



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

«Діаграми прецедентів, класів, структура системи бази даних»

Варіант 23

Виконала студентка групи ІА–13:

Сиваченко Дар'я

Перевірів :

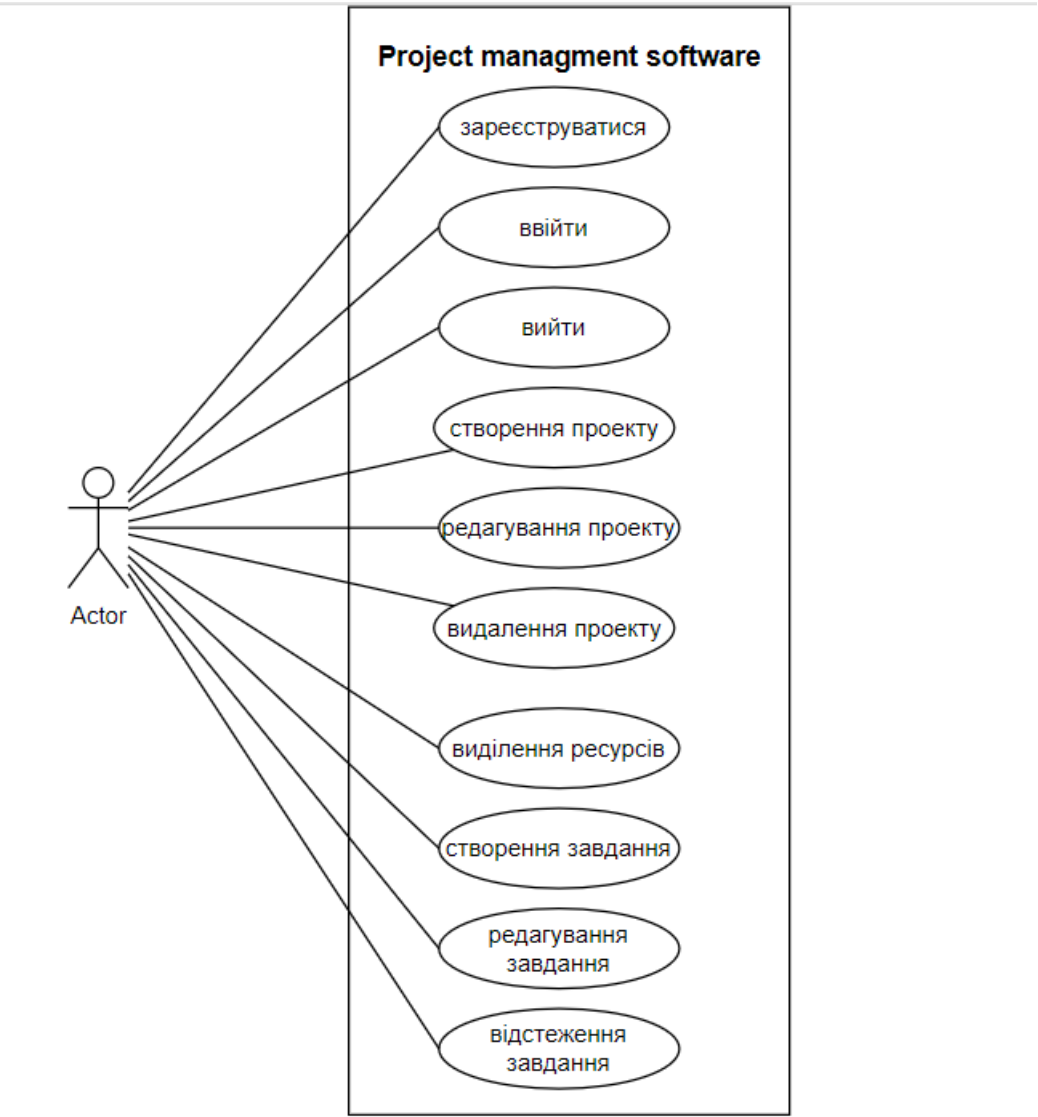
Мягкий Михайло Юрійович

Київ 2023

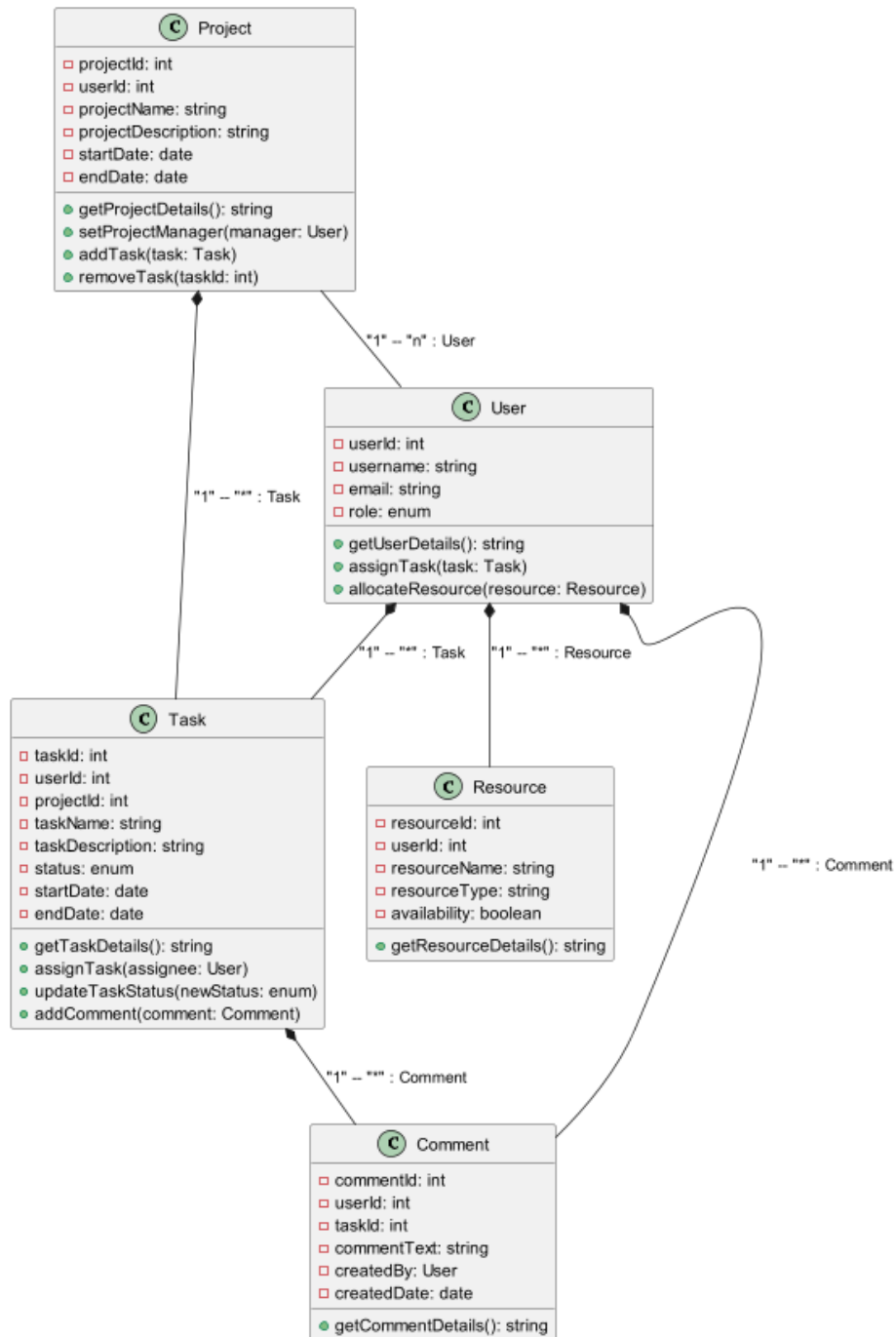
Тема: Project Management software

Хід роботи

Діаграма Прецедентів



Діаграма Класів



File labone.puml (код, який відображає діаграму класів) середовище - IntelliJ IDEA

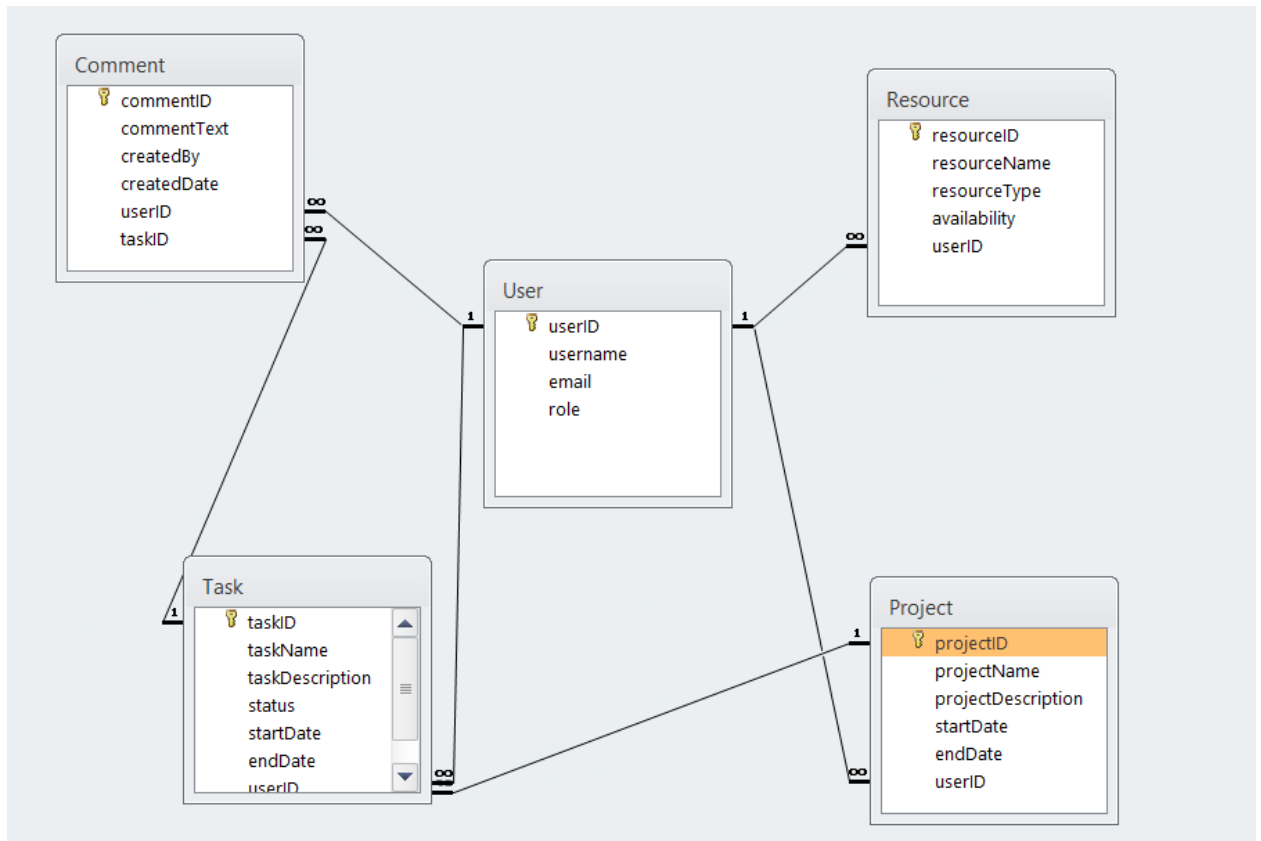
```
@startuml
class Project{
- projectId: int
- userId: int
```

```

- projectName: string
- projectDescription: string
- startDate: date
- endDate: date
+ getProjectDetails(): string
+ setProjectManager(manager: User)
+ addTask(task: Task)
+ removeTask(taskId: int)
}
class Task{
- taskId: int
- userId: int
- projectId: int
- taskName: string
- taskDescription: string
- status: enum
- startDate: date
- endDate: date
+ getTaskDetails(): string
+ assignTask(assignee: User)
+ updateTaskStatus(newStatus: enum)
+ addComment(comment: Comment)
}
class User{
- userId: int
- username: string
- email: string
- role: enum
+ getUserDetails(): string
+ assignTask(task: Task)
+ allocateResource(resource: Resource)
}
class Resource{
- resourceId: int
- userId: int
- resourceName: string
- resourceType: string
- availability: boolean
+ getResourceDetails(): string
}
class Comment{
- commentId: int
- userId: int
- taskId: int
- commentText: string
- createdBy: User
- createdAt: date
+ getCommentDetails(): string
}
Project -- User : "1" -- "n" : User
Project *-- Task : "1" -- "*" : Task
User *-- Resource : "1" -- "*" : Resource
Task *-- Comment : "1" -- "*" : Comment
User *-- Comment : "1" -- "*" : Comment
User *-- Task : "1" -- "*" : Task
@enduml

```

Діаграма Баз даних



Опис трьох прецедентів:

1. Користувач входить в систему

Актор: Користувач

Опис: Користувач вводить ім'я користувача та пароль для входу в систему

Результат: Система перевіряє та надає доступ до користування програмою.

2. Створення проекту

Актор: Користувач

Опис: Користувач натискає кнопку створити, заповнює необхідні поля

Результат: Новий проект

3. Редагування завдання

Актор: Користувач

Опис: Користувач натискає кнопку редагувати та редагує інформацію про завдання або ж змінює його статус

Результат: Відредагване завдання.

Зв'язки між класами:

- Відношення між Project та User вказує, що один проект може мати безліч користувачів та один користувач може мати багато проектів.
- Відношення між Project та Task вказує, що один проект містить багато завдань, але кожне з завдань відноситься до окремого проекту
- Відношення між User та Resource вказує, що один користувач може виділяти безліч ресурсів, але кожен з них відноситься до одного користувача.
- Відношення між Task та Comment вказує, що одне завдання може мати безліч коментарів, але кожен з них відноситься до одного завдання.
- Відношення між User та Comment вказує, що користувач може написати багато коментарів, але кожен з них відноситься до одного користувача
- Відношення між User та Task вказує, що користувач може мати безліч завдань, але кожне з них відноситься до одного користувача.

"1 -- n" вказує на точну кількість (один до багатьох)

"1 -- *" означає, що кількість об'єктів може бути будь-якою (один до багатьох або нуль до багатьох)

Висновок

Виконуючи дану лабораторну роботу, я дізналася про типи діаграм, а також закріпила ці знання, як теоретично, так і практично.