ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – ВАРНА

ФАКУЛТЕТ ПО ИЗЧИСЛИТЕЛНА ТЕХНИКА И АВТОМАТИЗАЦИЯ

Катедра „Софтуерно Инженерство “

A blue and white logo

Description automatically generated with medium confidence

**Курсова Работа по**

**УИС**

Управленски информационни системи

**Тема:**

Информационна система, подпомагаща управлението на музикален магазин.

**Разработил:**  **Проверил:**

Ивайло Пламенов Руменов  гл. ас. д-р Диян Динев

Фак. № 23651227

# Задание

Управленски информационни системи -Задание 12

Да се създаде информационна система, подпомагаща управлението на музикален магазин. БД да съхранява информация за стоки, клиенти, служители и продажби. В таблиците да се включат атрибути като номер на артикул, вид, година, наименувания, изпълнител, жанр, музикална компания, единична цена, наличност, номер на продажба, дата на продажбата, стоки, брой, име на клиент, адрес, телефон, име на служител, позиция, телефон, отдел и др. Базата от данни трябва да е нормализирана. При създаване на таблиците изберете подходящ тип данни и други свойства на полетата. Да се валидират данните. Ограничете броя и наличността да са само положителни числа, а датата на продажбата да е днешна. Наименуванията на вида, изпълнителя, жанра и музикалната компания да се избират от списъци. Изберете правилни ключови полета. Свържете таблиците с подходящи релации. В основните таблици да има въведени минимум 10 записа коректни данни. Системата да генерира справки на база информация съхранена в повече от една таблица. Справките да са минимум 5. Например: Да се генерира справка, която да показва последните 5 продажби на стоки издадени последната година. Създадената информационна система да позволява въвеждане, корекция, актуализация и търсене на данни. Направените справки да могат да се експортират в подходящ файлов формат за отпечатване.

# Увод

В курсовата работа е разгледано как се реализира информационна система подпомагаща музикален магазин. За реализацията на информационната система е разгледан как се декомпозира идеята за приложението на множество стъпки, като и реализацията на всяка стъпка. Започвайки от анализиране на нужната бизнес логика и тя как повлиява на използваните архитектури и програмни технологии за имплементация. Преминавайки към дълбоко разглеждане как е разработена базата от данни и нейните градивни елементи. Проекта включва и разглеждане на реализацията на бизнес логиката като програмен код. Както и как е тестван и застрахован от провал вътрешната логика на проекта. Проекта включва и разглеждане на как е реализиран потребителския интерфейс, както и как той комуникира с бизнес логика. И финализиране с оценка на трудностите, грешките и подобренията към проекта в негова цялост.

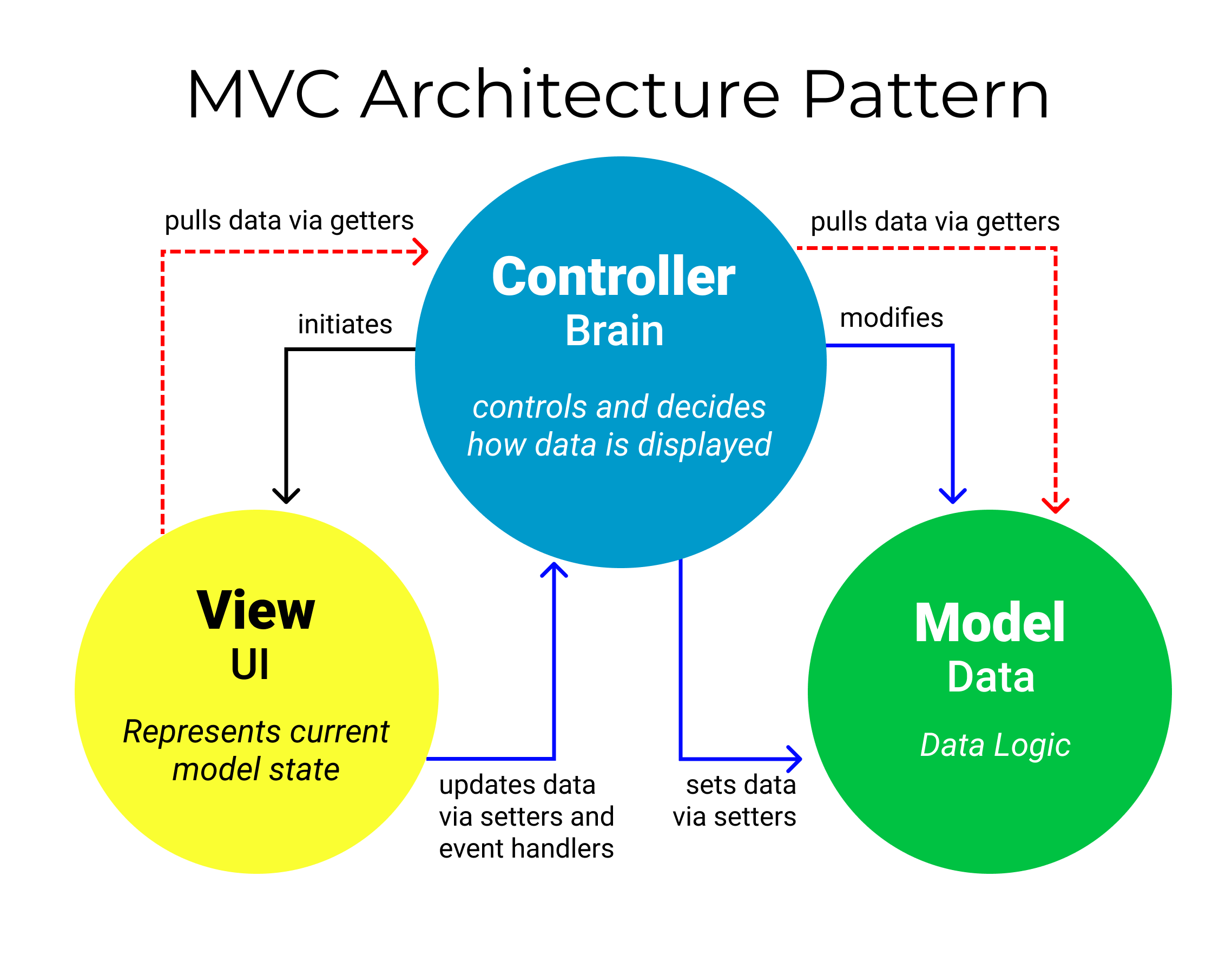
# Проектиране на спрямо бизнес процеси на заданието

При започване на задание трябва да се разчертае и цялостната структура спрямо бизнес логиката на информационната система. В този случай е предвидено информационната система да се използва като спомагателно средство на музикален магазин. Тоест цялостната цел на заданието е с комерсиална цел. Знаейки това се примамва към факта че съдържанието и менажирането на стоки и продукти ще е главната цел на информационната система. Както и при комерсиалните приложения за пазаруване се предполага че клиенти ще желаят множеството от артиколи да са сортирани и те самите при желание да сортират и търсят спрямо дадени критерий. Както при пазарува ще е нужно да се поддържа активна дадената количка на клиента. Тоест клиента да има свободата да добавя и премахва артикули в количката преди закупуване. И при закупуване на артикули информационната системата трябва да изведе дадените артикули от складова база с данни и да промени количество и да запише поръчката на клиента.

# Архитектура на информационната система

A diagram of a diagram

Description automatically generatedСлед обзора на бизнес логиката можем се стига до факта че информационната система ще бъде от вид Модел Вю Контролер (Фиг. №1). Този модел е често срещан при програмни продукти от разновиден вид. Той представлява групирани на програмния ко по неговата роля. Така се схематизира всяка единица да е отговорна за едно работа, което също подкрепя един от законите на обектното ориентирано програмиране. При Модел Вю Контролер се изготвят три елемента независими едни от друг. Модела отговорен за бизнес логиката както и връзка с базата от данни. Вю-то което е отговорно за потребителския интерфейс на информационната система. Контролера е последната част от модела и отговаря за комуникационния слой помежду двата елемента. За целта на изпълнението на задачата Контролера е реализиран чрез REST метода на работа и json стандарта за предаване на данни помежду програмните слоеве. REST e метод на реализиране на HTTP заявки помежду два така наречени крайни връзки ( “Endpoints”). Често използван е метода REST за комуникация и стандартизиран като доверчив и бърз.



Фиг. №1 Графика ма архитектурата Модел Вю Контролер

# Проектиране на базата от данни

След приемане на метод на работа се преминава към едни от базовите елементи на модела Модел Вю Контролер и това е базата от данни. За целта на разработване на информационна система за музикален магазин, както и вземайки предвид предишен опит с работата по и с бази от данни, е избрано да се използва PosgreSql база от данни. Тя използва налог на sql наречен psql. Базата е разгъната локално на машината, на която се разработва проекта. Съществуват облачно базирани услуги за поддържане на база от данни, като neon, digitaOcian, cauchebase и много други. Удобството на облачно разгъната база от данни е че множество души могат да работят едновременно по базата но и съществуват проблеми при заключване на достъп до полета и таблици. Но за целта на задачата е предприето базата от данни да е разгъната на локалната машина. Поради липсата от нуждата да се споделя данни, като при екипи от разработчици. Но при имплементация на реална система е силно препоръчано тя да е разгъната на специализиран сървър за менажиране единствено и само за база от данни. Съществуват както и специализирани операционно системи само за менажиране и поддръжка на бази от данни, техните инстанции и мрежовата комуникация нужна за работа с базата.

След като е избрана базата за работа и тя къде ще бъде разгъната може да се премине към писане на разработка на базата от данни и това може да се сведе до няколко ключови стъпки:

* Определете целта на база данни. Това ще помогне да се подготвите за останалите стъпки.
* Намиране и организиране на необходимата информация. Събиране на всички типове информация, която може да запишете в базата данни, като например име на продукт и номер на поръчка.
* Разделете информацията в таблици. Разделяне на информацията на елементи на основни обекти или теми, като например продукти или поръчки. След това всеки предмет се превръща в таблица.
* Превръщане на информационните елементи в колони. Решава се каква информация ще се съхранявате във всяка таблица. Всеки елемент става поле и се показва като колона в таблицата. Например таблицата "Служители" може да включва полета като "Фамилно име" и "Дата на наемане".
* Задаване на първични ключове. Избиране първичния ключ на всяка таблица. Първичният ключ е колона, която се използва за еднозначно идентифициране на всеки ред. Пример за това може да бъде ИД на продукт или ИД на поръчка.
* Настройване на релациите между таблиците. Поглежда се всяка таблица и решете как данните в една таблица са свързани с данните в други таблици. Добавете полета към таблици или създайте нови таблици, за да изясните релациите, ако е необходимо.
* Прилагане на правилата за нормализиране. Прилага се правилата за нормализиране на данните, за да видите дали вашите таблици са структурирани правилно. Направете корекции в таблиците, ако е необходимо.

Това е само един от вариантите за разработка на база от данни. Но е преценено че за конкретната задача ще е нужен този метод на разработка на база от данни.

Първо е нужно да се определи целта на базата от данни. За проекта тази цел е се състои от това да се съхранява информация за дадените артикули за продан, да се съдържа информация за клиента и неговата количка, да се съхранява потребителска информация за клиента както и да се съхранява информация за работещия персонал. Тези елементи от базата са ключови за нейното опериране. За визуализиране на тези изисквания е представена чрез графика(Фиг. №2)

A diagram of a flowchart

Description automatically generated

Фиг. №2 Графика представляваща чернова на базата от данни

След като е представяне черновата и е определено целта на база от данни. Се преминава към отделяне на откритата информация в таблици. След гарирането на нужните елементи и атрибути, в базата от данни, се влиза в конкретиката на дадените атрибути. Осмисля се какви полета са нужни за таблицата и как те ще са обвързани с полета от други таблици. Относно условията на проекта, които гласят че базата от данни ще се употребявал като спомагателно средство на музикален магазин, се извежда факта че трябва да се инстанции устойчива система за следене състоянието на артикулите за продан. Тази функционалност е ядрото на всеки комерсиална информационна система. За това е решено да има таблица “Item” (Фиг. № 3), която е състоене от всичко сходни полета нужни на един артиколи да е годен за продан. От тази таблица се извежда две таблици едната за музикални инструменти „Instrument“ и за грамофонни плочи „Record“. Така системата може да се разпределя класовете на продукти в различни таблици с тяхното наименование, и когато бизнеса си реши да си разшири предметната дейност то може само да дови нова таблица с релация към таблицата с артиколи „Item“. Това от друга страна е недостатък към системата. Тъй като за всяко ново разделение ще е нужна нова таблица и може да се стигне до момент където множество таблици имат релация към таблицата с артиколи. Водейки към голяма опашка при заключване и отключване на достъпа към ресурси.

A screenshot of a computer

Description automatically generated

Фиг. № 3 Таблицата за Артиколи „Item“

Но за целите на проекта е преценено е че най подходящия вариант за предметната дейност на магазина, която зависи от голяма разновидност при един вид продукти. На пример таблицата за инструменти „Instruments“ (Фиг. № 4), сама дефиниция на какво е инструмент варира от духово до струене. И в тази вариация се специализира допълнително, на пример от какъв материал е направен инструмента, от кой е настроен, дали е електронно съвместен и много други. Точно тази дълбока квалификация на инструментите е причината за разделяне на абстрактния артиколи, като нещо което се продава, и инструмента, като нещо което се използва за свирене. Така лесно се обединяват двете абстракции чрез релация и се получава инструмент за продаване. При това разделение на две абстрактни елемента лесно се добавя и премахнат един от друг. Затова е и възможно да се довят и други абстракции като таблицата за грамофонни плочи „Record“. Както и лесно може да се добавят допълнителни таблици към артикулите. Като е направено за производителя на артиколи с таблицата „Company“. Таблицата „Company“ сама по себе си представлява компания но тази компания в информационното приложение може да се използва като таблица на производител или друг вид предназначение.

A screenshot of a computer

Description automatically generated

Фиг. № 4 Таблицата за инструменти „Instrument“

Следващата ключова част е самата продажба на артикулите. След като сме запазили правилно артикулите и знаем е са устойчиви може да се помисли как те могат да се продава (Фиг. №5). За тази цел е реализирана така наречената количка за пазаруване, която запазва в себе си релация към артикулите желани от купувача и таблица събирайки информацията за тази неизпълнена покупка. И когато клиента реши да си изпълни поръчка се записва като изпълнена и на клиента се създава нова количка за пазаруване.

A screenshot of a computer

Description automatically generated

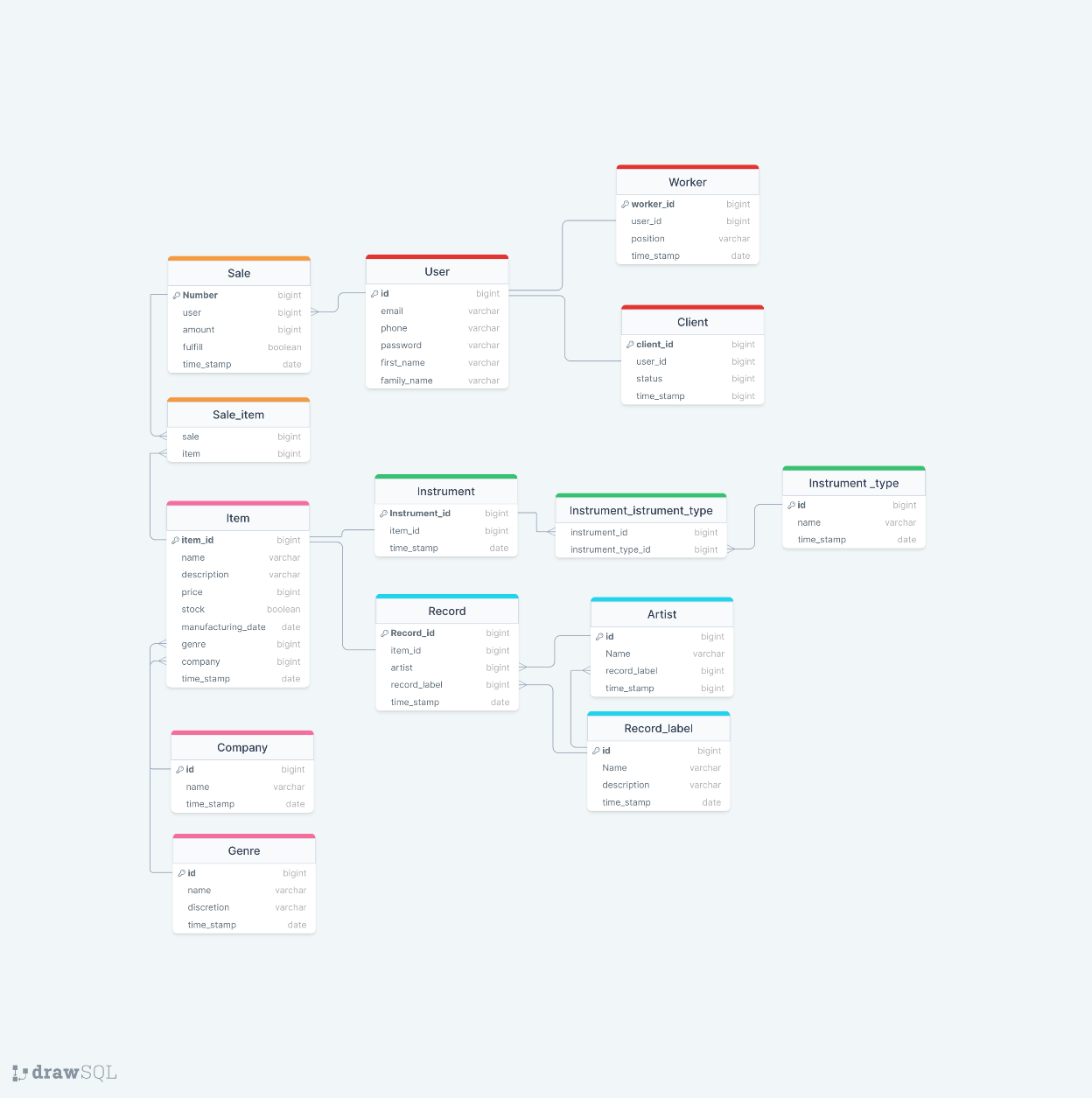
Фиг. №5 Таблиците за кошница за пазара „Sale\_item“ и таблицата за продажба „Sale“

Но за да има клиента количка той първо се нуждае от профил, в който ще се отбележи колко поръчки един клиент има и от какво са се състои дадената поръчка. Когато се имплементира модел за профил на потребите то лесно може да се добави профили за работници( Фиг. №6). Това е възможно когато се декомпозира какво е потребител и се раздели на купувач и работник. Те в същности си имат еднакви елементи.

A screenshot of a computer

Description automatically generated

Фиг. №6 Таблицата за потребител на системата „User“, таблицата за клиента на системата „Client“ и таблицата за работни в системата “Worker”



Фиг. №3 Графика на реализираните таблици в базата от данни и техните релации.

# Програмен код

След като е инициализиране базата на система може да се премини към вливането на програмния код. Тоест да се реализира бизнес процеси чрез вътрешни алгоритми на системата. За реализирането на това е избрано програмния език java, избран защото е отвратен на пазара, представлява устойчивост чрез силно типизиран обектно ориентиран модел на работа и лесен за имплементиране като модел. Но сам по себе си java е слаб затова е имплементиран чрез библиотечната средата Spring.

Когато имаме бизнес процеси за имплементиране про модела на работа се започва от горе надуло. Тоест се запазва от външните точки на

# Заключение

За бъдеще може да се помисли и за добавяне допълнителни абстракции към проекта. Като например опцията за вземане продукт на изплащане. Това лесно може да се постигне тъй като имаме таблица за артикули „Item“, към която може да се добави таблица за период на изплащане който е вързан и към клиента. Друга иновация за процеса на работа, който е лесно за разработване благодарение на модела на имплементация на базата, е отбелязването на служител който да проследява поръчката на клиента. Така всеки клиент ще знае към кой служител може да се обърне за проблеми с поръчката или проблеми с продукта.