

Київський національний торговельно-економічний університет
Кафедра програмної інженерії та кібербезпеки

ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ
ПРОЕКТ

на тему:

Розробка програмного забезпечення аналізу
успішності студентів

Студента 4 курсу, 10 групи,
Напям підготовки(6.050103

Потоцького Данііла
Станіславовича

підпис студента

Науковий керівник
кандидат технічних наук,
доцент кафедри програмної
інженерії та кібербезпеки

Харченко Олександр
Анатолійович

підпис керівника

Гарант освітньої програми
кандидат технічних наук,
доцент

Цензура Микола
Олександрович

підпис керівника

					КНТЕУ 6.050103 10-11.БР			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів Титульна сторінка	Стадія	Аркуш	Акрюшів
Зав. Каф.		Криворучко О.В				ТС	1	39
Керівник		Харченко О.А				Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група		
Гарант		Цензура М.О.						
Розробив		Потоцький Д.С.						

ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1. СТВОРЕННЯ ВЕБ-САЙТІВ	5
1.1 Визначення веб-додатків	5
1.2 Особливості web-додатків:	5
1.3 Види веб-додатків.....	7
1.4 Основні етапи створення веб- сайтів	9
1.5 Визначення проекту	9
1.6 Інформаційне наповнення сайту.....	10
РОЗДІЛ 2 ЗАСОБИ РОЗРОБКИ ПРОГРАМНОГО ПРОДУКТУ	11
2.1 Розробка інтерфейсу та дизайну веб- додатку.....	11
2.2 Написання програмного коду та верстка сайту	11
2.3 Тестування – фінальний етап створення проекту.....	12
2.4 Засіб веб-розробки HTML.....	12
2.5 База даних	14
2.6 Мова програмування C#.....	14
2.7 SQL.....	16
2.8 Фреймворк програмний каркас.....	17
2.9 ADO.NET.....	18
Висновки по розділу 2	19
РОЗДІЛ 3. ТЕХНІЧНЕ ЗАВДАННЯ ПРОЕКТУ. ПРОЕКТУВАННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА . РЕАЛІЗАЦІЯ ВЕБ-ДОДАТКУ ..	21
3.1 Технічне завдання проекту	21
3.2 Опис предметної області.....	21
3.3 Вимоги до програмного продукту.....	22

					КНТЕУ 6.050103 10-11.БР			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів	Стадія	Аркуш	Аркушів
Зав. Каф.		Криворучко О.В				Зміст	2	39
Керівник		Харченко О.А				Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група		
Гарант		Цензура М.О.						
Розробив		Потоцький Д.С.						
					Зміст			

3.4Надійність програмного забезпечення.....	22
3.5 Захист інформації від несанкціонованого доступу до веб-додатку	23
3.6Проектування користувацького інтерфейсу додатку	24
3.7 Реалізація веб-додатку	25
3.8 Програмна частина веб додатку	27
Висновки по розділу 3	32
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	37
Додатки:	38
Додаток А.....	38
Додаток Б	40

ВСТУП

Метою даної роботи є розробка програмного забезпечення, що дозволяє проводити аналіз успішності студентів.

Завдання – розгляд готових варіантів програмного забезпечення та розробка власного веб-додатку з базою даних студентів.

Об'єкт дослідження - освітній процес .

Предмет дослідження – розроблені користувацькі та Інтернет додатки, що є системами для аналізу активності та успішності студентів.

Сьогодні бази даних (БД) є потужним і зручним способом зберігання і користування інформацією. Вони необхідні у величезній кількості установам, організаціям, підприємствам і навіть навчальним для ведення поточних справ. Важливість баз даних важко переоцінити. Тому вони належать до пріоритетного напрямку розвитку програмних продуктів і займають значне місце на ринку прикладних пакетів, а на їх розробку витрачаються шалені гроші. Отже бази даних - це актуально і своєчасно.

					<i>КНТЕУ 6.050103 10-11.БР</i>			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів <i>Вступ</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. Каф.		<i>Криворучко О.В</i>				<i>В</i>	<i>4</i>	<i>39</i>
Керівник		<i>Харченко О.А</i>				<i>Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група</i>		
Гарант		<i>Цензура М.О.</i>						
Розробив		<i>Потоцький Д.С.</i>						

РОЗДІЛ 1. СТВОРЕННЯ ВЕБ-САЙТІВ

1.1 Визначення веб-додатків

Веб-додаток - додаток, в якому клієнтом є користувач Інтернету, а сервером - веб-сервер. Оглядач Інтернету може бути реалізацією так званих тонких клієнтів. Він відображає веб-сторінки і, як правило, входить до складу операційної системи, а його оновлення та супровід виконує постачальник операційної системи. Логіка додатка зосереджена на сервері, а оглядача Інтернету найчастіше відповідає лише за відображення інформації, завантаженої з сервера, і за передачу на сервер даних користувача. Однією з переваг такого підходу є той факт, що клієнти не залежать від конкретної операційної системи користувача, і веб-додатки, таким чином, є міжплатформеними сервісами[9].

1.2 Особливості web-додатків:

Веб-додаток отримує запит від клієнта і виконує обчислення, після цього формує веб-сторінку і відправляє її клієнту через мережу з використанням протоколу HTTP.

Веб-додаток може бути клієнтом інших служб, наприклад, бази даних або іншого веб-додатку, розташованого на іншому сервері.

Останнім часом набуває популярності новий підхід до розробки веб-додатків, який називається Аях.

Сторінки веб-додатка не перезавантажуються повністю, а лише завантажують з сервера ЗМІНИ, що робить їх більш інтерактивними і продуктивними.

Незалежність від операційної системи клієнта

					<i>КНТЕУ 6.050103 10-11.БР</i>			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів <i>Розділ 1</i>	Стадія	Аркуш	Акрюшів
Зав. Каф.		Криворучко О.В				P1	5	39
Керівник		Харченко О.А				Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група		
Гарант		Цензура М.О.						
Розробив		Потоцький Д.С.						

Додаток створюється один раз для довільно вибраної платформи і на ній розгортається. Проте різна реалізація HTML, CSS, DOM і інших специфікацій в браузерах може викликати проблеми при розробці веб-додатків і подальшої підтримки.

Простота використання

Відвідувач сайту має протягом 10-15 секунд зрозуміти, як почати користуватися сайтом. Не сподівайтесь, що хтось читатиме інструкції або витратить 1-2 години на вивчення системи меню та команд.

Успіх веб-додаку багато в чому зобов'язаний тому, що велика частина теоретичних побудов, присвячених гіпертексту, була відкинута на користь простих прагматичних рішень, які і послужили основою ідеальної конструкції. RSS став, можливо, єдиним широко поширеним веб-сервісом саме тому, що він простий. А складні корпоративні набори все ще чекають своєї години.

Підтримуйте спрощені моделі програмування і ви отримаєте вільно-зв'язаних партнерів. Проблема корпоративних веб-сервісів в тому, що вони передбачають жорстко обумовлене партнерство. У багатьох випадках це виправдано, але частенько найцікавіші додатки можуть бути побудовані на вельми крихкій основі.

Думайте про синдикацію, а не про координацію. Прості веб-сервіси - як RSS або сервіси на базі REST - займаються синдикацією даних, не намагаючись контролювати, що відбувається з інформацією на іншому кінці ланцюжка. Ідея кризової передачі даних є однією з базових ідей самого Інтернету.

Великий час реакції на дії користувача

Від моменту натискання на посилання до появи результату часто проходить більше 1 секунди (а буває й більше). Тому людина встигає помітити затримку і навіть занудьгувати. Задача програміста -

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		6

максимально скоротити час реакції додатка. Як мінімум, ПОЧАТОК відповіді має з'явитися менш ніж за 10 секунд.

Велика кількість користувачів

В успішних веб-проектах на сайт приходять десятки користувачів на секунду. Тому варто замислитися над оптимізацією швидкості виконання програми.

Стійкість до атак

Через відкритий доступ до керування web-додатком, кількість хакерських атак на такі програми дуже велика.

Гігантські масиви даних

Бази даних з мільярдами записів перестали бути екзотикою зовсім недавно, і тому ще немає інструментів для ефективного пошуку та маніпуляції такими великими кількостями інформації.

Мережні, розподілені, паралельні обчислення, системи агентів

Сучасні додатки Internet часто спілкуються між собою. Наприклад, сайт "електронний магазин" використовує сайт банку, щоб прийняти оплату за товар. Розважальні сайти показують рекламні блоки з інших сайтів і т.д.[5].

1.3 Види веб-додатків

Веб-застосунок -розподілений застосунок,якому клієнтом виступає браузер, а сервером - веб-сервер. Браузер може бути реалізацією так званих тонких клієнтів - логіка за стосунку зосереджується на сервері, а функція браузера полягає переважно у відображенні інформації, завантаженої мережею з сервера, і передачі назад даних користувача. Однією з переваг такого підходу є той факт, що клієнти не залежать від конкретної операційної системи користувача, тому веб-застосунки є міжплатформовими сервісами. Унаслідок цієї

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		7

універсальності і відносної простоти розробки веб-застосунки стали широко популярними в кінці 1990-х - початку 2000-х років.

Існує декілька підходів для створення веб-додатків для мобільних пристроїв:

Адаптивний веб-дизайн може бути використаний для створення як від звичайного веб-сайту так й для односторінкового додатку, який зручний у використанні на пристроях з невеликими екранами.

Прогресивний веб-застосунок поєднання звичайних веб-сторінок із мобільним додатком.

Мобільний застосунок або «рідний додаток» запускається на виконання безпосередньо на мобільному пристрою без веб-браузера й зазвичай не потребують наявності Інтернет з'єднання. Типово пишуться мовою Java для Android-пристроїв або на Objective-C чи Swift для iOS. Останнім часом, такі програмні каркаси, як React Native, Flutter, Xamarin дозволяють розробляти мобільні додатки відразу для декілька мобільних платформ використовуючи одну мову програмування (як правило, одну із поширеніших, на зразок JavaScript), замість стандартних для мобільних додатків.

Гібридні додатки вбудовують веб-сайт всередину мобільного додатка. Можуть бути побудовані за допомогою гібридних програмних каркасів (framework) таких як: Apache Cordova, Ionic або Appcelerator Titanium. Цей підхід дозволяє розробникам використовувати сучасні веб-технології, разом із збереженням певних переваг саме мобільних додатків: застосування апаратного прискорення, офф-лайн операції, доступ до додатку[10].

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		8

1.4 Основні етапи створення веб- сайтів

У розробці веб-сайтів бере участь величезна кількість людей-професіоналів – це, зокрема, програмісти, верстальники та веб-дизайнери. Спочатку веб-дизайнер повинен створити дизайн майбутнього сайту, враховуючи побажання замовника, мету сайту і його передбачувану аудиторію. Підбираються відповідні шрифти, картинки, розташування об'єктів. Після цього до роботи підключається верстальник. Він, в свою чергу, працює вже з готовим дизайном і його робота заключається в оптимізації дизайну під різні платформи та веб-браузери.

Наступним кроком йде програмування. Програмісту залишається лиш створити програмну частину сайту. Це стає можливим завдяки мові розмітки HTML, javascript-сценаріям та таблицям стилів CSS. Коли додаток вже у працездатному стані, його необхідно наповнити інформацією. Окрім всіх вищеперерахованих професій також існують і інші, адже ще потрібно розкрути та оптимізувати сайт, розмістити його в інтернет-каталогах, поширити рекламу вашого сайту[3]/

1.5 Визначення проекту

Перший етап процесу створення сайту включає в себе збір та аналіз інформації, що необхідна для розуміння об'єму проекту і підготовки до розробки проекту . Потрібно мати уявлення про те:

- Хто буде використовувати веб-додаток;
- З якою цілю;
- Які дії і завдання будуть виконувати за допомогою додатку

На основі побажань клієнта та зібраної інформації розроблюється прототип сайту і технічне завдання.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		9

На підставі технічного завдання розробляється концепт веб-додатку.

Типи побудови сайту:

- Статичні сторінки, які користувач не тільки використовує, але і при необхідності сам вносить в них зміни;
- Додаток, адміністратори яких, не маючи спеціальних знань, можуть самостійно вносити зміни в зміст сторінок, але не можуть внести зміни ні в структуру, ні в дизайн сайту;[1].

1.6 Інформаційне наповнення сайту

Коли проект готовий і відомо, які саме функції будуть наявні у додатку, можна приступати до створення наповнення, тобто до потоку інформації який буде використовувати та змінювати користувач.

Інформація на сайті може подаватися у вигляді тексту, таблиць, графіки.

Також, слід пам'ятати, що:

- Неможливо наповнити додаток без участі замовника !

Інформація не може залишатися незмінним протягом усього часу існування додатку, його необхідно періодично оновлювати роблячи висновки з встигає мсті студентів[3].

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		10

РОЗДІЛ 2 ЗАСОБИ РОЗРОБКИ ПРОГРАМНОГО ПРОДУКТУ

2.1 Розробка інтерфейсу та дизайну веб- додатку

До розробки інтерфейсу та дизайну додатку також можна переходити відразу після написання основної частини проекту.

Розробка інтерфейсу та дизайну - це творчий процес. Під дизайном сайту розуміється не просто шаблон, а повне оформлення сторінок сайту в єдиному стилі. Найважливіше значення тут має не нагромадження робочого простору щоб не відволікати користувача непотрібними речами.

Також додатки мають властивість становитися менш актуальними. Це означає, що для ефективної роботи необхідно проводити оновлення додатку як мінімум один раз на рік. Переважно особливості додатку повинні бути такими:

- Відповідати іміджу навчального закладу
- Функціональна навігація;
- Правильно скомпонованим;
- Інтуїтивний дизайн
- Сторінки повинні завантажуватися швидко;
- Гармонійним, щодо підбору кольорів щоб не втомлюватися при довгій роботі.

2.2 Написання програмного коду та верстка сайту

Під написанням коду мається на увазі програмування додатку та його функцій.

					<i>КНТЕУ 6.050103 10-11.БР</i>			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів <i>Розділ 2</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Акрюшів</i>
Зав. Каф.		Криворучко О.В				P2	11	39
Керівник		Харченко О.А				Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група		
Гарант		Цензура М.О.						
Розробив		Потоцький Д.С.						

В даний час жоден додаток не обходиться без програмних розробок.

Використання програмування дозволяє зробити додаток більш функціональним, а також, що важливо, спростити його подальше використання, адже саме правильно запрограмовані додатки дають можливість власнику або особі яка за нього відповідає вносити зміни якщо вони потребуються.

Верстка сайту означає процес інтеграції тексту, графіки і програмних елементів в єдине ціле. Саме верстка надає додатку остаточний вигляд, який і побачать користувачі на сайті. Це накладає на людину яка це розробляє відповідальність так як він повинен дати користувачу зрозумілий інтерфейс в якому буде легко розібратися самостійно.

2.3 Тестування – фінальний етап створення проекту

Цей етап являє собою контроль якості роботи, що виконується. Перевіряється все: зручність використання, працездатність і наявність всіх необхідних функцій, правильність написання..

На тестування додаток, як правило, віддається людям, які не брали участь в його створенні, так як в процесі роботи помилки стають непомітні, свіжий погляд тут просто необхідний.

2.4 Засіб веб-розробки HTML

Hypertext Markup Language (HTML) (укр. Мова розмітки гіпертекстових документів) -стандартна мова розмітки для створення веб-сторінок і веб-додатків. З Cascading Style Sheets (CSS) і JavaScript, вона утворює тріаду основних технологій для World Wide Web.

Веб-браузери отримують HTML-документи з веб-сервера або з локальної пам'яті і передають документи в мультимедійні веб-сторінки.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		12

HTML описує структуру веб-сторінки семантично і спочатку включені сигнали для зовнішнього вигляду документа.

Елементи HTML є будівельними блоками сторінок HTML. За допомогою конструкцій HTML, зображення та інші об'єкти, такі як інтерактивні форми, можуть бути вбудовані у візуалізовану сторінку. HTML надає засоби для створення структурованих документів, позначаючи структурну семантику тексту, наприклад заголовки, абзаци, списки, посилання, цитати та інші елементи.

Елементи HTML окреслені тегами, написаними з використанням кутових дужок. Теги, такі як і безпосередньо вводять вміст на сторінку. Інші теги, такі як `<img />` `<input />` `<p>` оточують і надають інформацію про текст документа і можуть включати інші теги як під-елементи. Браузери не показують теги HTML, але використовують їх для інтерпретації вмісту сторінки.

HTML може вбудовувати програми, написані на мові сценаріїв, наприклад JavaScript, що впливає на поведінку та вміст веб-сторінок. Включення CSS визначає вигляд і компонування вмісту. World Wide Web Consortium (W3C), які супроводжують як HTML і CSS стандартів, заохочує використання CSS над явним презентаційним HTML з 1997 року.

HTML впроваджує засоби для:

- створення структурованого документа шляхом позначення структурного складу тексту: заголовки, абзаци, списки, таблиці, цитати та інше;
- отримання інформації із Всесвітньої мережі через гіперпосилання;
- створення інтерактивних форм;
- включення зображень, звуку, відео, та інших об'єктів до тексту.[2]

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		13

2.5 База даних

База даних (англ. database) – сукупність даних, організованих відповідно до концепції, яка описує характеристики цих даних і взаємозв'язки між їх елементами; ця сукупність підтримує щонайменше одну з областей застосування (за стандартом ISO/IEC 2382:2015).

В загальному випадку база даних містить схеми, таблиці, подання, збережені процедури та інші об'єкти. Дані у базі організовують відповідно до моделі організації даних. Таким чином, сучасна база даних, крім саме даних, містить їх опис та може містити засоби для їх обробки.

В загальному випадку базою даних можна вважати будь-який впорядкований набір даних. Наприклад, паперову картотеку з формулярами про працівників підприємства у відділі кадрів. Але дана стаття зосереджена на використанні баз даних в інформаційних системах. На даний час застосунки для роботи з базами даних є одними з найпоширеніших прикладних програм[10]

2.6 Мова програмування C#

C# (вимовляється Сі-шарп) - об'єктно-орієнтована мова програмування з безпечною системою типізації для платформи .NET. Розроблена Андерсом Гейлсбергом, Скотом Вілтамутом та Пітером Гольде під егідою Microsoft Research (при фірмі Microsoft).

Синтаксис C# близький до C++ і Java. Мова має строгу статичну типізацію, підтримує поліморфізм, перевантаження операторів, вказівники на функції-члени класів, атрибути, події, властивості, винятки, коментарі у форматі XML. Переїнявши багато що від своїх попередників - мов C++, Delphi, Модула і Smalltalk - C#, спираючись на

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		14

практику їхнього використання, виключає деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем, наприклад множинне спадкування класів (на відміну від C++).

C# підтримує строго типізовані неявні оголошення змінних з ключовим словом `var` і неявно типізовані масиви з ключовим словом `new []`, за яким слідує ініціалізатор колекції.

C# підтримує суворий тип даних `Boolean`, `bool`. Вирази, які приймають умови, такі як `while` та `if`, вимагають висловлювання, що реалізує оператор `true` або `false`. Хоча C++ також має тип `Boolean`, він може бути вільно перетворений в цілі числа та з них, а вирази, такі як `if(a)`, вимагають тільки того, щоб `a` був конвертований в `bool`, що дозволяє бути `a` `int`-типу або вказівником. C# забороняє «ціле значення означає справжній або помилковий підхід» на тій підставі, що примус програмістів використовувати вирази, які повертають точно `bool`, можуть створювати деякі типи помилок програмування, наприклад `if (a = b)` (використання присвоювання `=` замість рівності `==`, які, хоча і не є помилкою на C або C++, все одно будуть спіймані компілятором).

C# безпечніший в порівнянні з C++. Єдиними неявними перетвореннями за умовчужанням є ті, які вважаються безпечними, наприклад, розширення цілих чисел. Це застосовується під час компіляції, під час JIT і, в деяких випадках, під час виконання. Не відбувається неявних перетворень між булевими і цілими числами, а також між членами перерахування і цілими числами (крім літерала 0, який може бути неявно перетворений в будь-який нумерований тип). Будь-яке призначене для користувача перетворення повинно бути явно

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		15

позначене як явне або неявне, на відміну від конструкторів копіювання C++ і операторів перетворення, які за умовчанням є неявними.

C# має явну підтримку коварианції та контраваріантності в родових типах, на відміну від C++, яка має певний рівень підтримки контраваріантності просто через семантику типів, що повертаються, на віртуальні методи.

Члени перерахування розміщуються в своєму власному обсязі.

Мова C# не допускає глобальних змінних або функцій. Всі методи і члени повинні бути оголошені всередині класів. Статичні члени відкритих класів можуть замінювати глобальні змінні та функції. [10]

2.7 SQL

SQL - це діалогова мова програмування для здійснення запиту і внесення змін до бази даних, а також керування базами даних. Багато баз даних підтримує SQL з розширеннями до стандартної мови. Ядро SQL формує командна мова, яка дозволяє здійснювати пошук, вставку, оновлення і вилучення даних за допомогою використання системи керування і адміністративних функцій. SQL також включає CLI (Call Level Interface) для доступу і керування базами даних дистанційно.

Мова SQL поділяється на кілька видів елементів:

- Пункти (диз'юнкти) (англ. Clauses), що є складовими частинами інструкцій та запитів. (Іноді вони не обов'язкові.)^[3]
- Вирази (англ. Expressions), які можуть генерувати скалярні значення, або таблиці з стовпчиками і рядками даних
- Предикати (англ. Predicates), які описують умови, результатом яких є значення тризначної логіки SQL (true/false/unknown) або Булеві значення істинності і які використовуються для обмеження ефекту інструкцій та запитів, або для зміни потоку виконання програми.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		16

- Запити (англ. Queries), які отримують дані на основі заданих критеріїв.
- Інструкції (англ. Statements), які чинять дію на схему даних чи самі дані, або контролюють транзакції, потік виконання програми, з'єднання, сесії, та виконують діагностику.
 - Інструкції SQL також включають крапку з комою (";") для позначення кінця інструкції. Хоча вона не є обов'язковою на кожній платформі, вона описується як стандартна частина граматики SQL.
- Незначими пропуски загалом ігноруються в інструкціях і запитах SQL, дозволяючи формувати код SQL з метою покращення читабельності. [10]

2.8 Фреймворк програмний каркас

Фреймворк (англ. Framework, каркас, платформа, структура, інфраструктура^[1]) - інфраструктура програмних рішень, що полегшує розробку складних систем. Спрощено дану інфраструктуру можна вважати своєрідною комплексною бібліотекою, але при цьому вона має ряд обмежень, що задають правила створення структури проекту та написання коду.

Програмний фреймворк (англ. software framework) - це готовий до використання комплекс програмних рішень, включаючи дизайн, логіку та базову функціональність системи або підсистеми. Відповідно - програмний фреймворк може містити в собі також допоміжні програми, деякі бібліотеки коду, скрипти та загалом все, що полегшує створення та поєднання різних компонентів великого програмного забезпечення чи швидке створення готового і не обов'язково об'ємного програмного продукту. Побудова кінцевого продукту відбувається, зазвичай, на базі єдиного API.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		17

Одна з головних переваг, при використанні каркасних застосунків, полягає в тому, що такі програми мають стандартну структуру. Каркаси за стосунків стали популярними з появою елементів інтерфейсу, які мали тенденцію до реалізації стандартної структури для додатків. З їх використанням стало набагато простіше створювати засоби для автоматичного створення графічних інтерфейсів, оскільки структура внутрішньої реалізації коду програми стала відома заздалегідь.

Для забезпечення каркасу, зазвичай, використовують підходи об'єктно-орієнтованого програмування, наприклад, частини програми можуть успадковуватися від базових класів фреймворка. [1]

2.9 ADO.NET

ADO.NET (ActiveX Data Objects .NET) - це набір бібліотек, що поставляється з Microsoft .NET Framework і призначений для взаємодії з різними сховищами даних з .NET-застосунків. Бібліотеки ADO.NET включають класи для приєднання до джерела даних, виконання запитів і обробки їхніх результатів. Крім того, ADO.NET можна використовувати як надійний, ієрархічно організований, відокремлений кеш даних для автономної роботи з даними.

Підтримка XML

ADO також підтримує XML, але не буде так само ефективно обробляти XML-дані, як це робить ADO.NET, оскільки ADO.NET створювався з врахуванням XML, а ADO-ні.

Простота модифікації

Протягом терміну служби системи в неї можна вносити незначні зміни, однак спроби провести архітектурні зміни трапляються рідко, через виняткову складність завдання. На жаль, при природному розвитку подій такі зміни іноді виявляються необхідними.

Простота програмування

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		18

Компоненти даних ADO.NET в Visual Studio інкапсулюють функціональні можливості доступу до даних різними способами, що допомагає розробляти програмні продукти значно швидше і з меншою кількістю помилок.

Продуктивність

Для непідключених застосунків набори даних ADO.NET дають вигаш в продуктивності в порівнянні з непідключеними наборами записів ADO. Передача непідключеного набору записів між рівнями за допомогою COM-упаковки може призвести до великої витрати обчислювальних ресурсів, тому що значення в наборі записів перетворюються до типів даних, відомих COM. У ADO.NET таке перетворення типів даних не потрібно. [10]

Висновки по розділу 2

У діловій або особистій сфері часто доводиться працювати з даними з різних джерел, кожне з яких пов'язане з певним видом діяльності. Для координації всіх цих даних необхідні певні знання й організаційні навички. Бази Даних об'єднують відомості з різних джерел.

Створювані форми, запити і звіти дозволяють швидко і ефективно оновлювати дані, отримувати відповіді на питання, здійснювати пошук потрібних даних, аналізувати дані, друкувати звіти, діаграми та ін.

Саме тому обрана тема щільно пов'язана з освітнім процесом мого навчального закладу а саме КНТЕУ .

Оскільки в даному навчальному закладі навчається дуже багато студентів та працює багато викладачів потрібно якось оптимізувати частину документообігу пов'язаного з активністю студентів та їх успіхами у навчанні. Роблячи з цього певні висновки я вирішив розробити веб-додаток щоб оптимізувати та прискорити роботу на

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		19

кафедрах . Саме тому обрана тема щільно пов'язана з освітнім процесом мого навчального закладу а саме КНТЕУ .

Оскільки в даному навчальному закладі навчається дуже багато студентів та працює багато викладачів потрібно якось оптимізувати частину документообігу пов'язаного з активністю студентів та їх успіхами у навчанні . Роблячи з цього певні висновки було прийнято рішення розробити веб-додаток щоб оптимізувати та прискорити роботу на кафедрах .

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		20

РОЗДІЛ 3. ТЕХНІЧНЕ ЗАВДАННЯ ПРОЕКТУ. ПРОЕКТУВАННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА . РЕАЛІЗАЦІЯ ВЕБ-ДОДАТКУ

3.1 Технічне завдання проекту

3.2 Опис предметної області

Університет – навчальний заклад що пропонує вищу освіту бакалавра та магістра . На відміну від менш серйозних навчальних закладів.

Основною метою діяльності навчального закладу є надання освіти згідно особливостей спеціальності . Пропонуючи широкий спектр навчальних програм популярних спеціальностей, університети є раціональною альтернативою Інтернет навчанню та спеціалізованим курсам . Через ширший вибір додаткових предметів студент університету має можливість обрати додаткові напрямки свого розвитку профільної спеціальності або додаткові предмети які дозволять розвинути додаткові навички.

Предметна область бази даних являє собою навчальний заклад а саме університет , що має кафедру, деканат , розділений на факультети , викладачі , предмети , методисти , кожен с переліченого списку виконує відповідні функції.

					КНТЕУ 6.050103 10-11.БР			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів	Стадія	Аркуш	Аркушів
Зав. Каф.		Криворучко О.В				P2	21	39
Керівник		Харченко О.А				Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група		
Гарант		Цензура М.О.						
Розробив		Потоцький Д.С.						
					Розділ 3			

Основні об'єкти навчального закладу :

- а) Викладач
- б) Деканат
- в) Персонал університету
- г) Факультет
- д) Студент
- е) Предмет

Навчання студентів та фіксація їх активності є одною з основних робіт навчального закладу.

3.3 Вимоги до програмного продукту

До основних функціональних вимог системи належать:

- здатність вводити та накопичувати інформацію про студентів викладачів, лекційні заняття ,практичні,лабаторні
- можливість редагувати інформації якщо того потребує ситуація;
- обробка інформації та формуванням звітів ;
- перелік складу групи ;
- перелік занять

Технічні вимоги полягають у:

- якість розробки ;
- передбачення ряду заходів, необхідних для ефективної роботи програми в межах програмно-апаратного комплексу.

3.4 Надійність програмного забезпечення

В даний час ми є свідками зближення управлінських та інформаційних технологій.

Інформаційна технологія — це комплекс взаємопов'язаних, наукових, технологічних, інженерних дисциплін, що вивчають методи ефективної організації праці людей, що зайняті обробкою і зберіганням інформації.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		22

Навчальні надають студенту можливість отримати не тільки диплом про закінчення навчального закладу, він надає можливість оволодіти корисними навичками та покращити їх володіння.

Успіх документообігу в значній мірі залежить від ефективного використання, в процесі використання, інформаційних ресурсів.

Тому перехід на нові форми організації документообігу повинен супроводжуватись збільшенням надійності обліку й контролю.

Для цього необхідна комп'ютерна система, яка може зберігати потрібний користувачу об'єм інформації в зручному для нього вигляді. Вона має відповідати вимогам комплексної автоматизації підрозділів, таких її ділянок як:

- Аналіз успішності студентів
- Зберігання інформації стосовно занять ;
- Редагування інформації;
- Зберігання переліку студентів.

Для таких систем джерелом інформації є спеціалізоване обладнання: персональні комп'ютери користувачів або телефони з доступом до мереж Інтернет. Але в основі автоматизованої системи лежить спеціальний програмний продукт для автоматизації навчальних процесів , що базується на системі автоматизованого документного обліку інформації.

3.5 Захист інформації від несанкціонованого доступу до веб-додатку

Великі навчальні заклади із розвинутими комп'ютерними інформаційними системами закладу мають скласти власні внутрішні стандарти з організації безпеки та захисту інформації.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		23

Завдання організації захисту та безпеки даних в закладі полягає в тому щоб зберегти її первозданність від зовнішніх несанкціонованих втручань.

Специфіка використання комп'ютерної техніки передбачає особливі методи забезпечення захисту облікової інформації.

Великою проблемою є забезпечення захисту інформації та створення електронних архівів документів у разі випадкового або навмисного пошкодження технічних засобів або електронних носіїв інформації.

Основне завдання захисту облікової інформації полягає у необхідності визначення:

- обмеженого кола осіб, які мають доступ до певного виду як електронної, так і інформації на паперових носіях;
- осіб, які мають право доступу до цієї інформації;
- місця зберігання цієї інформації;
- правил поведінки при роботі з інформацією.

Головною метою безпеки облікової інформації закладу є гарантування його стабільного максимально ефективної роботи та розвиток.

3.6Проектування користувацького інтерфейсу додатку

Оскільки якість процесу інтерактивної взаємодії користувача із системою пов'язана з характеристиками людини, можливості сприйняття візуальної інформації, то при розробці інтерфейсу необхідно взяти до уваги, що:

- інтерфейс – сама важлива частина системи підтримки прийняття рішень з точки зору безпосереднього користувача системи, який може працювати з нею по декілька годин поспіль;

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		24

- який саме конкретний тип інтерфейсу можна створити за допомогою вибраних засобів

Основними властивостями, яким повинні задовольняти інтерфейси, є такі:

- а) адаптованість
- б) не завантаженість інтерфейсу
- в) зручність
- г) простота інтерфейсу
- д) гнучкість інтерфейсу

Адаптованість означає, що інтерфейс повинен бути сумісним з потребами та можливостями користувача.

Не завантаженість інтерфейсу означає, що візуалізація певною інформації не повинна відволікати або заплутувати користувача під час роботи з додатком

Зручність інтерфейсу – це максимальна простота використання і готовність в задовольнити запити користувача.

Гнучкість інтерфейсу – це можливість його адаптування до розв’язання конкретної задачі.

При цьому простота означає, що інтерфейс не повинен бути перевантажений деталями щодо представлення розв’язку поставленої задачі.

3.7 Реалізація веб-додатку

Реалізація клієнт серверної взаємодії Мовою HTML я розробив головну сторінку, а через CSS я надав їй вигляд схожий із середовищем 1С.

Наступною задачею було вирішення із перезавантаженням сторінки, бо с кожною зміною у таблиці сторінці потрібно відправити на

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		25

сервер запит, а для цього потрібно заново завантажувати сторінку для виводу оновленої таблиці.

Для вирішення цієї проблеми я завантажив на сайт бібліотеку. В ній є функція AJAX яка при натисканні на кнопку буде відправляти на сервер дані з форми, а відповідь буде вставляти на сторінку без потреби у перезавантаженні

Модель клієнт-сервер використовується для розподілу завдань у мережі. Ця концепція надає користувачеві багато переваг, перерахованих нижче:

Центральне адміністрування: сервер знаходиться в центрі мережі, щоб він мав доступ до всіх користувацьких ресурсів. Технічне обслуговування також дуже просте: оновлення програмного забезпечення виконується тільки на серверах, так що клієнти зазвичай не помічають.

Економія ресурсів: оскільки дані зберігаються централізовано та доступні в будь - який час, не потрібно зберігати ці самі дані на різних клієнтських комп'ютерах.

Більша безпека доступу: централізоване зберігання даних дозволяє легко контролювати доступ.

Розширювана мережа: додавання та видалення клієнтів можливе, не впливаючи на роботу мережі та не вимагаючи серйозних змін.

Не залежить від місця розташування.

Висока надійність: сервери дуже відмово стійкі завдяки системі RAID .

Архітектура Clie / Sever - це системний дизайн, в якому обробка додатку поділяється на дві окремі частини. Одна частина працює на сервері ,інша - на робочій станції (на пристрої користувача). Обидві частини об'єднуються в мережі через мережі.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		26

3.8 Програмна частина веб додатку

Серверна частина веб-додатку виконана за допомогою сервера на базі фреймворків для веб-додатків, створений з використанням C# та бази даних SQL.

Користувач (викладач) заходить під своїми ідентифікаційними даними (логін та пароль) у систему та має змогу додавати ,переглядати та редагувати інформації. Все це стало можливим завдяки застосуванню бази даних SQL.

Але для того щоб це все працювало та було зручно користувачу створений користувацький інтерфейс у виді веб-сторінки, за допомогою мови розмітки HTML. Для того щоб веб-сторінка мала неодноманітне форматування було використано таблицю стилів CSS.

Задля інтерактивності (, авторизації, ідентифікації) було прийнято рішення про використання ADO.NET.

Для швидшого та ефективного налаштування веб-додатку були використані популярні та набір інструментів HTML,ASP.NET,ADO.NET для створення веб-сайтів та веб-додатків ASP.NET.

У додатку А представлений код додатку під час авторизації. У додатку Б –HTML-скрипт для генерації таблиці з даними.

В представлених нижче зображеннях на рисунках 3.1; 3.2; 3.3, 3.4; 3.5; 3.6; 3.7; зображений інтерфейс деякі функції веб-додатку.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		27

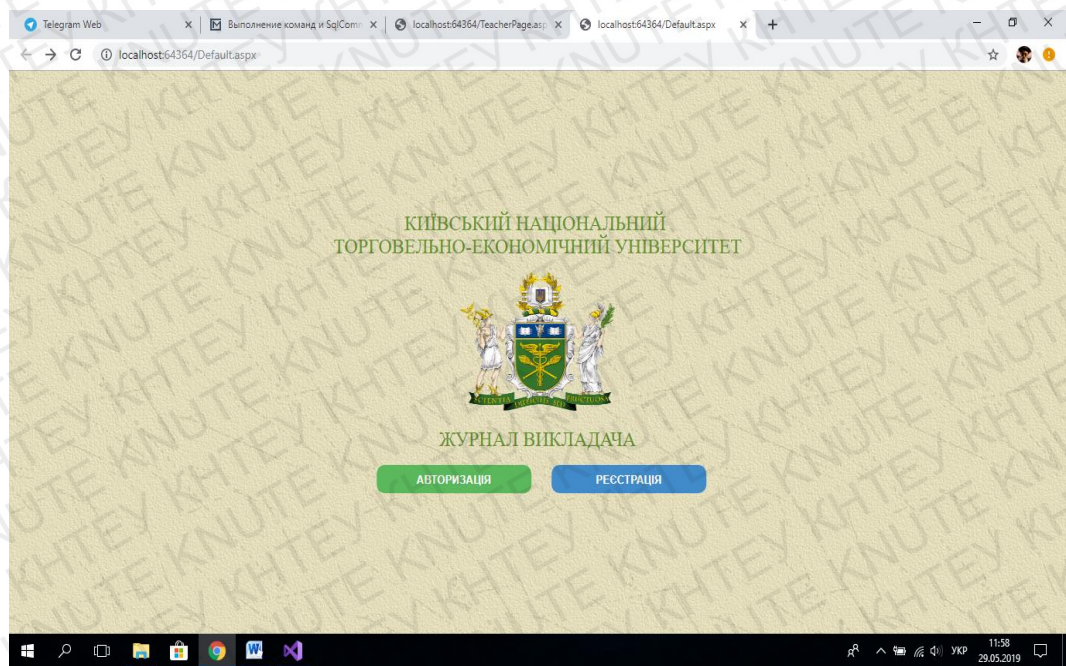


Рис. 3.1 – Стартова сторінка

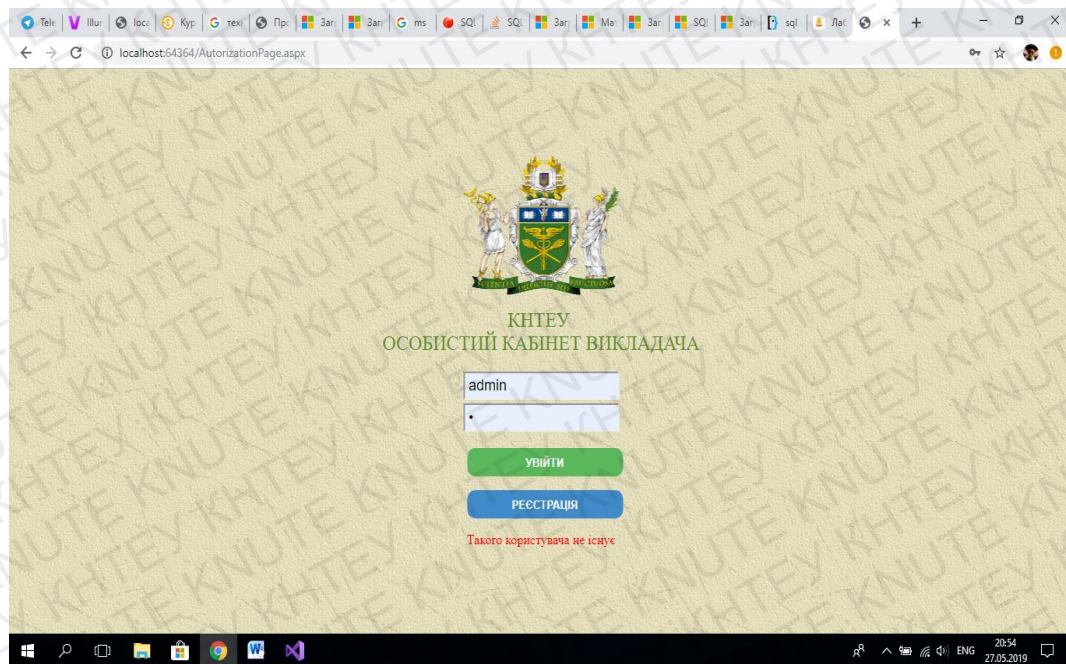


Рис. 3.2 – Відмова при авторизації

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		28

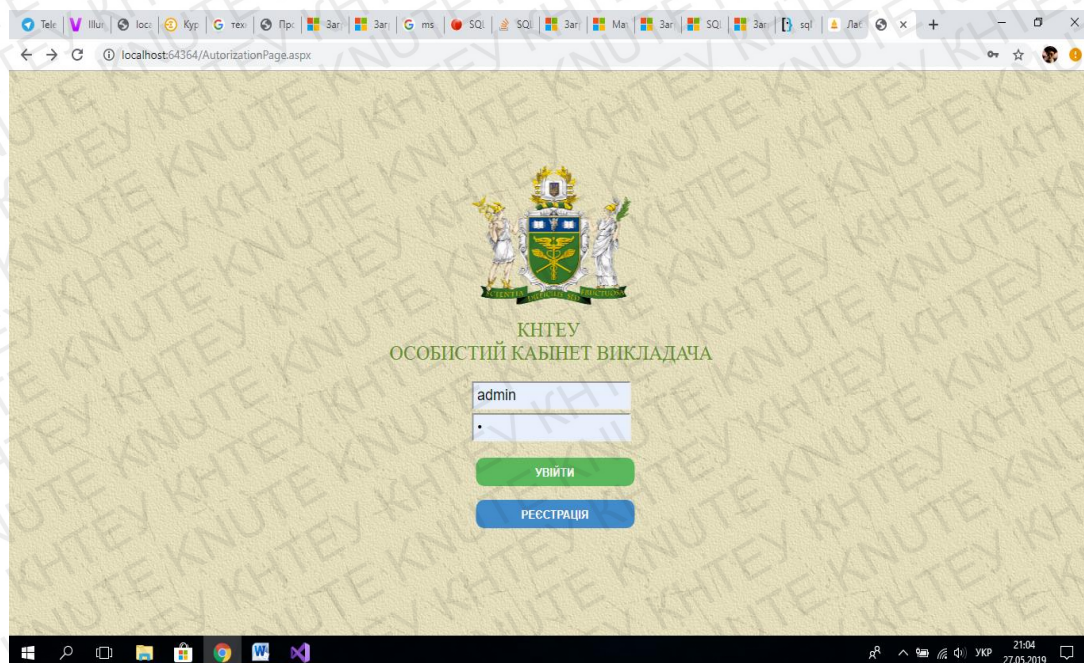


Рис. 3.3 – Сторінка авторизації

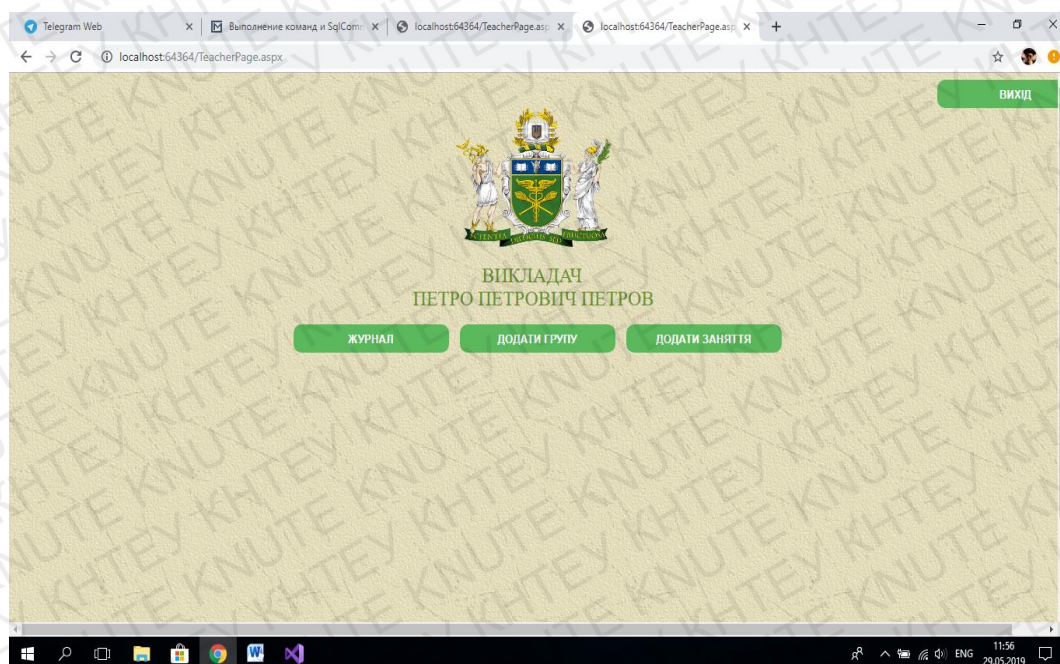


Рис. 3.4 – Сторінка успішної авторизації

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		29

TEACHERS:

	id_teacher	id_degree	firstName	middleName	lastName	phone	mail
Редагувати Видалити	1	1	Олександр	Олександрович	Олександров	11111	mail
Редагувати Видалити	2	2	Петро	Петрович	Петров	22222	yandex
Редагувати Видалити	3	3	Олег	Олегович	Олегов	33333	google
Редагувати Видалити	4	4	Роман	Романович	Романов	44444	ua
Додати							

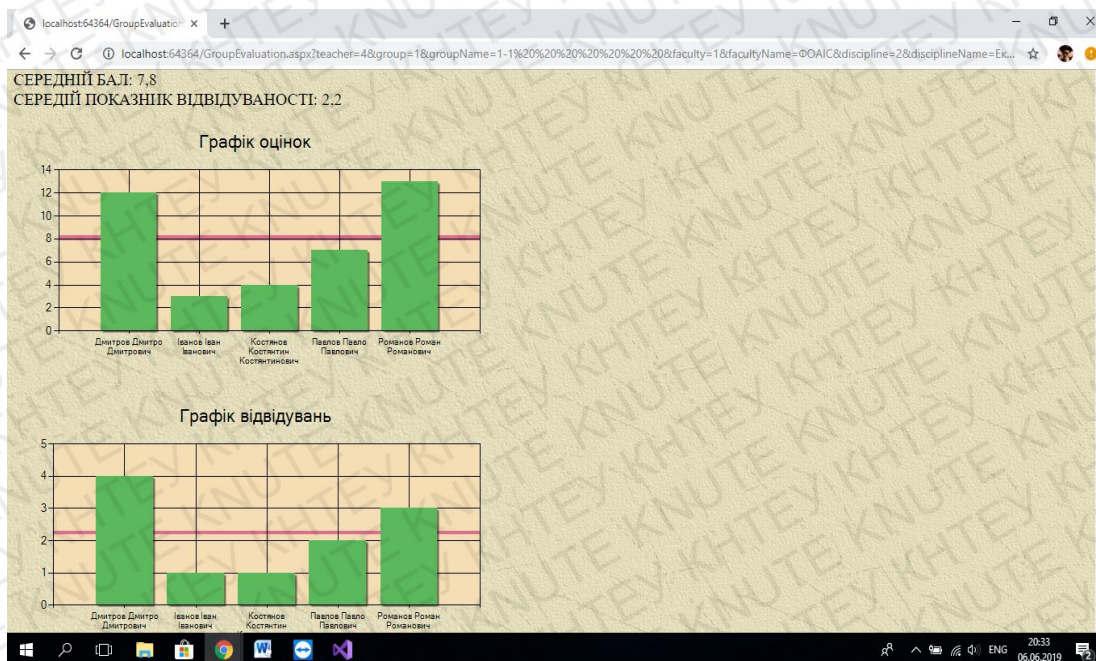
STUDENTS:

	id_student	id_group	firstName	middleName	lastName	phone	mail
Редагувати Видалити	1	1	Дмитро	Дмитрович	Дмитров		
Редагувати Видалити	2	1	Іван	Іванович	Іванов		
Редагувати Видалити	3	1	Роман	Романович	Романов		
Редагувати Видалити	4	1	Костянтин	Костянтинович	Костянов		
Редагувати Видалити	5	1	Павло	Павлович	Павлов		
Редагувати Видалити	6	2	Студент1	СтудентБ1	СтудентП1		
Редагувати Видалити	7	2	Студент2	СтудентБ2	СтудентП2		
Редагувати Видалити	8	2	Студент3	СтудентБ3	СтудентП3		
Додати							

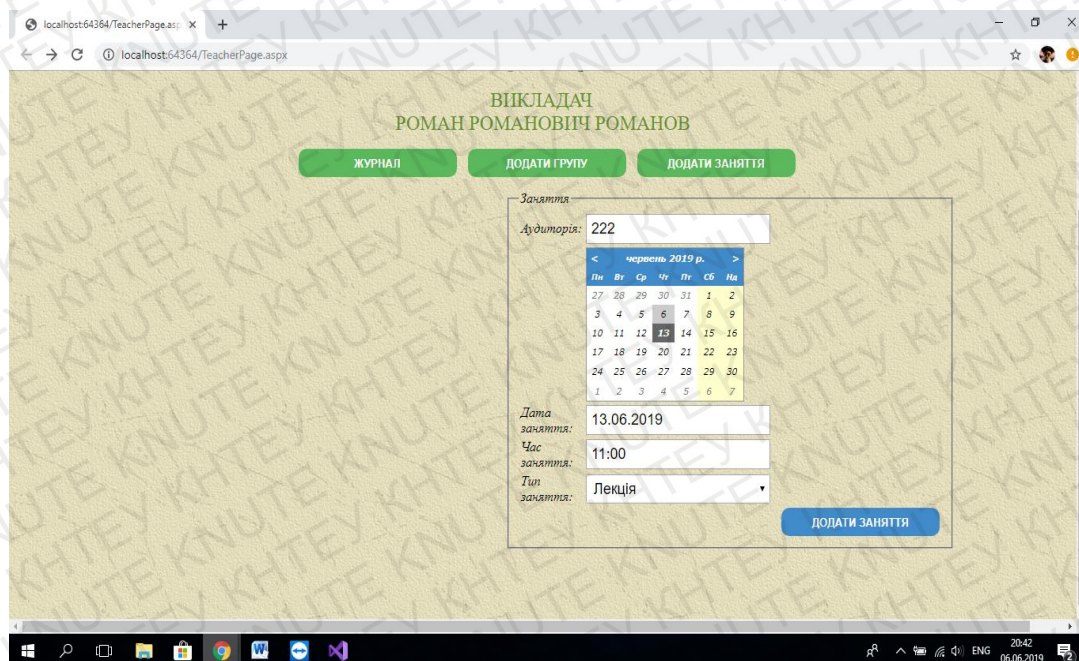
GROUPS:

	id_group	id_faculty	groupName	faculty	id_educational_department
--	----------	------------	-----------	---------	---------------------------

Рис. 3.5 – Сторінка адміністратора з таблицями вчителів та студентів



3.6 функція автоматичного підрахунку середнього балу студентів та відвідувань занять



3.7 функція додавання занять з вибором часу дати аудиторії та типу занять : лекція,лабораторна ,практична.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		31

Висновки по розділу 3

Основними властивостями, які повинні мати інтерфейси програмного продукту, є такі: адаптованість, достатність, дружність інтерфейсу, гнучкість інтерфейсу та простота інтерфейсу.

В даний час кожен власник повинен створити систему заходів безпеки, яка б сприяла виявленню ознак можливих правопорушень та злочинних дій на різних стадіях на етапі формування злого наміру та розробки планів злочинних дій, що дозволило б вчасно попередити та знешкодити злочинні наміри. Це означає, що захист інформації на підприємстві, в тому числі й облікової, повинен бути на першому місці.

Тому, перехід на нові форми організації повинен супроводжуватись збільшенням надійності.

Вона має відповідати вимогам комплексної автоматизації.

Отже для коректної роботи програмного забезпечення, в даному випадку, веб-додатку варто приділити увагу не тільки програмній реалізації, але й надійності, захисту інформації від несанкціонованого доступу, дотримуватися вимог до структури та складу програмного комплексу, до функцій, що виконуються системою, вимог до технічного, програмного та інформаційного забезпечення, вимог до організаційного забезпечення, ергономіки і технічної естетики.

Був покращений інтерфейс додатку були підібрані більш приємні для очей кольори щоб не відволікали від роботи та невтомлювались очі , Інтерфейс та дизайн буди переосмислені повністю та було прийнято зробити його частково інтуїтивним для прискорення звикання персоналу до даного додатку .

У даному додатку реалізовані та покращені нові функції такі як : реєстрація, авторизація , редагування оцінок , створення предметів по категоріям , вибір факультету, створення нових груп та ін.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		32

База даних створена в СУБД “MySQL” в середовищі “OpenServer”. Головною перевагою даного СУБД є легкість в доступі, швидкий прийом та передача запитів. При розробці баз даних я оперував інформацією отриманою з курсу лекцій по предмету “Програмування Інтернет ”. Метою моєї роботи було розробити базу даних для навчального закладу в якому и відображались активність студентів. Предметною областю в моїй роботі були: середовище “OpenServer”, СУБД та середовище розробки бази даних “MySQL” та текстовий редактор “Sublime Text”.

Для веб-

додатку були поставлені такі основні задачі:

1. Виведення повної бази даних яку можна легко прочитати та розібрати;
2. Додавання нової інформації в таблиці без зайвих зусиль;
3. Змінення інформації у полі в разі помилки при заповненні;
4. Видалення зайвої інформації у разі потреби;

Збереження цілісності таблиць та зв'язків між ними.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		33

ВИСНОВКИ ДО ВИПУСКНОГО КВАЛІФІКАЦІЙНОГО ПРОЕКТУ

Аби не нагромаджувати робоче поле кожна таблиця повинна виводитись окремо від інших, а щоб перехід між ними був швидкий, інтерфейс повинен бути зрозумілим, а також знаходитись завжди на видному місці.

Виведення бази даних повинно бути лише у візуальному варіанті, аби користувач не зміг випадково зробити щось незаплановане з таблицею. Для маніпулювання таблицями було вирішено додати 3 спеціальних кнопки, кожна з яких виконувала свою функцію.

Кнопка “Додати” для доповнення інформацією таблиць, “Змінити” для виправлення помилок, або при зміні інформації та кнопка “Редагувати” для вилучення рядків з таблиці.

На підставі технічного завдання пишеться проект сайту, в рамках якого розробляється функціональна структура сайту, навігація, визначається, які сторінки будуть статичними, які динамічним.

Коли проект готовий і відомо, які саме сторінки будуть присутні на сайті, можна приступати до створення контенту, тобто до інформаційного наповнення сайту.

Розробка веб-дизайну - це творчий процес. Під дизайном сайту розуміється не просто шаблон, а повне оформлення сторінок сайту в єдиному стилі.

Під написанням коду мається на увазі програмування сайту і його верстку.

					<i>КНТЕУ 6.050103 10-11.БР</i>			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів <i>Висновки та пропозиції</i>	Стадія	Аркуші	Акрушів
Зав. Каф.		Криворучко О.В				ВП	34	39
Керівник		Харченко О.А				Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група		
Гарант		Цензура М.О.						
Розробив		Потоцький Д.С.						

Тестування являє собою контроль якості виконуваної роботи. Перевіряється все: зручність навігації, працездатність і наявність всіх необхідних посилань, орфографія і пунктуація.

До основних функціональних вимог системи належать:

- можливість розраховувати підсумки відвідувань занять та середнього балу;
- редагування та додання інформації

Технічні вимоги полягають у:

- якості програмування (алгоритмів обробки даних);
- передбаченні ряду сервісних параметрів, необхідних для ефективної роботи програми в межах програмно-апаратного комплексу.

Перехід на нові форми організації документообігу повинен супроводжуватись збільшенням надійності обліку й контролю.

Для цього необхідна комп'ютерна система, яка може оновляти швидко інформації для того щоб вона не втрачала свою актуальність та обробляти великий обсяг інформації. Вона має відповідати вимогам комплексної автоматизації документообігу.

Завдання організації захисту та безпеки даних в документообігу полягає в забезпеченні всього комплексу заходів та кадрової роботи, спрямованого на збереження комерційної таємниці та належного контролю роботи облікових працівників.

Специфіка використання комп'ютерної техніки передбачає особливі методи забезпечення захисту облікової інформації.

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		35

Основними властивостями, яким повинні задовольняти інтерфейси, є такі:

- а) адаптованість
- б) не завантаженість інтерфейсу
- в) зручність
- г) простота інтерфейсу
- д) гнучкість інтерфейсу

Створений веб-додаток і підключена до моєї бази даних в СУБД MySQL, виконує усі прості функції, які потрібні для користувача даного додатку.

					КНТЕУ 6.050103 10-11.БР	Лист
						36
Изм.	Лист	№ Документа	Подпись	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Люк Веллинг, Томсон. Разработка web-приложений с помощью PHP и MySQL. [Текст] / Веллинг Люк и др. – Вильяме, 2010. – 848с.
2. Шклярський С.М. Прикладний інтернет для економістів : навч. посіб. - Київ : КНТЕУ, 2009. - 121с.
3. Роберт Шелдон, Джоффри Мойе. MySQL. Базовый курс. [Текст] / Шелдон Р. и др. – Вильяме, Диалектика, 2007. – 880с.
4. Інтернет ресурси:
5. Testing-Board Testautomatisierung Definition, Tutorial und Artikel Testautomatisierung [Електронний ресурс]. – [Режим доступу]: <https://www.testing-board.com/testautomatisierung/> - International
6. Realm Studio [Електронний ресурс]. – [Режим доступу]: <https://realm.io/products/realm-studio/>
7. Start up [Електронний ресурс]. – [Режим доступу]: <https://www.startupticker.ch/en/news/june-2017/startup-projekte-haben-immer-dieselben-schwachen>
8. SAP [Електронний ресурс]. – [Режим доступу]: <https://www.sap.com/index.html>
9. Сайт компанії Google [Електронний ресурс]. – [Режим доступу]: <https://www.google.com>
10. Вільна енциклопедія вікіпедія [Електронний ресурс]. – [Режим доступу]: <https://uk.wikipedia.org/wiki>

					КНТЕУ 6.050103 10-11.БР			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення аналізу успішності студентів Список використаних джерел	Стадія	Аркуш	Аркушів
Зав. Каф.		Криворучко О.В				СД	37	39
Керівник		Харченко О.А				Факультет обліку, аудиту та інформаційних систем, 4 курс, 10 група		
Гарант		Цензура М.О.						
Розробив		Потоцький Д.С.						

Додатки:

Додаток А

```
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <link rel="stylesheet" runat="server" href="~/css/styles.css" />
</head>
<body>
  <form id="form1" runat="server">
    <div class="div">
      <center>
        
        <p>
          <small class="headerText">
            КНТЕУ <br /> Особистий Кабінет Викладача
          </small>
        </p>

        <table>
          <tr>
            <td>
              <asp:TextBox ID="txtLogin" runat="server" CssClass="control"
                AutoCompleteType="Disabled" placeholder="Лорін" Height="33px"
                Width="200px"/>
            </td>
          </tr>
          <tr>
            <td>
              <asp:TextBox ID="txtPassword" runat="server" CssClass="control"
                TextMode="Password" AutoCompleteType="Disabled"
                placeholder="Пароль" Height="33px" Width="200px"/>
            </td>
          </tr>
        </table>

        <p>
```

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		38

```

        <asp:Button ID="btnAccept" Text="Увійти" runat="server"
OnClick="btnAccept_Click" CssClass="button-enter"
EnableTheming="True"/>
    </p>

    <p>
        <asp:Button ID="btnRegistration" Text="Реєстрація" runat="server"
OnClick="btnRegistration_Click" CssClass="button-registration"
EnableTheming="True"/>
    </p>

    <asp:Label ID="lbError" Text="Такого користувача не існує"
ForeColor="Red" Visible="false" runat="server" />

    </center>
</div>
</form>
</body>
</html>

```

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		39

Додаток Б

```

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <link rel="stylesheet" runat="server" href="~/css/styles.css" />
</head>
<body>
  <form id="form1" runat="server">
    <asp:ScriptManager ID="ScriptManager1" runat="server" />

    <asp:UpdatePanel runat="server">
      <ContentTemplate>
        <div style="height: 30%; width: 100%;">
          <asp:Button ID="btnExit" Text="Вихід" runat="server"
OnClick="btnExit_Click" CssClass="button-enter-1" EnableTheming="True"
/>

          <center>
            
            <p>
              <asp:Label Text="Викладач" runat="server" CssClass="headerText"/>
              <br />
              <asp:Label ID="lbTeacherFio" runat="server"
CssClass="headerText"/>
            </p>
            <p>
              <asp:Button ID="btnShedule" Text="Журнал" runat="server"
OnClick="btnShedule_Click" CssClass="button-enter"
EnableTheming="True"/>
              <asp:Button ID="btnGroupAdding" Text="Додати групу"
runat="server" OnClick="btnGroupAdding_Click" CssClass="button-enter"
EnableTheming="True"/>
              <asp:Button ID="btnLessonAdding" Text="Додати заняття"
runat="server" OnClick="btnLessonAdding_Click" CssClass="button-enter"
EnableTheming="True"/>
              <%--Переглянути--%>
            </p>
          </center>
        </div>
      </ContentTemplate>
    </asp:UpdatePanel>
  </form>
</body>
</html>

```

					КНТЕУ 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		40

```

<div class="div-left">

    <asp:Label ID="lbDisciplinesList" Text="Список дисциплин:"
    runat="server" CssClass="label" Visible="false" />

    <br />

    <asp:DropDownList ID="dropDownListDisciplines" runat="server"
    DataSourceID="SqlDataSourceDisciplines"
    DataTextField="discipline"
    DataValueField="id_discipline"

    OnSelectedIndexChanged="dropDownListDisciplines_SelectedIndexChange
    d"

    AutoPostBack="true"
    Visible="false"
    CssClass="control">
</asp:DropDownList>

    <asp:SqlDataSource ID="SqlDataSourceDisciplines" runat="server"
    ConnectionString="<=%$
    ConnectionStrings:ConnectionStringTeacherJournal %>"
    SelectCommand="
    SELECT 0 as id_discipline, N'(не обрано)' as discipline

    UNION

    SELECT [Disciplines].[id_discipline], [Disciplines].[discipline]
    FROM [Plan]
    LEFT JOIN [Disciplines] on [Disciplines].[id_discipline] =
    [Plan].[id_discipline]
    WHERE [Plan].[id_teacher] = @teacher">
    <SelectParameters>
    <asp:SessionParameter SessionField="teacher" Name="teacher"
    Type="Int32" />
    </SelectParameters>
</asp:SqlDataSource>

    <br />
    <br />

```

					KHTEY 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		41

```

<asp:Label ID="lbFacultiesList" Text="Список факультетів:"
runat="server" Visible="false" CssClass="label" />

<br />

<asp:DropDownList ID="dropDownListFaculties" runat="server"
DataSourceID="SqlDataSourceFaculties"
DataTextField="facultyName"
DataValueField="id_faculty"
AutoPostBack="true"

OnSelectedIndexChanged="dropDownListFaculties_SelectedIndexChanged"
Visible="false"
CssClass="control">
</asp:DropDownList>

<asp:SqlDataSource ID="SqlDataSourceFaculties" runat="server"
ConnectionString="<%= $ ConnectionStrings:ConnectionStringTeacherJournal
%>"

SelectCommand="
SELECT 0 as id_faculty, N'(не обрано)' as facultyName

UNION

SELECT [Faculties].[id_faculty], [Faculties].[facultyName]
FROM [Plan]
LEFT JOIN [Groups] on [Groups].[id_group] = [Plan].[id_group]
LEFT JOIN [Faculties] on [Faculties].[id_faculty] =
[Groups].[id_faculty]
WHERE ([id_teacher] = @teacher and [Plan].[id_discipline] =
@discipline)">
<SelectParameters>
<asp:SessionParameter Name="teacher" SessionField="teacher"
Type="Int32" />
<asp:ControlParameter ControlID="dropDownListDisciplines"
PropertyName="SelectedValue" Name="discipline" Type="Int32" />
</SelectParameters>
</asp:SqlDataSource>

<br />
<br />

```

					KHTEY 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		42

```

<asp:Label ID="lbGroupsList" Text="Список групп:" runat="server"
Visible="false" CssClass="label" />

<br />

<asp:DropDownList ID="dropDownListGroups" runat="server"
DataSourceID="SqlDataSourceGroups"
DataTextField="groupName"
DataValueField="id_group"
AutoPostBack="true"

OnSelectedIndexChanged="dropDownListGroups_SelectedIndexChanged"
Visible="false"
CssClass="control">
</asp:DropDownList>

<asp:SqlDataSource ID="SqlDataSourceGroups" runat="server"
ConnectionString="<%= $ ConnectionStrings:ConnectionStringTeacherJournal
%>"

SelectCommand="
SELECT 0 as id_group, N'(не обрано)' as groupName

UNION

SELECT [Plan].[id_group], [Groups].[groupName]
FROM [Plan]
LEFT JOIN [Groups] on [Groups].[id_group] = [Plan].[id_group]
WHERE ([id_teacher] = @teacher and [Plan].[id_discipline] =
@discipline and [Groups].[id_faculty] = @faculty)">
<SelectParameters>
<asp:SessionParameter Name="teacher" SessionField="teacher"
Type="Int32" />
<asp:ControlParameter ControlID="dropDownListDisciplines"
PropertyName="SelectedValue" Name="discipline" Type="Int32" />
<asp:ControlParameter ControlID="dropDownListFaculties"
PropertyName="SelectedValue" Name="faculty" Type="Int32" />
</SelectParameters>
</asp:SqlDataSource>

<br />
<br />

```

					KHTEY 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		43

```
<asp:Label ID="lbGroupInfo" Text="Інформація про групу"
runat="server" CssClass="label" Visible="false" />
```

```
<asp:Table ID="tableGroupInfo" runat="server" BackColor="White"
BorderColor="#CCCCCC" BorderStyle="None" BorderWidth="1px"
CellPadding="3"
```

```
Visible="False" border="1" frame="void" rules="rows">
```

```
<asp:TableRow>
```

```
<asp:TableHeaderCell BackColor="#428bca" Font-Bold="True"
ForeColor="White">
```

```
Факультет
```

```
</asp:TableHeaderCell>
```

```
<asp:TableCell>
```

```
<asp:Label ID="lbTableFaculty" runat="server"></asp:Label>
```

```
</asp:TableCell>
```

```
</asp:TableRow>
```

```
<asp:TableRow>
```

```
<asp:TableHeaderCell BackColor="#428bca" Font-Bold="True"
ForeColor="White">
```

```
Курс
```

```
</asp:TableHeaderCell>
```

```
<asp:TableCell>
```

```
<asp:Label ID="lbTableCourse" runat="server"></asp:Label>
```

```
</asp:TableCell>
```

```
</asp:TableRow>
```

```
<asp:TableRow>
```

```
<asp:TableHeaderCell BackColor="#428bca" Font-Bold="True"
ForeColor="White">
```

					KHTEY 6.050103 10-11.БР	Лист
Изм.	Лист	№ Документа	Подпись	Дата		44