистема “Інтерфейс користувача”:

* Надання інтерфейсу
* Валідація даних