

Проект на тему

“CRM-система.”

Виконують студенти групи ІВ-71:

Мазан Ян

Повх Михайло

Ідея проекту:

Розробити CRM систему для ефективного і зручного управління підприємством, яке надає транспортні чи побутові послуги.

Визначення:

CRM (Customer relationship management) - підхід для управління взаємодій компанії з поточними та потенціальними клієнтами. Використовує аналіз даних про історію відносин клієнта з компанією для збільшення обсягу продажів.

Вимоги до виконуваного проекту:

1. HR-модуль.
2. Управління списком клієнтів, заказів та платежів.
3. Список послуг підприємства.
4. Генерація звітності за визначений період (аналіз даних системи за допомогою діаграм та графіків).
5. База даних на сервері з інформацією підприємства.
6. Система логування.
7. Робота з клієнтами.
8. Обмін повідомленнями між користувачами CRM-системи.

Вимоги до елементів системи:

1. HR-модуль:
  - 1.1. Можливість додавати/видаляти користувача системи.
  - 1.2. Каталог користувачів системи з правами доступу.
  - 1.3. Про користувача має бути відома особиста інформація (ім'я, контактні дані)

- 1.4. Налаштування прав доступу користувача до кожного з компонентів системи (додавати/видаляти користувачів, доступ до CRM-системи, замовлень і т.д.)
  - 1.5. Змінювати відділ в якому працює користувач.
2. Управління списком клієнтів, заказів та платежів підприємства:
  - 2.1. Можливість додавати/видаляти клієнтів, заказів, платежів.
  - 2.2. Змінювати інформацію про клієнта(заказ, платіж) , статус заказа чи платежа.
3. Список послуг підприємства:
  - 3.1. Можливість додавати/видаляти послуги, які надає/потребує підприємство
  - 3.2. Послуга має мати в собі: назву, стандартну ціну, інформацію, час виконання і тд.
  - 3.3. Можливість змінити ціну послуги.
4. Генерація звітності за визначений період.
  - 4.1. Діаграма, яка показує кількість виконаних, не виконаних, прийнятих, не прийнятих чи відмінених замовлень.
  - 4.2. Порівняння способів оплати (кількість заказів – сума ).
  - 4.3. Відношення оплачених і не оплачених заказів (в процентах).
  - 4.4. Діаграма, яка показує кількість заказів в кожному статусі.
  - 4.5. Кількість оплачених і не оплачених платежів кожного виду замовлень.
  - 4.6. Сума (в грошових одиницях) оплачених і не оплачених платежів.
  - 4.7. Кількість користувачів системи в кожному відділі.
5. База даних на сервері з інформацією підприємства:
  - 5.1. Збереження і надання доступу до даних системи. Відповідно і захист даних у ній.
  - 5.2. Веб-сайт підприємства. (якщо такий існує).
6. Система логування.
7. Робота з клієнтами
  - 7.1. Доступ до статистичної інформації про замовлення кожної окремого товару за певний проміжок часу та її зручне подання
  - 7.2. Доступ до історії взаємодії з конкретним клієнтом, отримання інформації про товари, що ним замовлялись
  - 7.3. (Бажано) реалізувати можливість отримати інформацію щодо надання послуг відповідно до географічного положення клієнта.
  - 7.4. (Бажано) можливість змінити ціну послуги конкретно для угоди клієнта.
  - 7.5. Сповістити клієнта/постачальника щодо укладання угоди через надсилання e-mail
  - 7.6. Розсилка всім чи декільком клієнтам повідомлень (e-mail) по певному шаблону.
8. Обмін повідомленнями між користувачами CRM-системи:
  - 8.1. Інтерфейс для надсилання e-mail користувачам системи та/або внутрішніх повідомлень між ними.

Стек технологій:

1. Мови програмування: C#(.NET), TypeScript
2. Фреймворки: ASP.NET Core ( ASP.NET Core Web API – REST сервіс), Angular(v.8)
3. ORM: Entity Framework Core
4. СУБД: MS SQL (при розробці також використовується SQLite)
5. Бібліотеки: Bootstrap, Angular JWT library, Bootswatch.
6. HTML, CSS.

Чому .NET(C#):

Альтернатива: Java

Таблиця 1

Основні аспекти порівняння	.NET	Java
Синтаксис	Пропонує належну підтримку лямбда та дженериків, LINQ, методів розширення. Це значно пришвидшує розробку системи.	Немає контролю над збиранням сміття, оскільки Java не забезпечує такі функції, як delete (), free ().
Підтримка веб-служб	Вбудована	Невбудована
Продуктивність	.NET компілюється та запускається в системі, де вони розгорнуті. Ось чому C # працює краще в умовах виконання.	Java не вимагає перетворення на машинну мову, поки код не буде виконаний.
Сумісність з іншими компонентами системи	Добре сумісна з технологіями Microsoft, які використовуються в нашій системі. Це вплине на швидкість роботи з даними.	Сумісна лише з деякими елементами стеку технологій проекту.
Підтримка популярних фреймворків JavaScript	Вбудовані шаблони для двох найпопулярніших рамок JavaScript - Angular & React (плюс Aurelia).	-
Безпека	Оскільки .NET є власною платформою, Microsoft піклується про аспект безпеки. Він надає цілодобову підтримку своїм клієнтам підприємства.	Для платформ з відкритим кодом, таких як Java, безпека завжди залишається не зовсім надійною через відсутність професійної підтримки.
IDE	Платформа .NET інтегрована з Visual Studio, що дозволяє редагувати, компілювати та налаштовувати час поведінки API за допомогою стандартних макросів бібліотеки. Крім того, розробникам не	Java IDE складається з редактора коду, компілятора та налагоджувача. Eclipse, IntelliJ Idea, Oracle NetBeans та Oracle JDeveloper - основні IDE, розроблені для

	потрібно заздалегідь оцінювати IDE та інші інструменти.	спрощення написання та тестування коду.
--	---	---

Ruby, PHP, Python зазвичай є також популярним вибором для розробки CRM систем, адже розробка на цих мовах швидша і легша(в деяких аспектах). Але вагомою причиною нашого вибору мови програмування для Web API є те, що наша команда розробників володіє лише мовами програмування C# і Java, а для вивчення інших технологій знадобиться багато часу і матеріальних ресурсів.

З Таблиці 1 ми робимо висновок, що для нашого проекту C# є більш придатною.

Чому Angular:

Альтернативи: React, Vue.js

Таблиця 2

Основні аспекти порівняння	Angular	React	Vue.js
Продуктивність	Angular найкраще підходить для односторінкових програм (SPA), де вміст оновлюється періодично	React може бути обраний для програм будь-якого розміру, який потребує регулярного оновлення вмісту	Цей фреймворк легко інтегрується з іншими бібліотеками або існуючими програмами
Документація	+	+	+
Архітектура компонентів	Фреймворк	Бібліотека	Фреймворк
Рендерінг	На стороні клієнта	На стороні сервера	На стороні сервера
Повторне використання коду	Так	Тільки CSS	Так
Повний веб додаток	Може бути використаний як автономна база	Потребує інтеграції багатьох інших інструментів	Потребує другорядні інструменти
Сумісність з браузерами	Сумісний з будь-яким браузерному середовищі	Не сумісний з будь-яким браузером	Не сумісний з будь-яким браузером
Прив'язка даних і направленість	Двостороння або двонаправлена прив'язка даних гарантує, що зміна даних у моделі відобразатиметься у представленні негайно, а коли дані у поданні змінюються, модель також оновлюється.	Одностороння передача даних	Одностороння передача даних

Модульна архітектура Angular більш зручна при розробці великих додатків, таких як наша система . Модуль створюється для вирішення декількох завдань з схожою функціональністю. Отже, кінцевий розмір коду стає менше, а швидкість обробки вище.

Чому MS SQL:

Після аналізу необхідних характеристик СУБД, виокремимо найголовніші з них:

- швидкий доступ до даних;
- гарантія відпрацювання запису до бази даних;
- створення простих, мало залежних один від одної сутностей;
- стійкість бази даних до невеликих навантажень, без втрати даних;
- її робота на віддаленому сервері, без застосування великої кількості ресурсів обчислювальної машини.

Всі ці характеристики задовольняє реляційна база даних, бо виконує транзакції за принципом ACID.

Зараз є кілька реляційних баз даних, які є доволі популярними і використовуються в багатьох провідних компаніях: MySQL, Oracle SQL, Microsoft SQL Server, PostgreSQL.

Для даного проекту не потрібна дуже потужна база даних, більш підійде щось легке, щоб не займало багато оперативної пам'яті тому Oracle SQL доведеться залишити для більш потужних та популярних баз даних.

Але це не означає, що нам не потрібна швидкість обробки інформації. Також потрібно зберігати дані об'ємом в десятки гігабайт. Тому MySQL нам не підійде.

PostgreSQL та SQL Server підходять під наші умови, адже і не є дуже важкими і в той же час доволі швидкі. Тоді якщо вибирати між ними, то краще обрати SQL Server, адже це продукт Microsoft, а додаток також буде розроблятися в середовищі від Microsoft.

Спрощена архітектура проекту:



## Деталогічна модель БД:



