

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Факультет комп'ютерних наук, фізики та математики
Кафедра інформатики, програмної інженерії та економічної кібернетики

ПРОЄКТУВАННЯ ТА РОЗРОБЛЕННЯ СЕРВІСНОЇ АРХІТЕКТУРИ
УПРАВЛІННЯ БІЗНЕС-ПРОЦЕСАМИ УНІВЕРСИТЕТУ.
ЕЛЕКТРОННИЙ ЖУРНАЛ

Кваліфікаційна робота (проєкт)
на здобуття ступеня вищої освіти “магістр”

Виконав: студент 2 курсу 231М групи

Напряму підготовки: 122 Комп'ютерні науки

Освітньо-професійної програми

«Комп'ютерні науки» другого (магістерського)

Рівня вищої освіти Гунько Сергій Едуардович

Керівник: доктор педагогічних наук, професор

Співаковський Олександр Володимирович

Рецензент: кандидат педагогічних наук, доцентка

Єрмакова-Черченко Наталія Олександрівна.

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1. Технічне завдання	5
1.1 Дослідження існуючих програмних продуктів.....	5
1.2 Призначення, цілі створення інформаційної системи	8
1.3 Вимоги до інформаційної системи.....	9
РОЗДІЛ 2. Функціонально-орієнтоване проектування	11
2.1 Побудова контекстної діаграми	11
2.2 Побудова діаграм декомпозиції IDEF0	13
2.3 Інформаційний аналіз предметної області	19
2.4 Опис таблиць бази даних	23
РОЗДІЛ 3. Розробка клієнт-серверного додатка	27
3.1 Технологія «клієнт-сервер»	27
3.2 Засоби програмування.....	28
3.3 Програмна реалізація модуля	30
3.4 Програмна реалізація модуля «Керівник групи»	32
ВИСНОВКИ	36
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	37
Додаток А	40
Додаток Б	41
Додаток В	42

ВСТУП

Сучасні технології швидкими кроками впроваджуються в усі сфери людської діяльності, в тому числі та у сферу освіти. Рівень розвитку інформаційних ресурсів та характер їх використання значною мірою впливає на добробут суспільства й ефективність здійснення тієї чи іншої професійної діяльності. Головні позиції будь-якої організації визначаються в першу чергу її можливостями по грамотному використанню передових досягнень у сфері інформаційних технологій. Особливу роль при цьому відіграє сфера освіти, де формується інтелектуальний потенціал країни, що споживає і створює її інформаційні ресурси. Інформатизація — це комплекс робіт, спрямований на розробку, впровадження, супровід, розвиток і заміну у всіх сферах діяльності традиційних технологій на більш ефективні інформаційно-телекомунікаційні технології. Нині розвиток Веб-технологій вносить нові переваги в проектування і розробку таких систем. Сучасні засоби створення Веб-технологій можна порівняти та вони не поступаються за швидкістю і зручністю зі звичайними настільними додатками, будучи при цьому доступними в будь-який час і з будь-якої точки світу, що не вимагають спеціалізованого програмного забезпечення та операційної системи. Ці гідності нашоухують замовників, а, отже, і розробників, до створення саме додатків з Веб-інтерфейсом, як заміни настільним рішенням або локальним рішенням.

Херсонський державний університет вже довгі роки веде активну і досить успішну роботу в напрямку інформатизації навчального процесу для підвищення якості освіти.

Однією з систем, яка впроваджена, у зв'язку з її необхідністю використання, є електронний журнал успішності, який автоматизує процес контролю успішності, дублює записи традиційного, паперового журналу, захищаючи його від спотворень, дає можливість контролювати накопичення

оцінок з предметів і т.д.

Метою даної роботи є побудова моделі інформаційної системи «Електронний журнал» і розробка web-додатків для автоматизації процесу ведення журналу успішності і відвідуваності.

Об'єктом дослідження є бізнес-процес управління деканату Херсонського державного університету.

Предметом дослідження є Сервісна архітектура управління бізнес-процесами університету. Сервіс “Електронний Журнал”.

Метою дослідження є проектування сервісної архітектури управління бізнес-процесами університету та розробка сервісу “Електронний Журнал”.

В межах роботи:

- дано аналіз комерційних додатків та систем, які використовуються в інших університетах,
- дано аналіз структури електронного журналу,
- спроектована сервісна архітектура електронного журналу,
- дано формулювання завдання на проектування сервісу “Електронний журнал”.
- розроблено сервіс “ Електронний журнал ” для системи управління бізнес-процесами університету.

Розроблений сервіс може використовуватись по прямому призначенню.

РОЗДІЛ 1

ПОСТАНОВКА ЗАВДАННЯ

1.1 Дослідження існуючих програмних продуктів

На великому рівні є різні реалізації схожих систем. В інших країнах автоматизація навчального процесу стоїть на ще більш високому рівні, ніж в установах нашої країни. Напевно, пов'язано як і з загальним відставанням нашої країни в інформаційних технологіях, так і з широким розвитком в західних державах віддаленої освіти. Останнє, стимулює автоматизації дій в освіті, введенню web-інтерфейсів доступу до засобів автоматизації. Учень ВНЗ, який вчиться дистанційно, має можливість перевірити стан справ лише через інтернет, тому доступ до власного облікового запису через всевітню мережу обов'язковий для успішного введення віддаленої освіти.

В Україні ступінь розвитку дистанційної освіти тільки починає набирати обороти. Тому чисельність подібних сервісів ще не в такій мірі, як за кордоном. При цьому багато хто з установ, які вводять дистанційне навчання і сервіси успішності учнів, вважаються відділеннями закордонних інститутів і середніх навчальних закладів. До подібних реалізацій в Україні варто віднести програмне забезпечення контролю успішності в середніх школах, яке орієнтоване на повідомлення батьків про стан справ дітей. Такі рішення розробляються приватними компаніями. Так, у багатьох школах міста запроваджено систему електронних щоденників студентів, за допомогою яких, батьки студентів можуть здійснювати контроль за власними дітьми. Але майже всі з даних сервісів не дають доступу до даних по успішності підлітків через мережу Інтернет. Однак, не дивлячись на це, формування таких «електронних журналів» - не погана передумова більшого розвитку схожих систем в набагато більших масштабах. Найбільш вдалі реалізації системи «Електронний журнал»: www.antcol.ru/jur — Журнал

успішності студентів Міжнародного Коледжу Туризму (рис. 1.2.1).

Интересующий Вас месяц

Просмотр оценки

Предметы

Поведение в Обществе

Предмет	Сумма	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Общественно-педагогическая деятельность	Сумма: 5	4	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	
Активный отдых	Сумма: 5	5	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	
Поведение в обществе	Сумма: 5	5	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	
Курсы английского языка	Сумма: 7	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	
Поведение в обществе	Сумма: 5	5	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	

Рисунок 1.2.1 — Журнал успішності студентів Міжнародного Коледжу Туризму

- www.emsy.org — Незалежний онлайн-сервіс електронних журналів успішності, орієнтований на освітні установи країн СНД — школи, дошкільні установи, ліцеї, коледжі, технікуми, інститути, центри сертифікації та інші. Основна функція сервісу — організація і ведення журналів оцінок в мережі Інтернет (рис. 1.2.2).

		Математика	Информатика	Архитектура компьютера	БЖ	ЭТ	Английский язык	Посещаемость	Средняя ведомость	Студенты
Предмет: Математика, Преподаватель: Шевченко Н. М.										
Список темы/комментарии										
Страница 1 На странице: 20										
№	ФИО студента	Доступ	04/11	02/12	02/12	02/12	09/12	11/12	22/01	Средняя
			Кр	Э	Т	ЛБ	Т	Т	А	
1	Антонова Р. А.		4	4	5	4		5	4	4,33
2	Богданов А. В.									-
3	Волков Е. Н.									-
4	Воропылева О. Л.									-
5	Голубев В. П.		-/3	4		4	4	4	4	3,83
6	Демченко О. Л.									-
7	Зайцев А. С.			4	5	5	5		5	4,8
8	Иванченко В. Е.									-
9	Козлов А. В.		5	4		5	4		5	4,6
10	Кузнецова Т. В.		5	5	5	5			5	5
11	Марченко Т. В.									-
12	Морозова Н. Н.		5	5				5	5	5
13	Новиков П. Р.			3	3				3	3
14	Петров П. П.									-
15	Семенов И. В.		3	4	2		4		3	3,2
16	Соловьева В. А.		5	5			5	5	5	5
Средняя по колонке			4,21	4,18	4,14	4,33	4,29	4,5	4,31	4,25
Успеваемость %			100	100	-	100	-	-	100	100
Качество %			81	87	-	100	-	-	87	88
Подпись			+	+	+	+	+	+	+	

Рисунок 1.2.2 - Незалежний онлайн-сервіс електронних журналів

www.emsy.org

№	Предмет	1 сем.	2 сем.	3 сем.	4 сем.	5 сем.	6 сем.	7 сем.	8 сем.	9 сем.	10 сем.	Повторно	Итого
1	Алгебра	5	5	5	4	5	5	5	5	5	5		5
2	Английский язык	4	4	4	4	4	4	4	4	4	4		4
3	Биология	4	4	4	4	4	4	4	4	4	4		4
4	География	5	4	4	5	5	5	5	5	5	5		5
5	Геометрия	5	5	5	5	5	5	5	5	5	5	5	5
6	Информатика	4	5	5	4	5	5	5	5	5	5	5	5
7	Моторы	4	5	5	5	5	5	5	5	5	5	5	5
8	Музыка											5	5
9	Литература	5	5	5	4	5	5	5	5	5	5	5	5
10	Мат.	5	5	5	4	5	5	5	5	5	5		5
11	ОБЖ	4	4	5	5	5	5	5	5	5	5		5
12	Обществознание	4	4	4	4	4	4	4	4	4	4		4
13	Правоведение	4	4	4	4	4	4	4	4	4	4		4
14	Русский язык	5	5	5	5	5	5	5	5	5	5	5	5
15	Социология	4	4	4	4	4	4	4	4	4	4	4	4
16	Физкультура	4	4	4	4	4	4	4	4	4	4	4	4
17	Французский язык	5	4	4	4	4	4	4	4	4	4	4	4
18	Химия	5	4	4	4	5	5	5	5	5	5	5	5

Рисунок 1.2.3 - безкоштовна шкільна освітня мережа www.dnevnik.ru

Вивчивши наявні рішення згідно з темою, можна зробити висновок, що надана проблема вважається не до кінця вирішеною, що дає великі можливості для майбутнього дослідження і поліпшення. Напевно, в особливості актуально для нашої країни в цілому. Здійснення web-обслуговування електронного журналу має важливе практичне значення для учнів, тому що дозволить їм стежити за своїм станом справ в навчальному закладі в будь-який час і з будь-якого місця з доступністю всесвітньої павутини. Так само полегшує контроль батьків над своїми дітьми (успішність і відвідуваність).

1.2 Призначення, цілі створення інформаційної системи

Розробляємо інформаційну систему, яка повинна вести автоматизований облік успішності та відвідуваності студентів.

Інформаційна система «Електронний журнал» призначена для комплексного інформаційно-аналітичного забезпечення процесів

навчального закладу, в частині виконання наступних функцій:

- введення і редагування даних про успішність студентів;
- введення і редагування даних про відвідуваність студентів навчальних занять;
- введення і редагування списку студентів, список групи, і списку спеціальностей;
- введення і редагування навчального плану;
- висновок успішності по заданих параметрах;
- висновок відвідуваності.

Інформаційну систему «Електронний журнал» передбачається використовувати в навчальних закладах, які вирішують описані вище функції.

Основними цілями створення інформаційної системи є:

- підвищення ефективності виконання процесів, перерахованих вище, шляхом скорочення непродуктивних і що дублює операції, операцій, виконуваних «вручну», оптимізації інформаційна взаємодія учасників процесів;
- підвищення якості прийняття управлінських рішень шляхом оперативності подання, повноти, вірогідності та зручності форматів зображення інформації;
- підвищення інформаційної відкритості та прозорості діяльності навчального закладу;
- підвищення зручності та комфорту при отриманні інформації про послуги, що надаються споживачеві.

1.3 Вимоги до інформаційної системи

Ключовими вимогами до створюваної системи є:

— відкритість, тобто вона повинна відповідати всім сучасним стандартам,

- підтримка Веб-технологій, а також можливість додавання функціоналу як сторонніх розробників, так і напрацювань учнів;
- масштабованість, як ключова вимога з точки зору економії. Чи не доведеться заново перебудувувати систему при нарощуванні функціоналу;
 - багатоплатформність, здатність працювати на різних пристроях, операційних системах, серверах;
 - адаптованість, тобто можливість простого налаштування під потреби замовника;
 - розширюваність, тобто можливість нарощування функціональних можливостей системи, не змінюючи прийнятого раніше методу розвитку і бази,
 - локалізація, тобто підтримка національних вимог і стандартів в області документообігу, організації процесу навчання, особливостей Української системи освіти.

Основні вимоги до розроблюваної інформаційної системи щодо функціональних можливостей полягають у наступному:

- система повинна забезпечувати конфіденційність інформації таким чином, що персональні дані студента може переглядати тільки він сам, його батьки та педагогічний персонал технікуму, а вносити зміни — тільки відповідні педагоги;
- студент закріплений за однією з груп, а група, у свою чергу, закріплена за однією зі спеціальностей;
- кожна дисципліна закріплена за однією з груп, а також за одним з викладачів;
- код студента є унікальним і незмінним;
- номери дисциплін, груп, спеціальностей унікальні та незмінні, а їх найменування можуть змінюватися.

РОЗДІЛ 2

ФУНКЦІОНАЛЬНО-ОРІЄНТОВАНЕ ПРОЕКТУВАННЯ

2.1 Побудова контекстної діаграми

Сама верхня діаграма, на якій об'єкт моделювання продемонстрована цілим блоком з обмеженими стрілками позначається А-0. Стрілки на цій діаграмі показують зв'язки об'єкта моделювання з усім світом. З огляду на те, що єдиний блок репрезентує об'єкт, ім'я — універсальне для всього проекту. Це ж справедливо і для всіх стрілки діаграми, тому що вони зображують цілий набір зовнішніх інтерфейсів цього об'єкта. Діаграма А-0 розміщує район моделювання та її кордон. Контекстна діаграма А-0 також повинна включати короткі затвердження, що визначають позицію посадової особи або підрозділу, з позицій якого створюється модель, і мета, для досягнення якої її розробляють. Ці ствердження допомагають розпоряджатися розробкою моделі та ввести цей процес у певні міри. Позиція визначає, що і в якому розрізі можна побачити в межах контексту моделі. Зміна точки зору, призводить до розгляду інших аспектів об'єкта. Аспекти, які з однієї точки зору важливі, можуть не з'явитися в моделі, що розробляється під іншим кутом зору на той же самий об'єкт. Розвиток мети висловлює причину створення моделі, тобто містить перелік питань, на які повинна відповідати модель, що значною мірою визначає її структуру. Важливі критерії об'єкта найчастіше проявляються на верхніх рівнях ієрархії; функції вищого рівня уточнюються в міру розбиття їх на підфункції та декомпозицію. Кожна і, зі свого боку, декомпозується на компоненти подальшого рівня, цей процес буде до тих пір, поки релевантна система не буде отримана, яка допускає отримання відповідей на питання, сформульовані в мети моделювання. Будь-яка підфункція буде моделюватися своєрідним блоком. Кожен батьківський блок вичерпано відтворюється дочірньої діаграми на мінімальному рівні. В

межах території контекстної діаграми вищого рівня мають бути усі дочірні діаграми. Роботи (Activity) показують процеси, функції або завдання, які трапляються на протязі певного часу і мають розпізнавальні підсумки. Роботи відображаються у вигляді прямокутників. Всі роботи повинні бути названі та визначені. Назва роботи повинна бути виражена віддієслівним іменником, що позначає дію (наприклад, «Ведення електронного журналу успішності та відвідуваності»). Робота «Ведення електронного журналу успішності та відвідуваності» може мати, наприклад, наступне визначення: «Це модель, яка описує процес багаторазового введення і редагування персональних даних студентів та подальшого їх виведення користувачеві». Створення нової моделі автоматично створює контекстну діаграму з єдиною роботою, яка відображує систему в цілому. Співдія робіт із зовнішнім світом і між собою розглядається у вигляді стрілок. Стрілки становлять певну інформацію та іменуються іменниками.

Розрізняють 4 види стрілок:

— вхідні, вони ставляться з лівого боку та описують матеріал або інформацію, яка перетворюється в даній роботі;

— вихідні, вони ставляться з правого боку та описують матеріал або інформацію, яка виробляється даною роботою (обов'язково має бути присутня хоча б одна стрілка цього виду);

— управління- правила, процедури або стандарти, нормативні акти, на основі яких повинна здійснюватися дана робота (також має бути присутня хоча б одна стрілка);

— ресурси або механізми (матеріальні, трудові, фінансові ресурси).

На рис. 3.2.1 представлена контекстна діаграма «Ведення електронного журналу і відвідуваності».

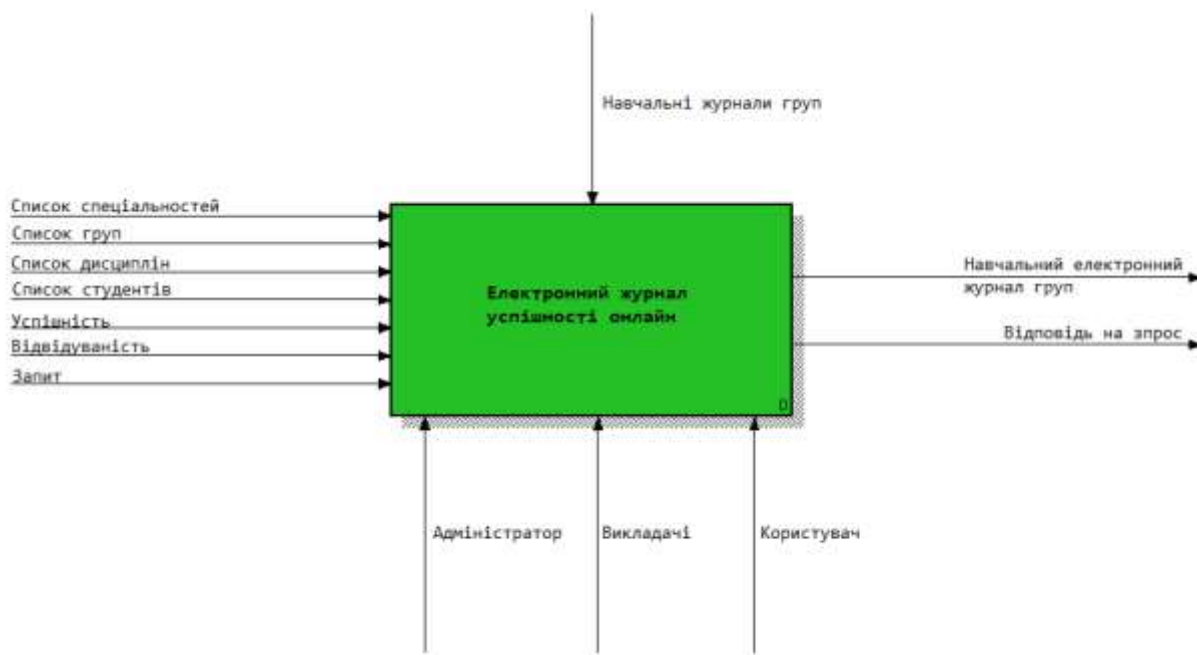


Рисунок 3.2.1 - Контекстна діаграма «Ведення електронного журналу і відвідуваності»

2.2 Побудова діаграм декомпозиції IDEF0

Описуючи систему, її потрібно розбити на об'ємні частини. Даний процес іменується механічною декомпозицією, а діаграми, які окреслюють будь-яких частин і взаємодій фрагментів, називають діаграмою декомпозиції. Після декомпозиції цієї контекстної діаграми ведеться фрагментація системи на більш малі та т.д., до досягнення відповідного значення подробности опису. Після кожного етапу декомпозиції робляться сеанси експертизи — фахівці предметної області показують на співвідношення реальних бізнес-процесів зроблених діаграм. Знайдені незбіжності удосконалюються, тільки після проходження повної експертизи в відсутності помилок дозволено прилучитися до подальшого етапу декомпозиції. Так досягається співвідношення моделі бізнес-процесів університету на будь-якому рівні моделі. Синтаксис відтворення системи в цілому і кожного її фрагмента схожий на протязі побудови всієї моделі. Діаграми декомпозиції залучають

схожі роботи, інакше кажучи, дочірні роботи, що мають унікальну роботу. При розробці діаграми декомпозиції слід вказати нотацію нової діаграми та чисельність робіт на ній. Можлива перерва кількості робіт - 2-8. Компонувати роботу на одну роботу не має суті: діаграми з чисельністю робіт більше ніж 8 виходять понад насиченими та некоректно читаються. Для забезпечення наочності та кращого осмислення імітованих процесів рекомендовано застосовувати від 3-х до 6 блоків на одній діаграмі. Якщо виявилось, що кількість робіт недостатньо, то роботу дозволено додати в діаграму, клацнувши спочатку по кнопці на палітрі інструментів, а потім по вільному місцю на діаграмі. Роботи на діаграмах традиційно всі розміщуються діагонально, починаючи від лівого кута, який знаходиться зверху і прямує до правого нижнього. Такий розпорядок іменується в лівому верхньому кутку поміщається найважливіша робота, виконувана за часом першої. Далі вправо донизу розміщуються найменш принципові або виконувані після цього роботи. Таке позиціонування спрощує читання діаграм, не рахуючи того, що на ньому базується поняття взаємовідношення робіт (рис. 3.3.1). Кожна операція на діаграмі декомпозиції може так само бути докомпонована (рис. 3.3.2). Усі роботи на діаграмі будуть нумеруватися автоматично тільки зліва направо. Номер роботи демонструється в правому нижньому кутку.

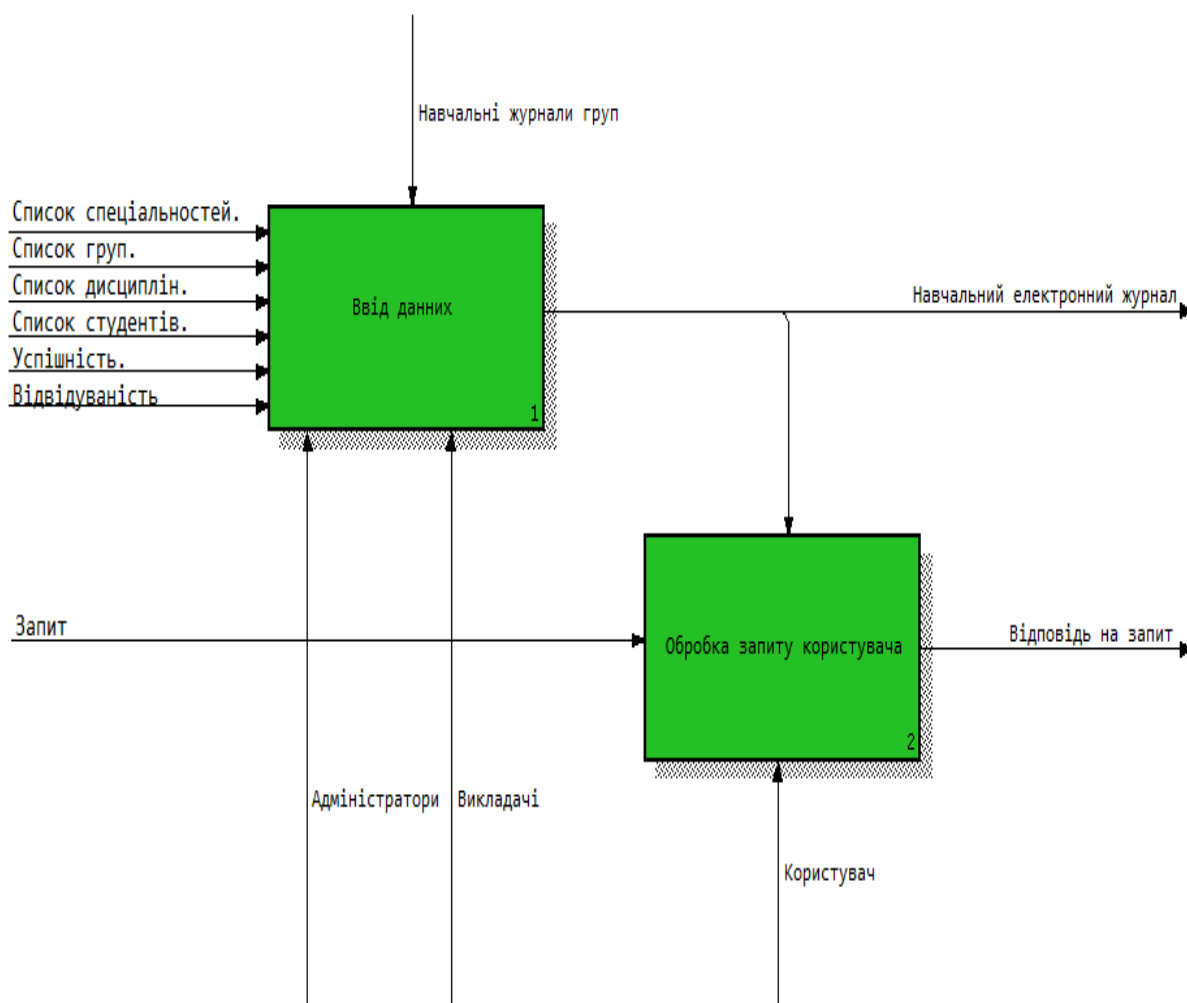


Рис. 3.3.1 — Декомпозиція першого рівня для процесу «Ведення електронного журналу успішності і відвідуваності»

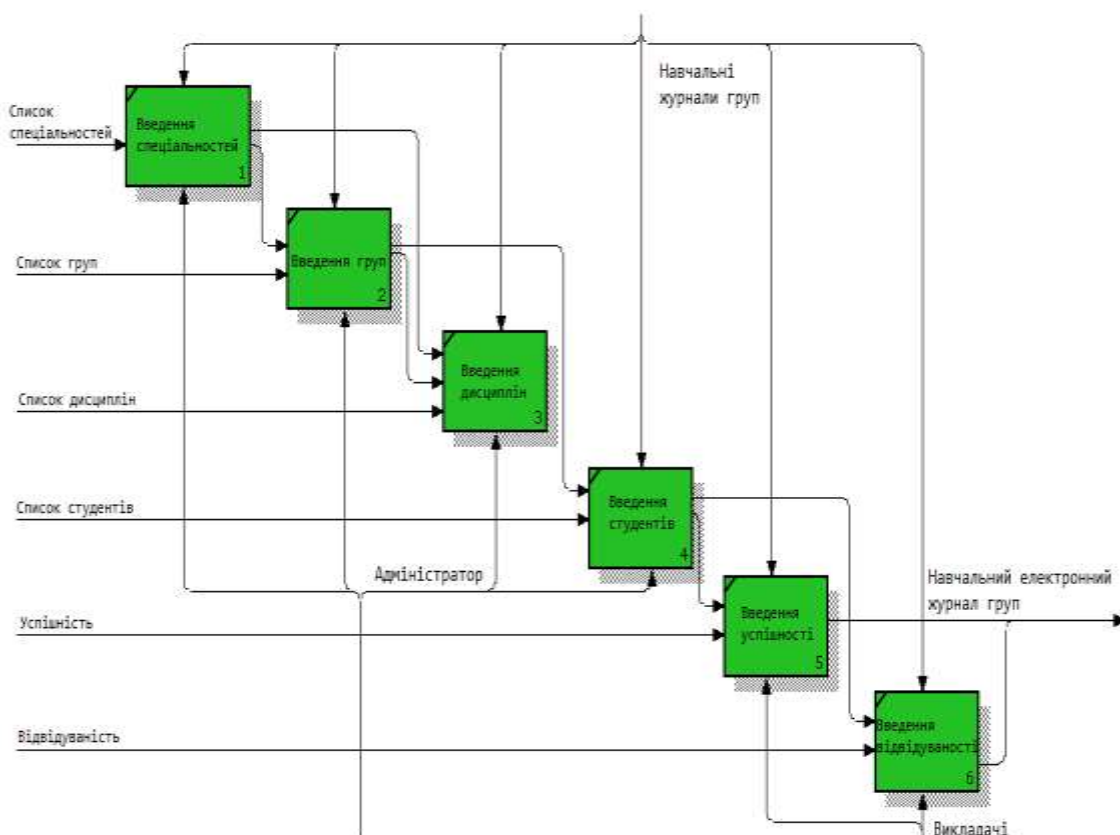


Рисунок 3.3.2 — Декомпозиція другого рівня для процесу «Введення даних»

— Декомпозиція діаграми розроблена з метою того, щоб була можливість роз'яснювати процес. Навпроти від цих моделей, що розкривають організацію цієї структури, робота цієї діаграми верхнього рівня в IDEF0 - це не частина напрямку підпорядкованої роботи. Роботи нижнього рівня — це те ж саме, що роботи верхнього рівня, але в більш доскональшому вигляді. Як кінцевий результат верхнього рівня цієї роботи це однакове з кордонами діаграми декомпозиції.

— Щоб декомпонувати роботи входять в неї і виходять з неї стрілки (крім стрілки виклику) одразу показуються на діаграмі декомпозиції (переміщення стрілок), але не належать роботі.

— Для зв'язування стрілок входу, управління або механізму потрібно досягти режиму виправлення стрілок, клацнути по наконечнику стрілки та

потім по доречному компоненту роботи. Щоб з'єднати стрілки виходу треба натиснути на режим редагування стрілок, клацнути на вихід роботи, а далі вже на стрілку.

— Між собою роботи з'єднуються з метою застосування внутрішніх стрілок, іншими словами, стрілки, які не належать кордону діаграми, розпочинаються з однієї та закінчуються в іншій роботі.

— Для того, щоб можна було намалювати внутрішні стрілки треба в розкладі малювання стрілок натиснути на сегмент (наприклад, виходу) однієї роботи і потім на сегмент (наприклад, входу) іншої. В IDEF0 розрізняють роботи по типу таких зв'язків:

— зв'язок по входу (output-input), коли стрілка виходу вищестоящої роботи прямує на вхід нижчої (наприклад, на рис. 2 стрілки «Відомість обліку продукції» з'єднує роботи «Введення і редагування даних про випуск продукції» та «Пошук даних про продукцію»);

— зв'язок з управління (output-control), коли вихід вищестоящої роботи прямує на управління нижчої. Панування вищестоящої роботи демонструє зв'язок з управління. Дані або об'єкти виходу вищестоящої роботи не перетворюються в нижчий;

— зворотний зв'язок по входу (output-input feedback), коли вихід нижчої роботи прямує до входу вищестоящої. Цей зв'язок, найчастіше, вживається для зображення циклів;

— зворотний зв'язок з управління (output-control feedback), коли вихід нижчої роботи спрямовується на управління вищестоящої. Кожна стрілка має основою одну-єдину роботу і метою теж одну-єдину роботу. Відгалужуються та об'єднуються стрілки. Одні і ті ж дані або об'єкти, є результатом однією роботою, вони також можуть бути застосовані одразу в різноманітних роботах. З одної сторони, стрілки, знаходяться в різноманітних роботах, можуть розкривати собою такі самі або аналогічні дані та об'єкти, які надалі застосовуються або змінюються в єдиному

місці. Моделювання цієї ситуації в IDEF0 застосовується розгалуження та зливання стрілок роботи. Для розгалуження стрілки подобає в режимі редагування стрілки натиснути по фрагменту стрілки та за відповідним сегментом роботи. Для того, щоб обидві стрілки виходу злилися необхідно в режимі редагування стрілки насамперед натиснути на сегмент виходу роботи, а потім вже по доречному фрагменту стрілки. Зміст розгалуження та зливання стрілок, переходить ім'ям кожної гілки стрілок. Зображуються певні правила іменування цих стрілок. Розберемо їх на прикладі розгалуження стрілок. Якщо стрілка має ім'я до розгалуження, а після розгалуження ніяка з гілок не йменувалася, то передбачається, що всі гілки моделюють ті ж дані або об'єкти, що і гілка до розгалуження. Якщо стрілка має ім'я до розгалуження, а після розгалуження котрась із гілок теж іменована, то мається на увазі, що ці гілки відповідають іменуванню. Якщо при цьому певна гілка після розгалуження залишилася найменованою, то йдеться про те, що вона моделює ті ж дані або об'єкти, що і гілка до розгалуження. В роботі нумеруються абсолютно всі моделі. Складається номер моделі з префікса і числа. Зазвичай використовують префікс А, але може бути вжитий та префікс будь-якої довжини. Контекстна (коренева) робота дерева має номер А0. Роботи декомпозиції А0 мають номери А1, А2, А3 і т.д. Номер батьківської роботи та черговий порядковий номер можуть мати тільки роботи, які знаходяться на нижньому рівні, приміром служать роботи декомпозиції А3, які мають ці номери А31, А32, А33, А34 і т.д. Роботи формують ієрархію, де кожна робота може мати одну батьківську і кілька дочірніх робіт, утворюючи дерево. Таке дерево називають деревом вузлів, а вищеописану нумерацію — нумерацією по вузлах. Двократну нумерацію можуть мати тільки діаграми програми IDEF0. Всі діаграми в роботі мають номери по вузлу. Номер А0 має тільки контекстна діаграма, номер А0 — декомпозиція контекстної діаграми, інші діаграми декомпозиції —

номери за відповідним вузлу (наприклад, A1, A2, A21, A213 і т. Д.). ВРwin одразу забезпечує нумерацію по вузлах, інакше кажучи при організації декомпозиції формується нова діаграма і їй автоматично видається відповідний номер. Як наслідок здійснення експертизи діаграми можуть уточнювати та перетворюватися, таким чином, можуть бути засновані різноманітні адаптації однієї та тієї ж (з точки зору її розподілу в дереві вузлів) діаграми декомпозиції. В ВРwin допускається мати моделі однієї діаграму декомпозиції в цьому вузлу.

2.3 Інформаційний аналіз предметної області

При проектуванні інформаційного забезпечення системи вивчається предметна область, проводиться аналіз даних, встановлюються основні об'єкти предметної області[2. С. 384-386].

Першим кроком при проектуванні інформаційного забезпечення є виділення складу документів і їх реквізитів на основі аналізу предметної області. Так само на даному етапі проводиться аналіз складу форм документів предметної області. На підставі аналізу предметної області можна виділити наступні вхідні документи: перелік спеціальностей, список груп, список студентів, список дисциплін, журнали з успішністю і відвідуваністю студентів.

Форми даних документів предметної області розглянуті в додатках (див. Додатки А, Б, В).

Наступним кроком при побудові інформаційно-логічної моделі є встановлення функціональних залежностей між реквізитами на основі аналізу предметної області та аналізу документів предметної області.

На даному етапі визначається залежність одного реквізиту від іншого, і якщо така залежність є, то між даними реквізитами будується зв'язок.

Інформацію, отриману в ході виконання даного кроку, представимо у

вигляді наступної таблиці.

Далі необхідно розділити всі реквізити на описові і ключові і як ключова вимога з точки зору економії вкладень, яке гарантуватиме, що не доведеться перебудовувати систему в міру зростання обсягу оброблюваної інформації і збільшувати функціональні залежності реквізитів.

Для аналізу реквізитів складемо таблицю (табл. 4.1.1).

Таблиця 4.1.1

Визначення типів реквізитів

документ	описовий реквізит	ключовий реквізит	вид ключа	Ім'я інформаційного об'єкта
Список студентів	names_student ns_group passwords	ns_zbook	П, У	Студенти
Список викладачів	names_teacher logins passwords	ns_teacher	П, У	Викладачі
Список спеціальностей	names_speciality terms_study sabbreviation	ns_speciality	П, У	Спеціальності
Список груп	ns_spciality names_group ns_teacher	ns_group	П, У	Групи
Список дисциплін	ns_group names_discipline ns_teacher sems_start sems_end	ns_discipline	П, У	Дисципліни
Рубіжна успішність	marks	ns_zbook ns_discipline mons	С, У	Успішність
Семестрова успішність	marks	ns_zbook ns_discipline ns_sem	С, У	Успішність
Рубіжна відвідуваність	nos_reazon totals	ns_zbook mons	С, У	Відвідуваність

Семестрова відвідуваність	nos_reazon totals	ns_zbook ns_sem	С, У	Відвідуваність
------------------------------	----------------------	--------------------	------	----------------

Наступним кроком при побудові інформаційно-логічної моделі є опис інформаційних об'єктів, тобто виконується структурування описових реквізитів, які однаково залежні від одного або декількох ключових реквізитів. У кожному групі включимо загальні для цієї групи ключові реквізити. Кожна така група і буде називатися інформаційним об'єктом.

Для аналізу реквізитів складемо таблицю (табл. 4.1.2).

Таблиця 4.1.2

Опис інформаційних об'єктів

Реквізити ІВ	Ознака ключа	ім'я ІВ	Семантика
ns_zbook names_student ns_group passwords	П, У	Студенти	Відомості про студентів
ns_teacher names_teacher logins passwords	П, У	Викладачі	Відомості про викладачів
ns_speciality names_speciality terms_study sabbreviation	П, У	Спеціальності	Відомості про спеціальності
ns_group ns_spciality names_group ns_teacher	П, У	Групи	Відомості про групи
ns_discipline ns_group names_discipline ns_teacher sems_start sems_end	П, У	Дисципліни	Відомості про дисциплінах
ns_zbook, ns_discipline, mons,	С, У	Успішність	Відомості про рубіжної

mark			успішності студентів
ns_zbook, ns_discipline, ns_sem marsk	C, Y	Успішність	Відомості про семестрової успішності студентів
ns_zbook, mons nos_reazon, totals	C, Y	Відвідуваність	Відомості про рубіжної відвідуваності студентів
ns_zbook, ns_sem nos_reazon, totals	C, Y	Відвідуваність	Відомості про семестрової успішності студентів

Інформаційні об'єкти, які повинні справджувати всім вимогам нормалізації:

- винятковий ідентифікатор може бути включеним в інформаційний об'єкт.
- взаємно незалежними мають бути всі описові реквізити; - усі реквізити, що входять в складову ключів повинні бути взаємно незалежні; - практично-повно може залежати будь
- який спеціалізований стовпчик від ключа інформаційного об'єкта; - ілюстративні реквізити при складеному ключі мають бути повністю залежними від реквізитів складових ключів;
- кожен пояснювальний реквізит не може бути в залежності від ключа.

Між інформаційними об'єктами відповідно вимогам нормалізації допускають зв'язку 1: 1 і 1: M. У (табл. 4.1.3). визначимо тип зв'язку між інформаційними об'єктами.

Таблиця 4.1.3

Визначення типів

Номер зв'язку відносини	Головний ІВ	Підлеглий ІВ	Тип відносини
1	Спеціальності	Групи	1:М
2	Групи	Студенти	1:М
3	Викладачі	Групи	1:1
4	Викладачі	Дисципліни	1:М

При складанні інформаційно-логічної моделі інформаційні

дослідження повинні бути впорядковані за рівнями (рис.

4.1.1).[BR]Представлена інформаційно-логічна модель розглянутого предметного сегменту, зроблена відповідно до виявлених інформаційних об'єктів та зв'язків між ними.

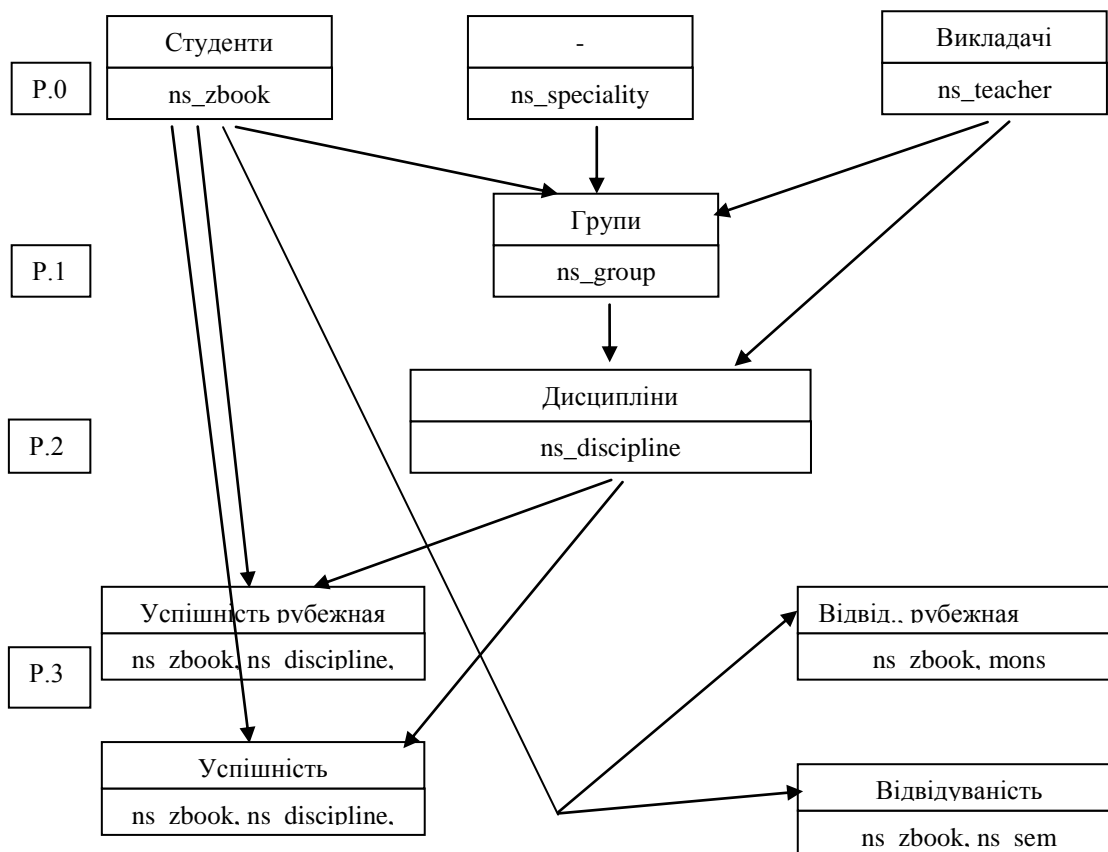


Рисунок 4.1.1 - Рівні інформаційних об'єктів

Інформаційно-логічна модель приведена в обов'язковому образі, і об'єкти в ній розташовані за рівнями. На мінімальному рівні розміщуються об'єкти, які не підпорядковані ніяким іншим об'єктам. Рівень інших об'єктів визначається найдовшим шляхом до об'єкту від нульового рівня. Таке розміщення об'єктів дає уявлення про їх ієрархічну підпорядкованість, робить модель більш науковою та полегшує розуміння багатозначних відносин між об'єктами.

2.4 Опис таблиць бази даних

Логічна структура реляційної бази даних є адекватним відображенням отриманої інформаційно-логічної моделі предметної області [6, С. 592]. Для канонічної моделі не потрібно додаткових перетворень. Кожен інформаційний об'єкт моделі даних відображається відповідно реляційної таблиці. Структура реляційної таблиці визначається реквізитний складом відповідного інформаційного об'єкта, де кожен стовпець (поле) відповідає одному з реквізитів об'єкта. Ключові реквізити об'єкта утворюють унікальний ключ реляційної таблиці. Для кожного стовпця таблиці (поля) задається тип, розмір даних і інші властивості. Рядки (записи) таблиці відповідають екземплярам об'єкта і формуються при завантаженні таблиці.

У наведених формах документів з нормативно-довідкової і оперативної інформації представлені реквізити, значення яких повинні зберігатися в базі даних інформаційної системи. Ці значення вводяться з клавіатури комп'ютера або вибираються зі списків в екранних формах. Нижче наведені характеристики реквізитів документів предметної області. Проектувальник при необхідності може їх змінити, а також додати інші реквізити.

Таблиця 5.2.1

Таблиця «Список студентів»

Атрибут		Ознака ключа	Формат поля	
Ім'я	Назва		Тип	Довжина
Ids	Автоматичне поле	Первинний (Primary)	Числовий (Long Integer)	8
names_student	ПІБ студент		Текстовий (Text)	250
ns_group	Номер групи		Числовий (Long Integer)	8

Таблиця 5.2.2

Таблиця «Список викладачів»

Атрибут		Ознака ключа	Формат поля	
Ім'я	Назва		Тип	Довжина
ns_teacher	Номер викладача	Первинний (Primary)	Числовий (Long Integer)	8
names_teacher	ПІБ викладача		Текстовий (Text)	250

Таблиця 5.2.3

Таблиця «Список спеціальностей»

Атрибут		Ознака ключа	Формат поля	
Ім'я	Назва		Тип	Довжина
ns_speciality	Номер спеціальності	Первинний (Primary)	Числовий (Long Integer)	8
names_speciality	Назва спеціальності		Текстовий (Text)	250
terms_study	Час навчання		Числовий (Long Integer)	8
sabbreviation	Абревіатура		Числовий (Long Integer)	10

Таблиця 5.2.4

Таблиця «Список груп»

Атрибут		Ознака ключа	Формат поля	
Ім'я	Назва		Тип	Довжина
ns_group	Номер групи	Первинний (Primary)	Числовий (Long Integer)	8
ns_speciality	Номер спеціальності		Числовий (Long Integer)	8
names_group	Назва групи		Текстовий (Text)	250
ns_teacher	Класний керівник		Числовий (Long Integer)	8

Таблиця 5.2.5

Таблиця «Список дисциплін»

Атрибут		Ознака ключа	Формат поля	
Ім'я	Назва		Тип	Довжина
ns_discipline	Номер дисципліни	Первинний (Primary)	Числовий (Long Integer)	8
ns_group	Номер групи		Числовий (Long Integer)	8
names_discipline	Назва дисципліни		Текстовий (Text)	250
ns_teacher	Номер викладача		Числовий (Long Integer)	8
sems_start	Семестр початку навч.		Числовий (Long Integer)	2
sems_end	Семестр кінця навч.		Числовий (Long Integer)	2

Таблиця 5.2.6

Таблиця «семестрова відвідуваність»

Атрибут		Ознака ключа	Формат поля	
Ім'я	Назва		Тип	Довжина
ns_sem	Семестр		Числовий (Long Integer)	2
nos_reason	Без ув. причини		Числовий (Long Integer)	5
totals	Всього		Числовий (Long Integer)	5

РОЗДІЛ 3

РОЗРОБКА КЛІЄНТ-СЕРВЕРНОГО ДОДАТКА

3.1 Технологія «клієнт-сервер»

У міру розвитку уявлень про розподілення обчислювальних процесів і процесів обробки даних сформувалася теорія архітектури «клієнт-сервер» - узагальнене поняття про взаємодію 2-ух складових інформаційної технології в обчислювальних системах і мережах, серед яких логічно або фізично можуть бути виділені: функціональна сторона (джерело запитів, замовник); пасивна сторона (сервер, сервіс запитів, джерело відповідей).

Взаємодія «клієнт - сервер» в мережі здійснюється відповідно до визначеного протоколом.

У концепції «клієнт-сервер» мається на увазі, що крім зберігання централізованої бази даних центральна машина (сервер бази даних) повинна забезпечувати виконання основного обсягу обробки даних. Запит на дані, що видається клієнтом, породжує пошук і вилучення даних на сервері. Витягнуті дані транспортуються по мережі від сервера до клієнта. Специфікою архітектури клієнт-сервер є використання мови запитів SQL (Structured Query Language) [6. С. 592].

Запит до бази даних ініціюється клієнтом, однак виконується на сервері. Клієнту повертається по мережі тільки результат. Цей процес складається з шести етапів:

1. Клієнт запитує дані.
2. Запит транслюється в SQL.
3. SQL-запит передається по мережі на сервер.
4. Сервер бази даних здійснює пошук.
5. Необхідні записи повертаються клієнтові.
6. Дані пред'являються користувачеві.

Технологія «клієнт-сервер» створює потужне середовище, яка дає організаціям безліч реальних переваг. Зокрема, добре спланована клієнт-серверна система забезпечує відносно недорогу платформу, яка володіє, в той же час, обчислювальними можливостями мейнфреймів і легко налаштовується для виконання конкретних завдань. Крім того, при клієнт-серверної обробки різко зменшується мережевий трафік, так як через мережу посилаються тільки результати запитів.

Вантаж файлових операцій лягає в основному на комп'ютер-сервер, який набагато могутніше клієнтів і здатний тому краще обслуговувати запити.

Мережа моделі «клієнт-сервер» зменшує потребу комп'ютерів-клієнтів в оперативній пам'яті, оскільки вся робота з файлами виконується на сервері. Сервери в клієнт-серверних системах здатні зберігати велику кількість даних. Завдяки цьому на комп'ютерах-клієнтах звільняється значний обсяг дискового простору.

Також в значній мірі спрощується резервне копіювання даних.

Нарешті, управління всією системою, включаючи контроль за її безпекою, стає набагато простіше, так як всі файли і дані централізовано розміщуються на сервері або на невеликому числі серверів.

3.2 Засоби програмування

Для створення програми основним засобом розробки була мова програмування Python 3.8[1. С. 1104], [9. С. 624].

Високорівнева мова програмування загального призначення, орієнтована на підвищення продуктивності розробника і читання коду. Синтаксис ядра Python мінімалістичний. У той же час стандартна бібліотека включає великий обсяг корисних функцій.

Python підтримує кілька парадигм програмування, в тому числі

структурне, об'єктно-орієнтоване, функціональне, імперативне і аспектно-орієнтоване. Основні архітектурні риси - динамічна типізація, автоматичне керування пам'яттю, повна інтроспекція, механізм обробки виключень, підтримка багато поточних обчислень і зручні високорівневі структури даних. Код в Python організовується у функції та класи, які можуть об'єднуватися в модулі (вони в свою чергу можуть бути об'єднані в пакети).

Мова і його інтерпретатор розробляються групою ентузіастів в рамках проекту з відкритим кодом. Проект не є вільним і поширюється під власною ліцензією.

В області програмування для мережі Python - один з найпопулярніших скриптових мов (разом з JSP, Perl і мовами, використовуваними в ASP.NET) завдяки своїй простоті, швидкості виконання, багатій функціональності, платформ і розповсюдженню початкових кодів.

Популярність в області побудови web-сайтів визначається наявністю великого набору вбудованих засобів для розробки web-додатків.

Основні з них:

- автоматичне витяг POST і GET-параметрів, а також змінних оточення web-сервера в зумовлені масиви;
- файлові функції успішно обробляють як локальні, так і віддалені файли;
- автоматична відправка HTTP-заголовків;
- робота з cookies і сесіями;
- обробка файлів, що завантажуються на сервер;
- робота з HTTP заголовками і HTTP авторизацією;
- робота з XForms;
- робота з віддаленими файлами і сокетами.

В даний час Python використовується розробниками для високонавантажених проєктів. Згідно з рейтингом Tiobe, що базується на даних пошукових систем, в грудні 2009 року Python знаходиться на 3 місці

серед мов програмування (поступаючись Java і C), піднявшись за рік на дві позиції.

На даний момент існує єдина реалізація Python, жодна стороння компанія не підтримує виконуваних модулів, відмінних від офіційної збірки. Такий стан речей, з одного боку, дозволяє швидко впроваджувати і поширювати нововведення серед спільноти розробників, з іншого боку, розробляти мову програмування в умовах відсутності стандарту, так як єдина реалізація забезпечує його по факту. В таких умовах великого значення набуває версія інтерпретатора, що визначає поточну функціональність (зворотна сумісність між версіями інтерпретатора не дотримується строго).

У 1994 році данський програміст Расмус Лерддорф створив набір скриптів на Perl / CGI для висновку і обліку відвідувачів його онлайн-резюме, що обробляє шаблони HTML-документів. Лерддорф назвав набір Personal Home Page (Особиста Домашня Сторінка). Незабаром функціональності і швидкості Perl - інтерпретатора скриптів - перестало вистачати, і Лерддорф розробив з використанням мови C новий інтерпретатор шаблонів PHP/FI..

3.3 Програмна реалізація модуля

Для роботи з автоматизованою інформаційною системою «Електронний журнал» необхідно ввести URL в адресний рядок браузера. Після завантаження даних у вікні браузера з'явиться сторінка з якої починається робота в інформаційній системі - це сторінка аутентифікації користувача в системі (рис. 6.3.1).

Херсонський державний університет

Ім'я користувача:

Пароль:

Увійти

Рисунок 6.3.1 — Сторінка аутентифікації користувача

У центрі екрана розташована форма аутентифікації користувача, але якщо користувач, ще не зареєстрований в системі, то нижче наводиться посилання на сторінку реєстрації.

Після успішної аутентифікації викладач потрапляє на сторінку вибору відображення потрібного йому предмета, де він може виставляти оцінки, відзначати відвідуваність і редагувати тему уроку і домашнє завдання (рис. 6.3.2).

Херсонський державний університет

Домівка · Журнал успішності студентів

Адміністрування Журнал успішності студентів

ЖУРНАЛ УСПІШНОСТІ СТУДЕНТІВ		
Gradebook lessons	+ Додати	✎ Змінити
Gradebook subjects	+ Додати	✎ Змінити
Gradebooks	+ Додати	✎ Змінити
Відвідувачі лекційних занять	+ Додати	✎ Змінити
Відвідувачі практичних занять	+ Додати	✎ Змінити

Рисунок 6.3.2 — Сторінка вибору відображення інформації

Протягом всієї роботи з системою відображається персональна інформація про викладача: ПІБ викладача, доступні предмети, також поруч з цією інформацією розташована гіперпосилання для виходу користувача з системи, на додаток технічної підтримки або посилання на неї, а так само зміну пароля.

Після натискання на дату викладач може заповнити тему уроку і домашнє завдання (рис. 6.3.3).

Змінити Відвідувач практичного заняття

Gradebook lesson: ✎ + ✖
Журнал

Student: ✎ + ✖
Студент

Control: ▼
Вид контролю навчальних досягнень студентів

Result:
Оцінка (за 100-бальною шкалою)

Рисунок 6.3.3 — Сторінка поточну успішність студента

Для виставлення оцінок, необхідно просто вибрати комірку і поставити туди позначку, або «Н», якщо студент був відсутній на уроці.

3.4 Програмна реалізація модуля «Керівник групи»

Робота з системою для керівника групи також починається зі сторінки аутентифікації (рис. 6.4.1).

Рисунок 6.4.1 — Сторінка аутентифікації керівника

Після успішної аутентифікації керівник потрапляє на сторінку вибору потрібної дії: Додавання або виключення студентів, переглядати позначки і відвідуваність своєї групи, створювати звіти по соціальним групам, і групами здоров'я та оцінками (рис. 6.4.2).

Рисунок 6.4.2 — Сторінка складання звіту успішності

Після натискання на гіперпосилання з меню «Керівництво» керівник побачить сторінку, на якій розташовано меню для вибору дії, там він може вибрати додавати зв'язку або керувати студентами своєї групи (рис. 6.4.3).

Вибрати Студент для перегляду

Пошук

PERSON	EDUCATIONAL PROGRAMME	EDUCATION FORM	FUNDING FOR
Кісельгоф Анна Євгенівна	121 Інженерія програмного забезпечення (Бакалавр)	Денна	Бюджетна
Гунько Сергій Едуардович	121 Інженерія програмного забезпечення (Магістр)	Денна	Бюджетна
Попов Сергій Андрійович	121 Інженерія програмного забезпечення (Магістр)	Денна	Бюджетна
Давиденко Іван Іванович	121 Інженерія програмного забезпечення (Магістр)	Денна	Бюджетна
Сенчишен Денис Олександрович	121 Інженерія програмного забезпечення (Магістр)	Денна	Бюджетна
Лемещук Олександр Ігорович	121 Інженерія програмного забезпечення (Магістр)	Денна	Бюджетна

6 Студенти

Рисунок 6.4.3 — Сторінка управління студентами

Керівник може редагувати зв'язку з усіх дисциплін обраної групи тільки в тому випадку, якщо він є керівником цієї групи. Якщо він не є керівником обраної групи, то можливість редагування зв'язків буде тільки з тих дисциплін, які викладаються в цій групі.

Після введення змін керівник повинен натиснути кнопку «Зберегти» для того щоб нові дані були записані в базу даних або перейти на іншу сторінку щоб відмовитися від внесених змін.

На рис. 6.4.5 зображена сторінка редагування зв'язків.

Херсонський державний університет
ВІТАЄМО, СЕРГІЙ. [ДИВИТИСЯ САЙТ](#) / [ДОКУМЕНТАЦІЯ](#) / [ЗМІНИТИ ПАРОЛЬ](#) / [ВИЙТИ](#)

[Домівка](#) > [Навчальний процес](#) > [Навчальні предмети](#)

Вибрати Навчальний предмет для перегляду

НАВЧАЛЬНИЙ ПРЕДМЕТ
Математичне моделювання систем и процесів [Математичне моделювання систем и процесів [121 Інженерія програмного забезпечення (Магістр)]] [90]
Алгоритми і структури даних [Алгоритми і структури даних [121 Інженерія програмного забезпечення (Магістр)]] [90]
Алгоритми і структури даних [Алгоритми і структури даних [121 Інженерія програмного забезпечення (Магістр)]] [90]

3 Навчальні предмети

Рисунок 6.4.5 — Сторінка редагування зв'язків

ВИСНОВКИ

У дипломному проекті нами побудована модель інформаційної системи «Електронний журнал», розроблено сервіс для автоматизації процесу ведення журналу успішності та відвідуваності. В ході роботи було проаналізовано сучасний стан проблеми автоматизації діяльності освітніх установ, виявлено переваги та недоліки, обрано оптимальну структуру системи бізнес-процесу. Вивчена предметна область автоматизації та розроблена структура системи. При розробці інформаційного забезпечення системи були проаналізовані: склад і структура інформації, встановлені функціональні залежності реквізитів, виділені інформаційні об'єкти. Побудовано функціональна, логічна, фізична моделі системи. Реалізована база даних. Програмне забезпечення написано мовою програмування Python і Web framework Django.

Були виконані наступні завдання:

- 1) Дослідження наявних програмних продуктів для управління бізнес-процесами університету.
- 2) Побудова контекстної діаграми.
- 3) Розроблено програмний інтерфейс API та документація до нього.
- 4) Розроблено сервіс «Електронний журнал».

При розробці проекту використовували сервіс Git з публічним репозитарієм на сервісі GitHub, також такий метод розробки дозволяє використовувати програмний код при сумісній роботі та дозволяє використовувати результати проведеного дослідження всім охочим під ліцензією MIT. Розроблена система та сервіси можуть використовуватись за прямим призначенням.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гагаріна Л.Г. Розробка і експлуатація автоматизованих інформаційних систем / Л.Г. Гагаріна, Д.В. Кисельов, Е.Л. Федотова; Ред. Л.Г. Гагаріна. - М.: ФОРУМ: ИНФРА-М, 2013. - 384 с.
2. Веллінг Л. Розробка Web-додатків за допомогою PHP і MySQL - 3-е изд. / Л. Веллінг, Л. Томсон; під ред. Ю.Н. Артеменко; пер. з англ. - М.: Видавничий дім «Вільямс», 2010. - 880 с.
3. Компанія MySQL AB. MySQL. Довідник по мові / за редакцією Ю.М. Артеменко; пер. з англ. - М.: Видавничий дім «Вільямс», 2005. - 432 с.
4. Ємельянова Н.З. Основи побудови автоматизованих інформаційних систем: навч. посібник / Н.З. Ємельянова, Т.Л. Партика, І.І. Попов. - М.: ФОРУМ: ИНФРА-М, 2005. - 416 с.
5. Діго С.М. Бази даних: проектування та використання / С.М. Діго. - М.: Фінанси і статистика, 2005. - 592 с.
6. Голіцина О.Л. Основи алгоритмізації і програмування / О.Л. Голіцина, І.І. Попов. - М.: ФОРУМ: ИНФРА-М, 2005. - 432 с.
7. Гвоздьева В.А. Вступ до спеціальності програміста / В.А. Гвоздьева. - М.: ФОРУМ: ИНФРА-М, 2005. - 208 с.
8. Шлоссейгл Д. Професійне програмування на Python / Д. Шлоссейгл; пер. з англ. - М.: Видавничий дім «Вільямс», 2009 - 624 с.
9. Таненбаум Е.М. ван Стеен. Розподілені системи. Принципи та парадигми / Е.М. ван Стен. Таненбаум; пер. з англ. - М.: Видавничий дім «Пітер», 2013. - 877 с.
10. Попова О.Г. Методичні вказівки по виконанню дипломного проекту для студентів денної форми навчання спеціальності 230103 «Автоматизовані системи обробки інформації та управління» / О.Г. Попова, А.А. Дроздова. - Вологда: ВКТ, 2008. - 28 с.
11. Скакун К.К. Методичні вказівки до виконання економічної частини

дипломного проекту для студентів за фахом 230103-51 «Автоматизовані системи обробки інформації та управління» / К.К. Скакун. - Вологда: НОУ СПО ВКТ, 2007 - 36 с.

12. ДСТУ ISO / ІЕС 12207-99 Інформаційна технологія. Процеси життєвого циклу програмних засобів / Введ. 01.07.2000

13. ГОСТ 19.102-77 Єдина система програмної документації. Стадії розробки.

14. ГОСТ 19.105-78 Єдина система програмної документації. Загальні вимоги до програмних документів.

15. Расс Унгер. Кэролайн Чендлер. UX-дизайн. Практичний посібник з проектування досвіду взаємодії. 2011. с. 26-115.

16. Бирман Ілья. Інтерфейс користувача. 2016. с. 15-46.

17. Итана Маркотта. Чуйний веб-дизайн. 2012. с. 31-39.

18. Ден Сідерхолм. CSS3 для веб-дизайнерів. 2017. с. 50-89.

19. Джеймі Леві. UX-стратегія. Чого хочуть користувачі і як їм це дати. 2017. с. 25-96.

20. Дронов В.А. Django 2.1. Практика створення веб-сайтів на Python. 2019. с. 89-115.

21. Малярчук С. М. Основи інформатики у визначеннях, таблицях і схемах: Довідково-навчальний посібник. 2017. с. 112.

22. Едріен Моует. Використання Docker. Розробка і впровадження програмного забезпечення за допомогою технології контейнерів. 2017. с. 18-356.

23. Дронов В. А. Django: практика створення Web-сайтів на Python. 2016. с. 29-36.

24. Парміндер С.К. Мікросервіси і контейнери Docker. 2016. с. 34-215.

25. Джефф Форсьє. Django. Розробка веб-додатків на Python. 2009. с. 64.

26. Прохоренко Н.А. Python 3 і PyQt. Розробка додатків. 2018. с. 26.

27. Персіваль Г. Python. Розробка на основі тестування. 2018. с. 87.

28. Алчин М. Pro Django, 2-е видання. 2013. с. 168.
29. Тарек Зиаде. Розробка Python Microservices. 2017. с. 201.
30. Офіційний сайт PandaJs - Режим доступу <http://www.pandajs.net/>
31. Коржинський С. Н. Настольна книга Web-мастера: ефективне примінення HTML, CSS и JavaScript.
32. Офіційний сайт Python Режим доступу - <https://www.python.org>
33. Офіційний сайт вікіпедія Режим доступу - <https://ru.wikipedia.org>
34. Створення сайтів з Python Режим доступу - <http://python-3.ru/page/php-vs-python>
35. Владимир Дронов. Практика создания веб-сайтов на Python,. С. 220-235.
36. Алексей Васильев. Python на примерах. Практический курс по программированию, 2019. С. 156-160.
37. Веб-разработка. Исчерпывающее руководство . Режим доступу - <https://habr.com/ru/company/piter/blog/314562/>
38. Співаковський О. В. Web-портал Херсонського державного університету. — 2013. — [URL:http://kspu.edu/](http://kspu.edu/).
39. Співаковський О. Вінник М. Тарасіч Ю. Побудова ІКТ інфра-структури ВНЗ: проблеми та шляхи вирішення // Інформаційні технології засоби навчання.— 2014. — 39, вип. 1. — С. 99—116.
40. Львов М. Співаковський О. Щедролосьєв Д. Інформаційна система управління вищим навчальним закладом як платформа реалізації управління академічним процесом // Комп'ютер у школі та сім'ї. — 2007. — № 2. — С. 3—6.

Додаток Б
Журнал навчальних занять

Назва дисципліни _____

№	ПІБ		місяць		Дата	
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						
24						
25						
26						
27						
28						
29						
30						
31						
32						
33						
34						
35						
36						
37						
38						
39						
40						
41						
42						
43						
44						
45						
46						
47						
48						
49						
50						

Додаток В

Додаток 1

**КОДЕКС АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ
ЗДОБУВАЧА ВИЩОЇ ОСВІТИ ХЕРСОНЬКОГО
ДЕРЖАВНОГО УНІВЕРСИТЕТУ**


Я, Туніко Сергій Едуардович
учасник(и) освітнього процесу Херсонського державного університету, УСВІДОМЛЮЮ, що академічна доброчесність – це фундаментальна етична цінність усієї академічної спільноти світу.

ЗАЯВЛЯЮ, що у своїй освітній і науковій діяльності **ЗОБОВ'ЯЗУЮСЯ**

- дотримуватися
 - вимог законодавства України та внутрішніх нормативних документів університету, зокрема Статуту Університету;
 - принципів та правил академічної доброчесності;
 - нульової толерантності до академічного плагіату;
 - моральних норм та правил етичного поведінки;
 - толерантного ставлення до інших;
 - дотримуватися високого рівня культури спілкування;
- надавати згоду на
 - безпосередню перевірку курсових, кваліфікаційних робіт тощо на ознаки наявності академічного плагіату за допомогою спеціалізованих програмних продуктів;
 - оброблення, збереження й розміщення кваліфікаційних робіт у відкритому доступі в інституційному репозитарії;
 - використання робіт для перевірок на ознаки наявності академічного плагіату в інших роботах виключно з метою виявлення можливих ознак академічного плагіату;
- самостійно виконувати навчальні завдання, завдання поточного й підсумкового контролю результатів навчання;
 - надавати достовірну інформацію щодо результатів власної навчальної (якубової, творчої) діяльності, використаних методик досліджень та джерел інформації;
 - не використовувати результати досліджень інших авторів без використання посилань на їхню роботу;
 - своєю діяльністю сприяти збереженню та примноженню традицій університету, формуванню його позитивного іміджу;
 - не чинити правопорушень і не сприяти такому скоєнню іншими особами;
 - підтримувати атмосферу довіри, взаємної відповідальності та співпраці в освітньому середовищі;
 - поважати честь, гідність та особисту недоторканність осіб, незалежно від її статі, віку, матеріального стану, соціального становища, расової належності, релігійної й політичної переконаності;
 - не дискримінувати людей на підставі академічного статусу, а також за національним, расовою, статевим чи іншим належністю;
 - відповідально ставитися до своїх обов'язків, вчасно та сумлінно виконувати особливі навчальні та науково-дослідницькі завдання;
 - запобігати виключенню у своїй діяльності конфлікту інтересів, зокрема не використовувати службових і родинних зв'язків з метою отримання нечесної переваги в навчальній, науковій і трудовій діяльності;
 - не брати участі в будь-якій діяльності, пов'язаній із обманом, нечесністю, списуванням, фабрикацією;
 - не підбюроувати документи;
 - не поширювати неправдиву та компрометуючу інформацію про інших здобувачів вищої освіти, викладачів і співробітників;
 - не отримувати і не пропонувати винагород за несправданне отримання будь-яких переваг або здійснення впливу на зміну оцінювальної академічної оцінки;
 - не залякувати й не проявляти агресії та насильства проти інших, сексуальні домагання;
 - не завдавати шкоди матеріальним цінностям, матеріально-технічній базі університету та особистій власності інших студентів та/або працівників;
 - не використовувати без дозволу ректорату (деканату) символіки університету в заходах, не пов'язаних з діяльністю університету;
 - не здійснювати і не заохочувати будь-яких спроб, спрямованих на те, щоб за допомогою нечесних і легітимних методів досягти власних корисних цілей;
 - не завдавати загрози власному здоров'ю або безпеці іншим студентам та/або працівникам.

УСВІДОМЛЮЮ, що відповідно до чинного законодавства у разі недотримання Кодексу академічної доброчесності буду висті академічну та/або інші види відповідальності й до мене можуть бути застосовані заходи дисциплінарного характеру за порушення принципів академічної доброчесності.

05.05.20 (дата)

 (підпис)

Сергій Туніко (ім'я, прізвище)