



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №2
Технології розроблення програмного забезпечення
ДІАГРАМА ВАРІАНТІВ ВИКОРИСТАННЯ. СЦЕНАРІЙ
ВАРІАНТІВ ВИКОРИСТАННЯ. ДІАГРАМИ UML. ДІАГРАМИ
КЛАСІВ. КОНЦЕПТУАЛЬНА МОДЕЛЬ СИСТЕМИ
Е-mail клієнт

Виконав

студент групи ІА–22:

Артамонов Дмитро

Перевірив:

Мягкий Михайло Юрійович

Київ 2024

Зміст

Хід роботи.....	3
Теоретичні відомості	4
Побудуємо схему прецедентів.....	5
Оберемо 3 прецеденти і напишемо для них сценарії використання	5
Побудуємо діаграму класів (рисунок 3, рисунок 4)	8
Висновок	9

Тема: Діаграма варіантів використання. Сценарії варіантів використання.
Діаграми UML. Діаграми класів. Концептуальна модель системи

Мета: Проаналізувати тему, намалювати схему прецеденту, діаграму класів,
розробити основні класи і структуру бази

Хід роботи

..15 E-mail клієнт (singleton, builder, decorator, template method, interpreter, SOA)

Поштовий клієнт повинен нагадувати функціонал поштових програм Mozilla Thunderbird, The Bat і т.д. Він повинен сприймати і коректно обробляти

pop3/smtp/imap протоколи, мати функції автонастройки основних поштових провайдерів для України (gmail, ukr.net, i.ua), розділяти повідомлення на папки/категорії/важливість, зберігати чернетки незавершених повідомлень, прикріплювати і обробляти прикріплені файли.

Теоретичні відомості

У 2 лабораторній роботі розглядається використання діаграм UML для моделювання різних аспектів системи. Ключовими елементами є діаграми варіантів використання (Use Case Diagrams), які допомагають визначити основні сценарії взаємодії користувачів із системою.

1. UML (Unified Modeling Language) – це універсальна мова візуального моделювання, призначена для опису структури і поведінки програмних систем. Вона дозволяє створювати діаграми для різних аспектів системи, забезпечуючи чітке та наочне уявлення про її роботу. UML використовується на всіх етапах розробки: від аналізу вимог до тестування.
2. Діаграми варіантів використання (Use Case Diagrams) – це інструмент для моделювання функціональності системи з точки зору користувача. Вони відображають, як зовнішні актори (користувачі, інші системи) взаємодіють із системою для досягнення своїх цілей. Основні елементи діаграми: актори, варіанти використання (use cases) і зв'язки між ними.
3. Діаграма класів (Class Diagrams) – це інший важливий інструмент UML, що моделює структуру системи на рівні класів і їх взаємодій. Вона відображає класи, їх атрибути, методи, а також зв'язки між ними (асоціації, агрегації, композиції). Ця діаграма є основою для реалізації об'єктно-орієнтованого програмування, оскільки дозволяє чітко визначити структуру об'єктів у системі.
4. Концептуальна модель системи – це початковий етап побудови моделі, який допомагає визначити ключові компоненти системи, їх взаємодії та ролі. Вона може містити як діаграми варіантів використання, так і діаграми класів, забезпечуючи базу для подальшого аналізу та реалізації.
5. Репозиторій – це шаблон проектування, який забезпечує абстракцію над операціями з базою даних. Він дозволяє взаємодіяти з базою даних через інтерфейси, що спрощує обслуговування та тестування коду. Репозиторій приховує деталі доступу до даних, що робить код більш організованим і підтримуваним.

Побудуємо схему прецедентів. Схема зображена на рисунку 1.

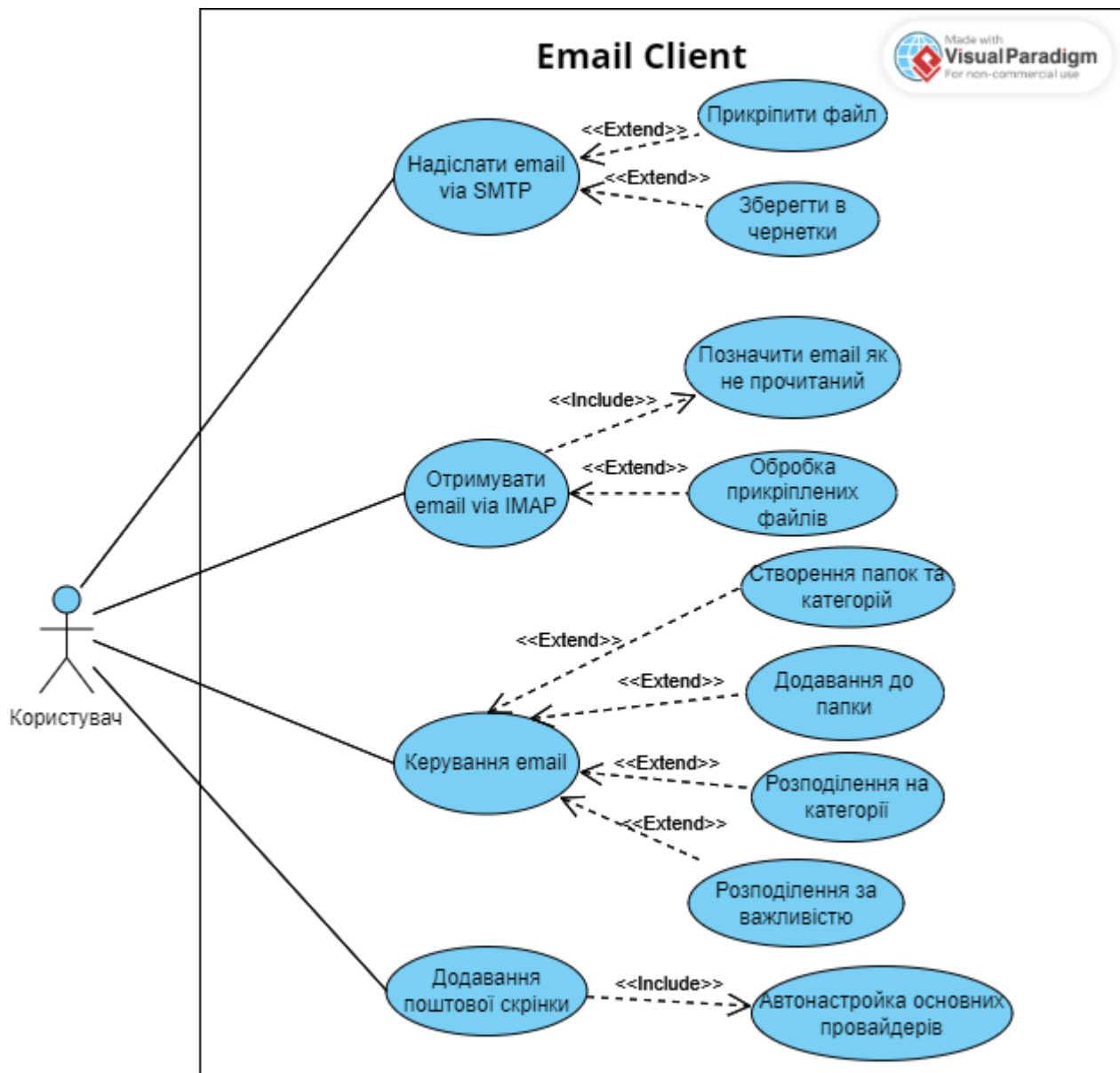


Рисунок 1. – Схема прецеденту

Оберемо 3 прецеденти і напишемо для них сценарії використання

Сценарій використання 1: Відправка email

Актори: Користувач.

Опис: Користувач з поштового клієнта надсилає електронного листа.

Основний хід подій:

1. Користувач за необхідності додає поштову скриньку.
2. Користувач натискає кнопку «Створити email».
3. Вказує адресу отримувача та поштову скриньку з якої буде відправлено email.
4. Користувач вказує Тему листа та інші корисні дані.
5. Система валідує введені дані та передає листа на поштовий сервер.
6. Надісланий лист поміщається в папку «Відправлені».

Виключення: Якщо введенні дані некоректні(наприклад, не вказано адресу отримувача чи відправника, не заповнено поле «Тема») система сповістить про помилку.

Сценарій використання 2: Додавання email до папки

Актори: Користувач.

Опис: Цей сценарій описує процес додавання отриманого email до папки

Основний хід подій:

1. Користувач обирає опцію "Всі листи".
2. Обирає листи якими він хоче керувати.
3. За потреби створює папку.
4. Обирає потрібну папку і додає листи.

Виключення: Якщо поштовий клієнт не може отримати доступ до списку папок (наприклад, через помилку доступу до сервера або відсутність підключення до інтернету), система повідомляє про це користувача.

Сценарій використання 3: Додавання поштової скриньки

Актори: Користувач.

Опис: Цей сценарій описує процес додавання нової поштової скриньки в поштовий клієнт.

Основний хід подій:

1. Користувач відкриває налаштування поштового клієнта і обирає "Додати поштову скриньку".
2. Вводить свою електронну адресу.
3. Поштовий клієнт автоматично налаштовує поштові скриньки для одного з провайдерів (gmail, ukr.net, i.ua).
4. Користувач вводить пароль і підтверджує додавання.
5. Поштовий клієнт перевіряє правильність підключення.

Виключення: При неправильному введенні облікових даних система повідомить про це користувача.

Побудуємо структуру бази даних (рисунок 2)

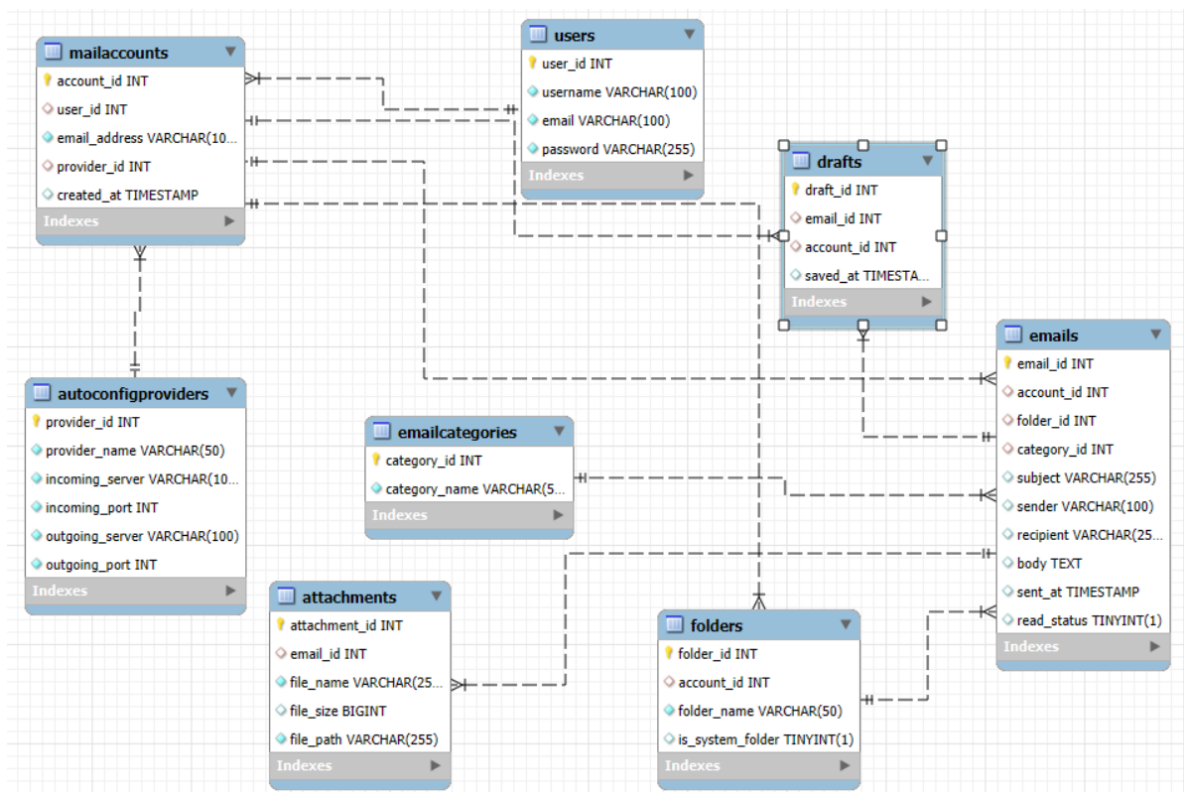


Рисунок 2. Структура бази даних

Побудуємо діаграму класів (рисунок 3, рисунок 4)

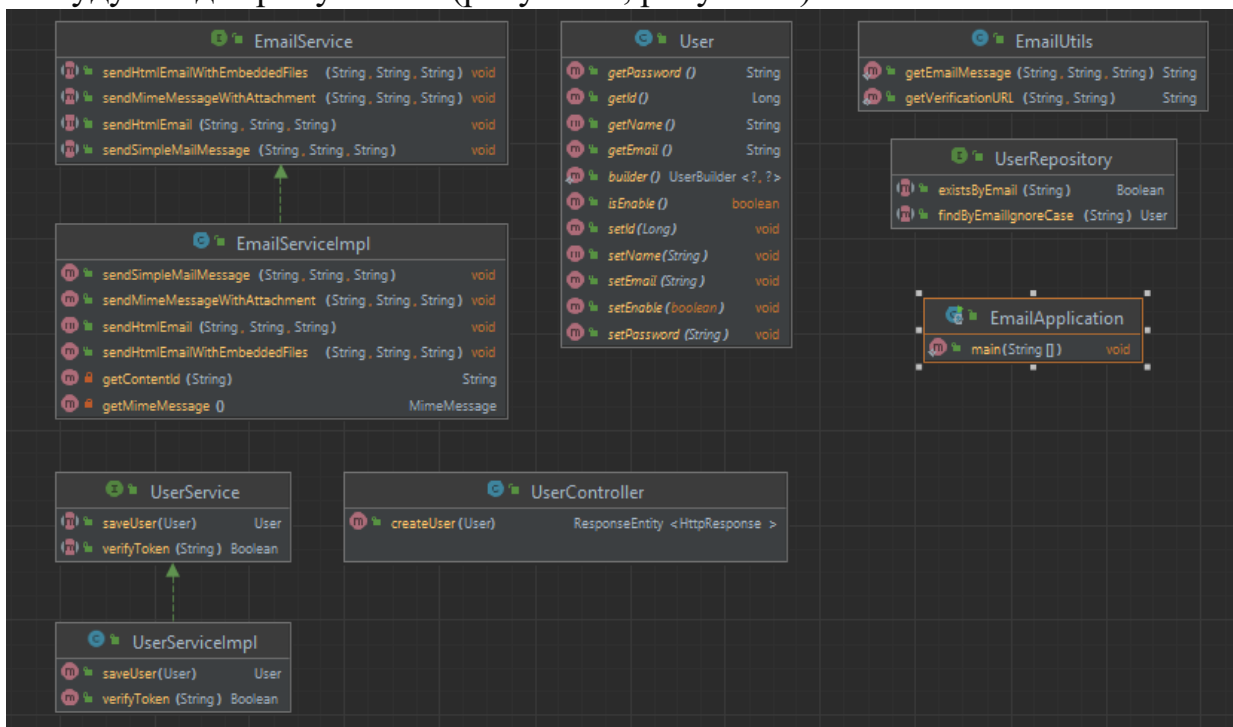


Рисунок 3. Діаграма реалізованих класів

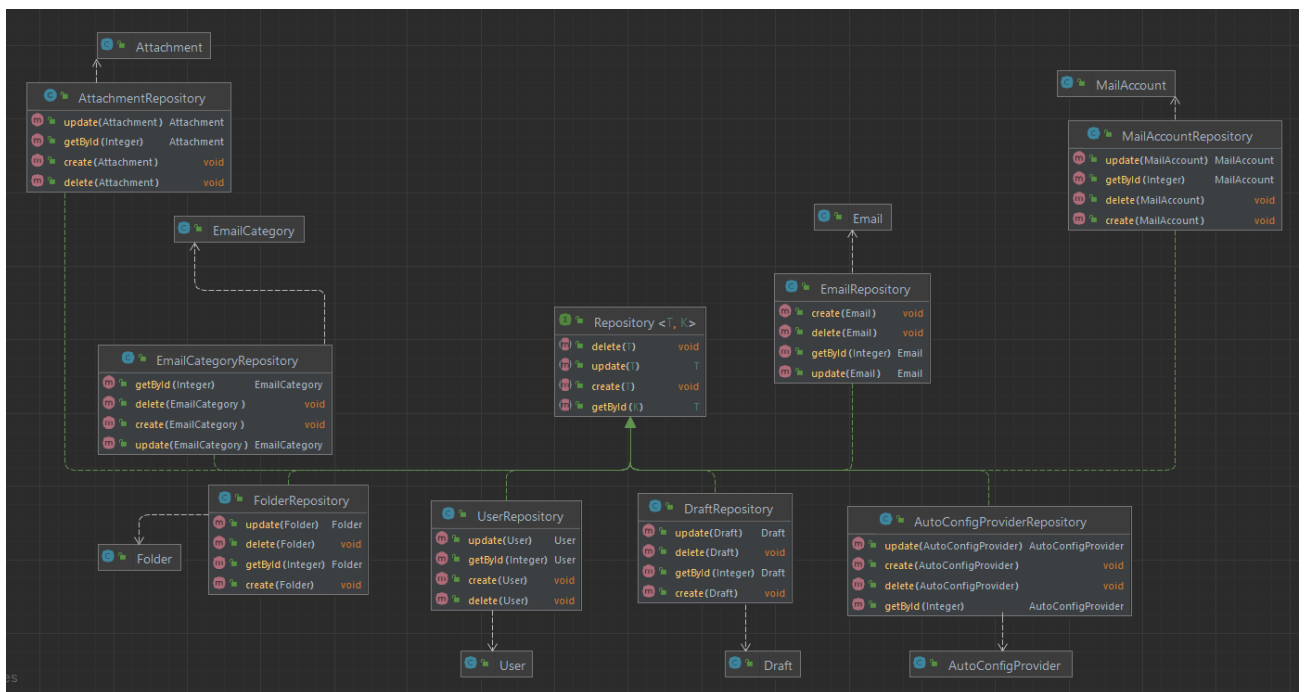


Рисунок 4. Класи даних, які реалізують шаблон Репозиторію

Посилання на GitHub: <https://github.com/qjsoq/trpz>

Висновок: В даній лабораторній роботі я проаналізував тему, намалював схему прецедентів, діаграму класів, розробив основні класи і структуру бази даних.