

1 ПРОЕКТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ

1.1 КОНЦЕПТУАЛЬНЕ ПРОЕКТУВАННЯ

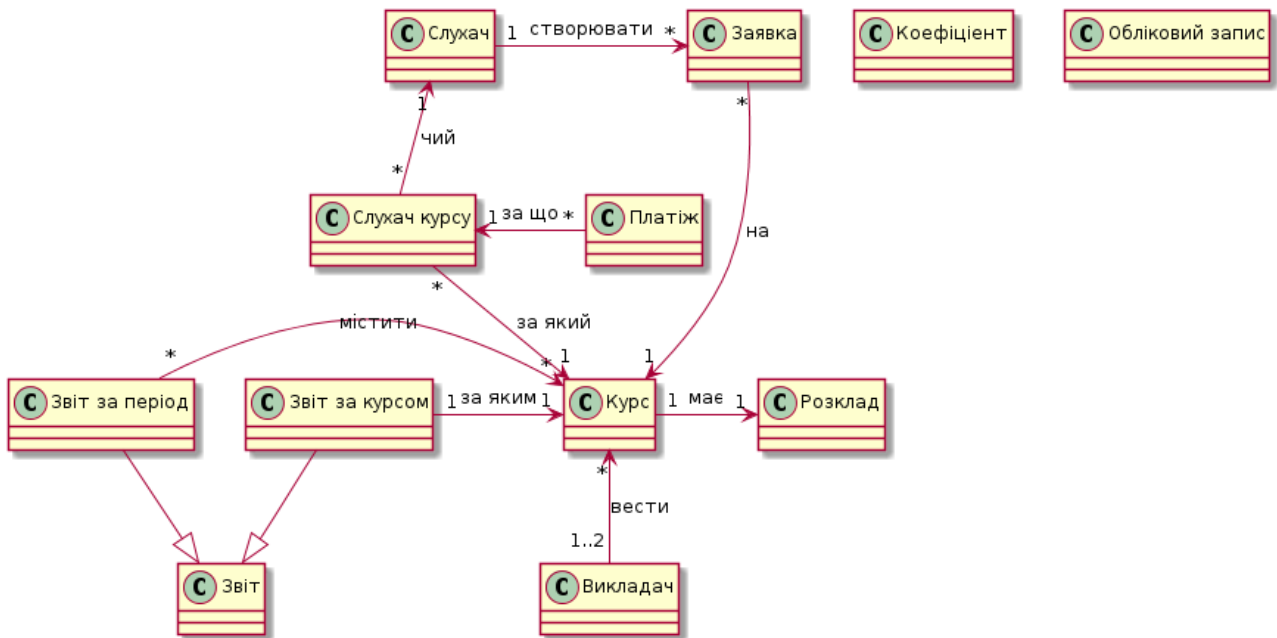


Рис. 1. Діаграма концептуальних класів

1.2 ЛОГІЧНЕ ПРОЕКТУВАННЯ

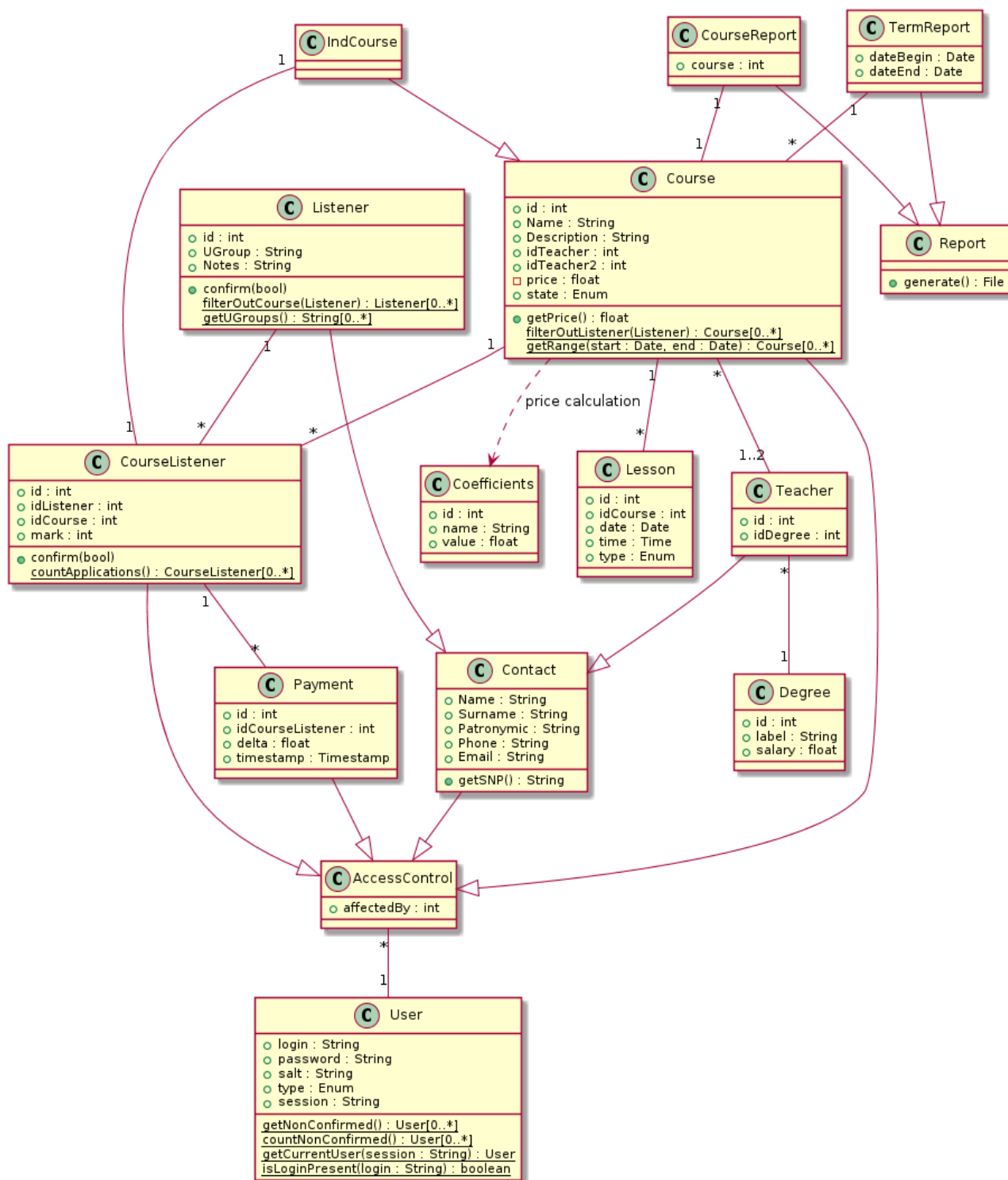


Рис. 2. Діаграма програмних класів

1.3 БАЗА ДАНИХ

Таблиця 1.1 — Опис структури таблиці "Слухачі" (listeners)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Ідентифікатор	idListener	int(6)	Ні	A_I
	Ім'я	Name	varchar(30)	Так	
	Прізвище	Surname	varchar(30)	Ні	
	По батькові	Patronymic	varchar(30)	Так	
	Університетська група	UGroup	varchar(6)	Так	
	Телефон	Phone	varchar(20)	Так	
	E-mail	Email	varchar(40)	Так	
FK	Ким змінено	affectedBy	varchar(32)	Ні	

Таблиця 1.2 — Опис структури таблиці "Курси" (courses)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Ідентифікатор	idCourse	int(6)	Ні	A_I
	Назва	Name	varchar(50)	Ні	
	Опис	Description	varchar(400)	Так	
	Ознака індивідуального курсу	isIndividual	bit(1)	Ні	
FK	Викладач	idTeacher	int(6)	Так	0..2 (Йде набір, набрано, завершений)
FK	Другий викладач	idTeacher2	int(6)	Так	
	Ціна	Price	decimal(10,2)	Так	
	Стан	state	tinyint(1)	Ні	
FK	Ким змінено	affectedBy	varchar(32)	Ні	

Таблиця 1.3 — Опис структури таблиці "Слухачі курсу" (Course_Listeners)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Ідентифікатор	idCL	int(6)	Ні	A_I
FK	Курс	idCourse	int(6)	Ні	
FK	Слухач	idListener	int(6)	Ні	
	Оцінка	mark	tinyint(3)	Так	
FK	Ким змінено	affectedBy	varchar(32)	Ні	

Таблиця 1.4 — Опис структури таблиці "Платежі" (payments)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Ідентифікатор	idPayment	int(6)	Ні	A_I CURRENT_TIMESTAMP
FK	Слухач курсу	idCL	int(6)	Ні	
	Кошти	delta	int(6)	Ні	
	Мітка часу	timestamp	timestamp	Ні	

Таблиця 1.5 — Опис структури таблиці "Викладачі" (teachers)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Ідентифікатор	idTeacher	int(6)	Ні	A_I
	Ім'я	Name	varchar(30)	Так	
	Прізвище	Surname	varchar(30)	Ні	
	По батькові	Patronymic	varchar(30)	Так	
	Телефон	Phone	varchar(20)	Так	
	E-mail	Email	varchar(40)	Так	
FK	Ким змінено	affectedBy	varchar(32)	Ні	
FK	Вчений ступінь	degree	tinyint(2)	Так	

Таблиця 1.6 — Опис структури таблиці "Розцінки" (prices)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Ідентифікатор	degree	tinyint(2)	Ні	A_I
	Вчений ступінь	deglab	varchar(30)	Ні	
	Зарплата	salary	decimal(10,2)	Ні	

Таблиця 1.7 — Опис структури таблиці "Коефіцієнти" (coefficients)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Назва	name	varchar(64)	Ні	
	Мітка	label	varchar(128)	Так	
	Значення	value	float	Так	

Таблиця 1.8 — Опис структури таблиці "Заняття" (lessons)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
PK	Ідентифікатор	idLesson	int(11)	Ні	A_I
FK	Курс	idCourse	int(11)	Ні	
	Дата	date	date	Так	
	Час початку	time	time	Ні	
	Тип	type	tinyint(1)	Ні	
					0..1 (Лекція, Практика)

Таблиця 1.9 — Опис структури таблиці "Користувачі" (users)

Ключ	Назва	Ім'я поля	Тип	NULL	Дод.
РК	Логін	login	varchar(32)	Ні	0..2 (Оператор, Адміністратор, Переглядач)
	Хеш паролю	password	text	Ні	
	Сіль	salt	text	Ні	
	Тип	type	tinyint(1)	Ні	
	Ключ сесії	sessionid	text	Так	

1.4 ОЦІНКА АЛГОРИТМІЧНОЇ СКЛАДНОСТІ

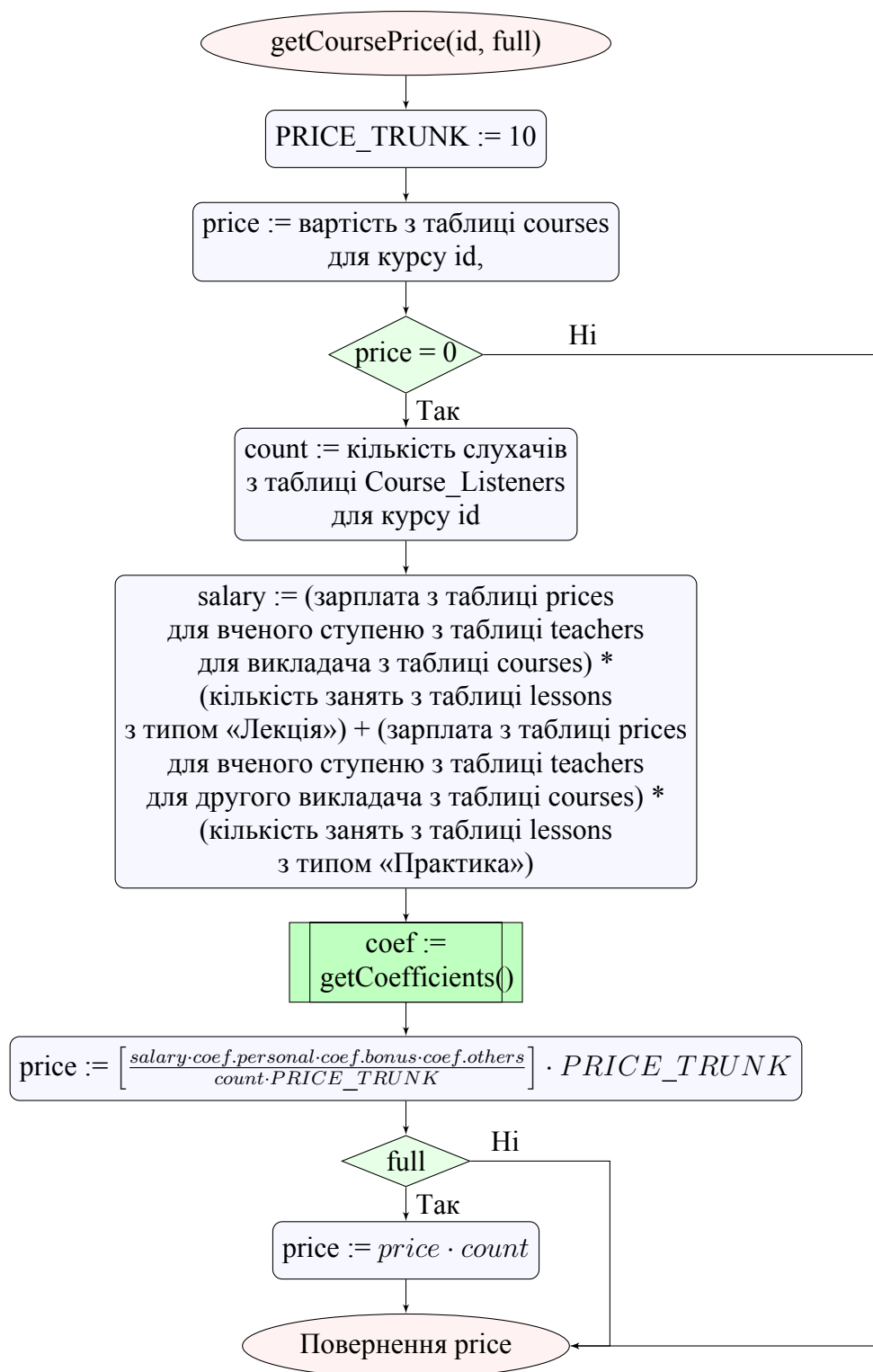


Рис. 3. Алгоритм розрахунку вартості курсу

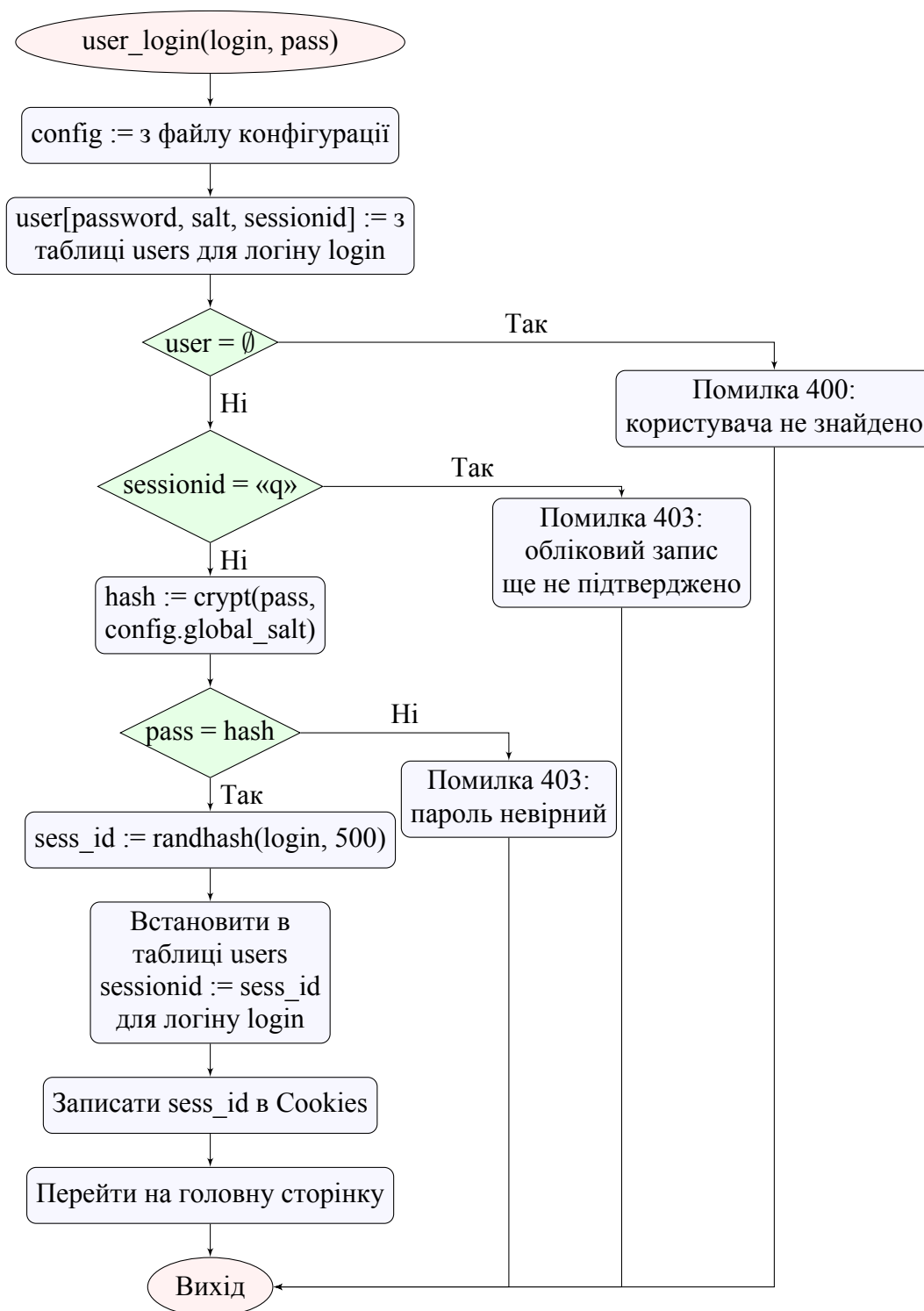


Рис. 4. Алгоритм аутентифікації та авторизації

На рис. 3 зазначено складну операцію розрахунку вартості за даними з БД. Вона реалізується наступним SQL-запитом:

```

SELECT ifnull(s1,0)*ifnull(c1,0)+ifnull(s2,0)*ifnull(c2,0) FROM ((
  SELECT salary as s1 FROM prices WHERE degree IN (
    SELECT degree FROM teachers WHERE idTeacher IN (
      SELECT idTeacher FROM courses WHERE idCourse=?
    )
  )
)

```

```

))) s1 LEFT OUTER JOIN (
  SELECT salary as s2 FROM prices WHERE degree IN(
    SELECT degree FROM teachers WHERE idTeacher IN (
      SELECT idTeacher2 FROM courses WHERE idCourse=63
    ))) s2 ON 1=1) LEFT OUTER JOIN ((
  SELECT count(*) as c1 FROM lessons WHERE idCourse=? AND type=0
) c1 ON 1=1 LEFT OUTER JOIN (
  SELECT count(*) as c2 FROM lessons WHERE idCourse=? AND type=1
) c2) ON 1=1;

```

План виконання запиту:

1. Вибірка вмісту таблиці courses
2. Проекція кортежів за значенням атрибуту idCourse
3. Проекція атрибутів за атрибутом idCourse
4. Вибірка вмісту таблиці teachers
5. Проекція кортежів за значенням атрибуту idTeacher
6. Проекція атрибутів за атрибутом degree
7. Вибірка вмісту таблиці prices
8. Проекція кортежів за значенням атрибуту degree
9. Проекція атрибутів за атрибутом salary
10. Використання вибірки з п. 1.
11. Проекція кортежів за значенням атрибуту idCourse
12. Проекція атрибутів за атрибутом idCourse
13. Використання вибірки з п. 4.
14. Проекція кортежів за значенням атрибуту idTeacher
15. Проекція атрибутів за атрибутом degree
16. Використання вибірки з п. 7.
17. Проекція кортежів за значенням атрибуту degree
18. Проекція атрибутів за атрибутом salary
19. Вибірка вмісту таблиці lessons
20. Проекція кортежів за значенням атрибуту idCourse
21. Проекція кортежів за значенням атрибуту type
22. Підрахунок кортежів
23. Використання проекції з п. 20.
24. Проекція кортежів за значенням атрибуту type

25. Підрахунок кортежів
26. З'єднання результатів пп. 9 та 18.
27. З'єднання результатів пп. 22 та 25.
28. З'єднання результатів пп. 26 та 27.
29. Розрахунок математичного виразу за результатом п. 28.

Операції проєкції кортежів у пп. 2, 3, 5, 6, 8, 9, 11, 12, 14, 15, 17 та 18 повинні повертати один кортеж. Пошук у таблицях `courses` та `teachers` відбувається за первинним ключем, а таблиця `rgices` складається з кількох кортежів. Для проєкцій з пп. 20, 21 та 24 доцільно створити у таблиці `lessons` індекс за атрибутом `idCourse`. За атрибутом `type` індекс недоцільний, оскільки проєкції за ним відбуваються не з усієї таблиці, а можливих значення тільки два. SQL-запит для створення індексу:

```
CREATE INDEX course_of_lesson ON lessons (idCourse);
```

Циклів в алгоритмах на рис. 3 та 4 немає, тож їх складність — $O(N)$.

1.5 ОПИС ЗОВНІШНІХ ІНТЕРФЕЙСІВ

Інтерфейс користувача побудовано із використанням бібліотеки Kendo UI, дизайн визначається наявними для неї темами оформлення. Доступні самостійні концептуальні класи присутні у головному меню та розташовані за частотою використання: найактивніша робота проводиться зі слухачами та курсами, викладачі та користувачі заповнюються на початку роботи і потім змінюються рідко, звіти здаються рідко, розцінки та коефіцієнти змінюються у виключних випадках. В кінці головного меню розміщено кнопку виходу з системи. Екрану авторизації є окремим та лаконічним, містить лише форму та кнопку перемикання форм реєстрації та входу.

Для роботи серверної частини системи потрібен веб-сервер Apache 2.x, інтерпретатор PHP5 не нижче 5.4, СКБД MySQL або MariaDB 5. Для користування клієнтською частиною потрібен web-браузер із підтримкою EcmaScript 5; тестування проводиться у поточних версіях браузерів Mozilla Firefox та Chromium для десктопу.

Система може працювати у межах однієї машини, через локальну мережу та через мережу Інтернет, в залежності від мережевих підключень та налаштувань машини, на якій встановлено серверну частину. Підключення, відповідно, може здійснюватись будь-яким доступним дротовим або бездротовим каналом підключення до локальної мережі або мережі Інтернет: Ethernet, Wi-Fi, ADSL, HSPA, Dial-up та ін. Оскільки звіти генеруються у форматі PDF, для їх перегляду потрібна програма — переглядач PDF: Mozilla Firefox, Google Chrome,

Adobe Reader, Foxit Reader тощо. Для друку потрібен принтер, підключений безпосередньо до машини, на якій відкрито файл звіту, або доступний для неї через мережу.