ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - ВАРНА
Technical University of Varna

Факултет по изчислителна техника и автоматизация

Катедра „Софтуерни и интернет технологии”

КУРСОВ ПРОЕКТ

ПО

Обектно-ориентирано програмиране (2 част)

Тема: Система за организиране и следене на инвентар

Изготвен от:	Йордан Пламенов Христов, Филип Стефанов Димитров
Факултетен №:	19621802, 19621689
Специалност:	СИТ
Курс:	Трети
Група:	1б

Съдържание

Задание

X. Система за организиране и следене на инвентар

Да се разработи информационна система, предоставяща възможност за организиране на информация за инвентаризация.

Системата поддържа два вида потребители – администратор и материално отговорно лице (МОЛ).

Операции за работа с потребители:

· създаване на МОЛ-ове от администратор.

Системата поддържа операции за работа с инвентар:

· регистриране на продукт с инвентарен номер, описание, тип (ДМА, МА), степен на амортизация (за ДМА);

· въвеждане на критерий за бракуване – години, състояние или др.;

· регистриране на клиенти;

· регистриране на продукти към картоните на клиентите;

· отписване на продукт от картон;

· автоматично изчисляване за прехвърляне от ДМА към МА;

· бракуване на продукт.

Системата поддържа Справки по произволен период за:

· клиенти със регистрираните на тяхно име продукти;

· пълен списък на продуктите;

· списък по категории – ДМА,МА;

· бракувани продукти;

· статус на продукта – наличен, липсващ;

Системата поддържа Известия за:

· процес на трансформация от ДМА към МА

· бракуване на продукт

1. Въведение

Приложението представлява система за организиране на информация за инвентар. Продуктите, които се записват, се делят на два типа: дълготрайни материални активи (ДМА) и материални активи (МА). Те се делят основно на 3 категории – сгради, земи и машини. ДМА са всички материални ресурси на дадено предприятие, които се използват за стопански и други цели. Те не могат да бъдат лесно превърнати в пари. Един актив се признава и се отчита като дълготраен материален актив, когато:

- отговаря на определението за дълготраен материален актив
- стойността на актива може надеждно да се изчисли
- предприятието очаква да получи икономически изгоди, свързани с актива

Материалните активи се различават от ДМА с това, че са вече негодни за употреба. Те не покриват горните изисквания и могат да бъдат бракувани. Програмата набляга на функционалностите за добавяне на актив към картона на клиент, премахването му, следене на информацията за даден актив, бракуване и други важни операции.

2. Анализ на проблема

2.1 ФУНКЦИОНАЛНИ ИЗИСКВАНИЯ

Приложението се използва от два типа потребители – администратор и материално отговорно лице (МОЛ). След стартиране на програмата се визуализира прозорец за вход на потребителя. В първото поле се въвежда потребителско име, а във второто парола. Налично е поле за отметка за показване на паролата. При въвеждане на правилни данни потребителят получава достъп до системата и бива прехвърлен към началната страница.

Потребителският интерфейс е организиран под формата на меню и прозорец, визуализиращ съответната функция. Менюто е съставено от опция за изход на потребителя и други бутони, прехвърлящи фокуса на програмата. Зарежда се на всеки прозорец в програмата с изключение на прозореца за вход. Налично е изображение за известия и поздравителен надпис.

Началният екран извежда статистическа информация за броят продукти, потребители и клиенти, регистрирани в системата.

Администраторът е предварително регистриран в базата данни и има достъп до всички функции на програмата. Само аминистраторът има достъп до формуляра за създаване на МОЛ – ове, тоест добавяне на потребители. Администраторът попълва полета за потребителско име, парола и имейл адрес. Извършва се проверка дали

форматът на въведените данни е правилен. При заето потребителско име или парола, съдържаща невалидни символи, или липсващи данни, се извежда съобщение за съответната грешка. Визуализира се също така и таблица с всички регистрирани потребители в базата данни. Налични са функции за редакция и за изтриване на потребители от системата.

В страницата за Products се намира таблица с всички продукти и информация за тях. Представена е справка, която отговаря на няколко от изискванията. Потребителят разполага с поле за търсене по име на продукта и също така по зададен период. Извеждат се продуктите, регистрирани на дата, която е в зададения диапазон. Реализира се справка на пълният списък с продукти, справка на списък с категориите (МА и ДМА) и справка за наличност на продуктите. Възможна е редакция или изтриване на продукта след като бъде селектиран.

В същия прозорец се намира и бутон за добавяне на продукт, който прехвърля потребителя към формуляр за попълване на данни. Попълват се име, което трябва да е под 45 символа, описание, пълно описание, начална цена преди амортизация, години(възраст), количество, степен на амортизация и степен на нарастване на амортизацията. Чрез combo box, в който предварително са въведени като опции ДМА и МА, потребителят избира типа на продукта. По същият начин е изборът и за категория. Някои от полетата са задължителни, а други, като дата на регистрация и трансформация, не. Дата на регистрация се попълва с времето, в който продуктът е въведен, ако не е предварително зададена. Всяко поле има свое собствено изискване спрямо въведените символи и дължината. При удовлетворяване на изискванията продуктът се регистрира успешно, в противен случай се показва грешка, известяваща кое не е попълнено според правилата.

След регистрацията на актива автоматично се извършва и неговата амортизация като се използва формулата: price = price – (price*rate/100), rate = rate + growth , като итерацията се извършва толкова на брой пъти, колкото са годините на продукта. Новата цена на актива се съхранява. Степените се измерват в проценти, а цената е десетично число.

На прозорецът за критерия са изобразени две таблици. Едната таблица обхваща всички регистрирани критерии, а другата всички налични дълготрайни материални активи. Преди добавяне на критерията към продукта изскача запитване дали потребителят е сигурен в решението си. Ако критерията удовлетворява стойностите на актива за години, степен на амортизация и новата цена спрямо старата, то той се трансформира в МА. Ако не, остава като ДМА и може да му бъде зададена нова критерия, която да замести старата.

Налична е опция за добавяне на критерия, редактирането и изтриване й. За въвеждане в текстовите полета са позволени само числа. Критерията price drop представлява процент от спада на цената на актива след неговата амортизация.

Бутонът Client прехвърля потребителя към формуляр за регистриране на клиенти. Прозорецът отново разполага с функции за редакция и изтриване на избрания клиент. Потребителят въвежда име, фамилия, телефонен номер, адрес, ЕГН и град. Ако градът не съществува в базата данни, той автоматично се добавя. Имената трябва да са на латиница и да не съдържат цифри или други символи. Телефонният номер трябва да е в числов формат, а ЕГН-то да е точно 10 на брой числа. Налични са и други изисквания към останалите полета, като брой въведени символи. В таблица се изведени наличните за системата клиенти.

В добавяне на продукт към клиент се реализира регистрирането на актив към картона на клиент. Видими са таблица с клиентите и таблица с налични продукти. При селектиране на даден клиент се запълва третата таблица, която се явява като картон на клиента под формата на списък от продуктите му. Могат да се добавят и премахват продукти, като също така може да се избере и количество за самото прехвъляне. По подразбиране е един продукт. Когато даден продукт бива перхвърлен към картона, от неговото количество се изважда броят на постъпилите в картона екземпляри на актива. Наличният брой за дадения актив намалява или продуктът става липсващ, в зависимост от прехвърленото количество. При премахване на количество от даден актив се извършва обратната операция. В картона на клиента се съхранява датата на премахване и записът остава само като справка, която може да се достъпи при натискане на бутона Query.

На разположение е таблица, извеждаща клиентите, и таблица с регистрираните на тяхно име продукти. И двете справки имат възможност за търсене по дата и по име. Горната визуализира регистрираните клиенти в зададения период от време, а долната регистрираните продукти към съответния клиент отново за зададен период от време (клиентският картон). Там се визуализират всички продукти, които клиентът някога е притежавал.

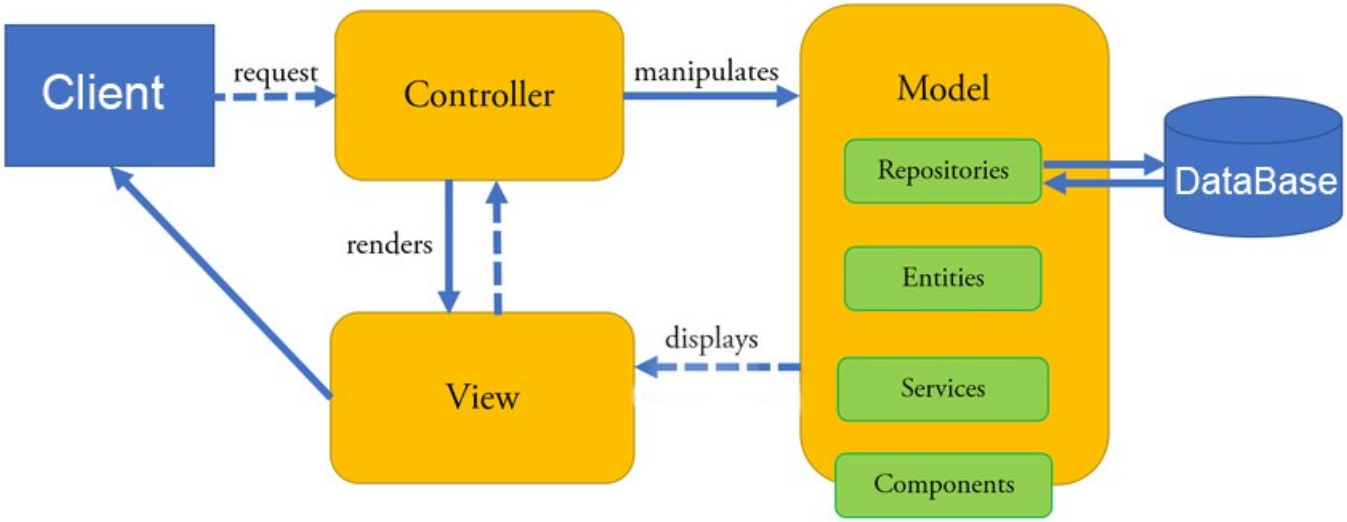
Следващата операция е за категории на продукта. Там по същият начин са реализирани добавяне, редактиране и изтриване на категория.

В Scrap се реализира бракуването на продукти. Под ръка са всички МА, включително и тези, които принадлежат на клиенти. При избор за унищожаване на продукта изскача съобщение за потвърждение. Когато се бракува МА на клиент, той изчезва от картона. Когато се бракува МА, непринадлежащ на никого, се нулира количеството и продуктът преминава в статус липсващ.

На разположение е и другата справка, която се изисква. Системата поддържа извеждане на бракуваните продукти по произволен период или по име.

Всеки потребител разполага с достъп до известия на системата. Съществуват два типа известия – за трансформация от ДМА към МА и за бракуване на продукт. След всяко задействане на тези операции се генерира известие, където е описан активът е и операцията, извършена върху него и датата. При бракуване също се изписва потребителят, извършил бракуването. Известията се изпращат автоматично към всеки потребител и могат да бъде изтрити, както едно по едно, така и изцяло. Различните потребители нямат достъп до известията на други потребители в системата.

2.2 СТРУКТУРА НА ПРОЕКТА И ДЕФИНИЦИЯ НА МОДУЛИТЕ НА СИСТЕМАТА



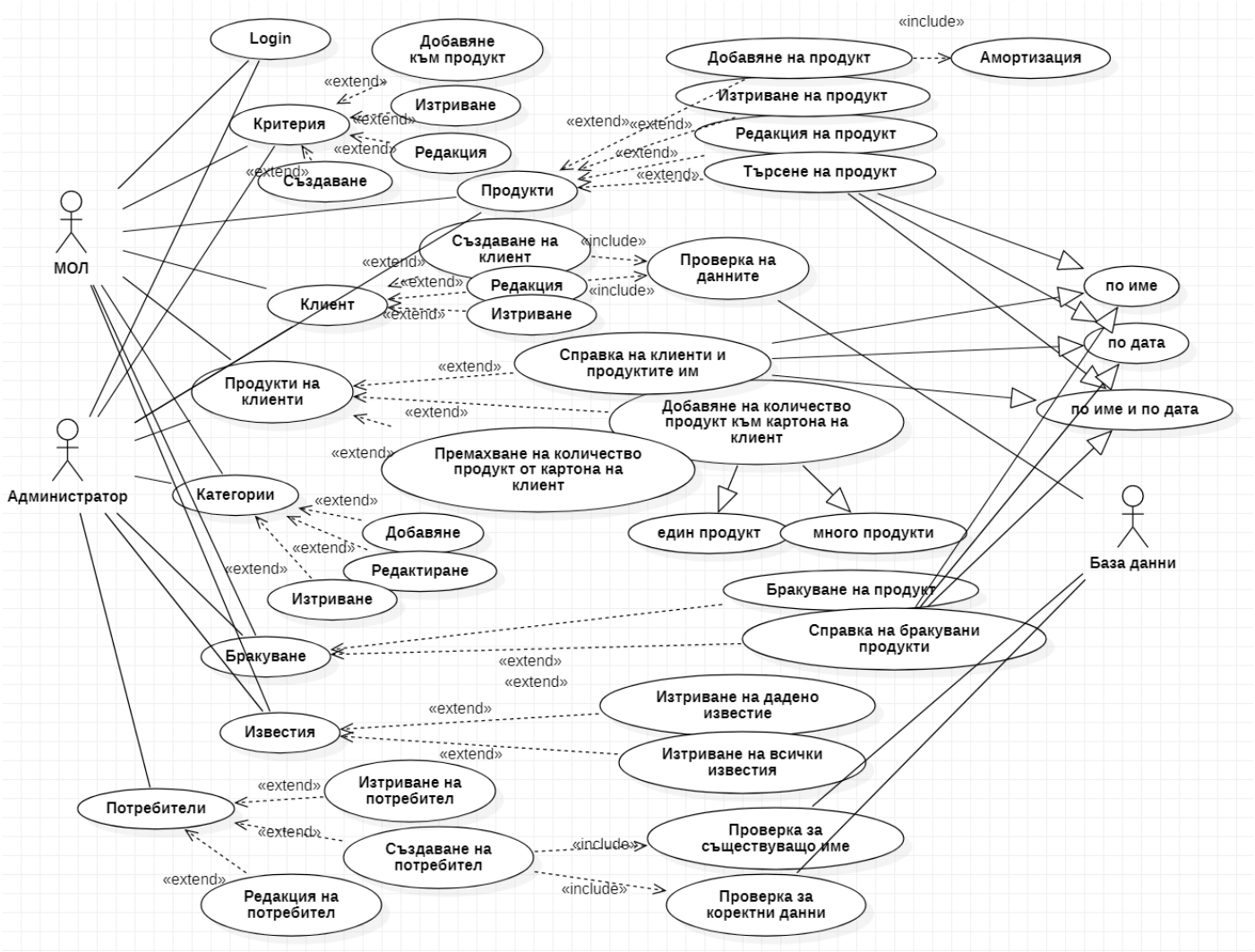
Използвана е MVC архитектура за структура на приложението.

Приложението е разделено на различни модули. В модула за Views се намират файловете, изграждащи изгледа на програмата. Те от своя страна се конторлират и управляват от Controllers. Контролерите са връзката между бизнес логиката и презентационния слой. В модулът Service е логиката, реализираща функционалностите. Там се проверява коректността на данните преди въвеждането им и се осъществява връзка с другия модул – на Repositories. В репозиторитата се осъществява комуникация с базата данни, като се извлича информация, записват,редактират или изтриват данни. Данните в приложението се съхраняват в entities. В access се създава връзката между базата данни и приложението. В модулът common са записани константи за изгледа, конфигурациите и изображенията, като се гарантира техният лесен достъп в контролерите.

Изборът на следната архитектура е поради организираността на кода и лесната комуникация между модулите.

3. Проектиране на системата

3.1 Use Case диаграма

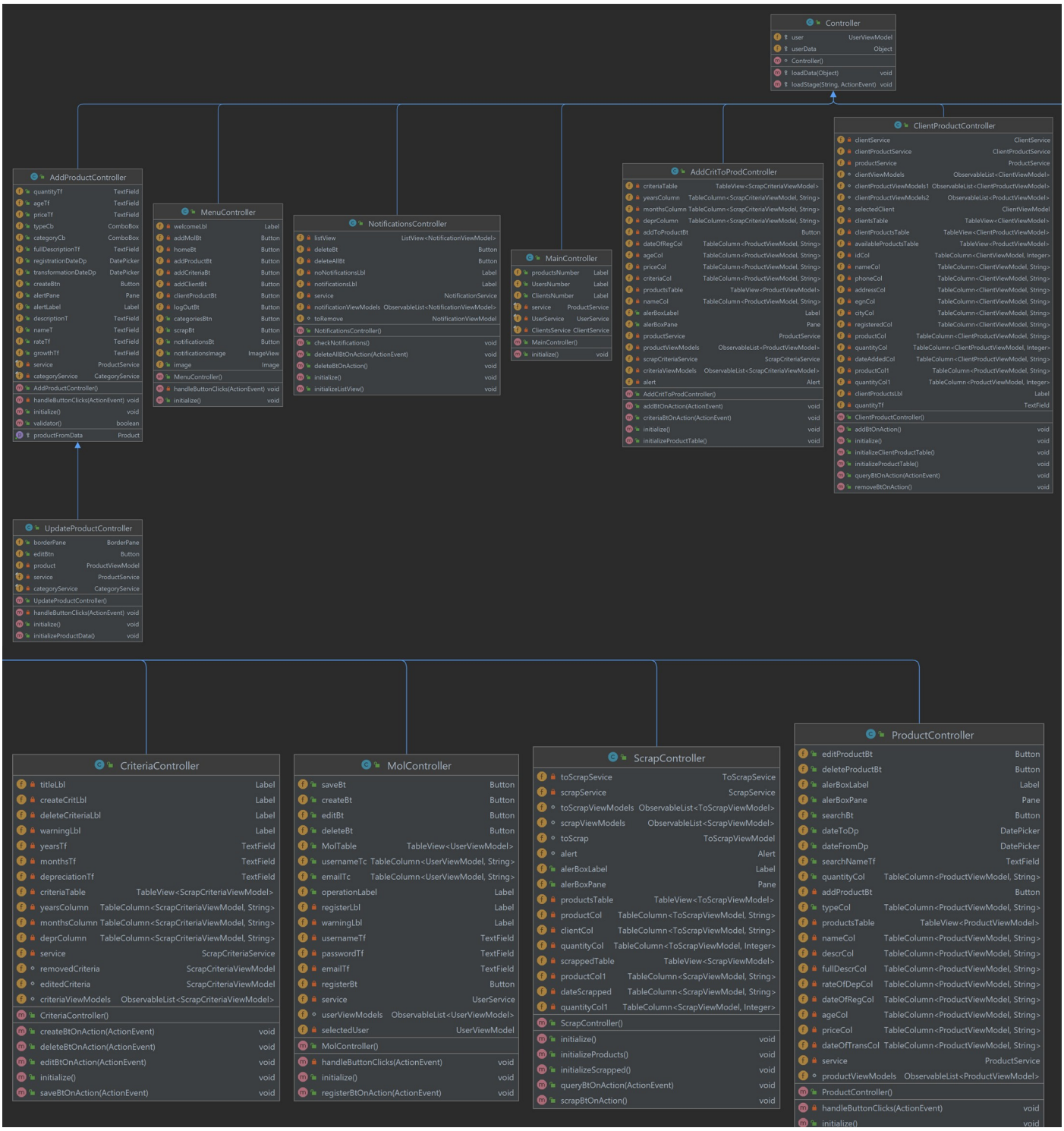


Диаграмите на случаи на употреба се използват за разясняване на работни изисквания към една система. Чрез тях се описват връзките между участниците и случаите.

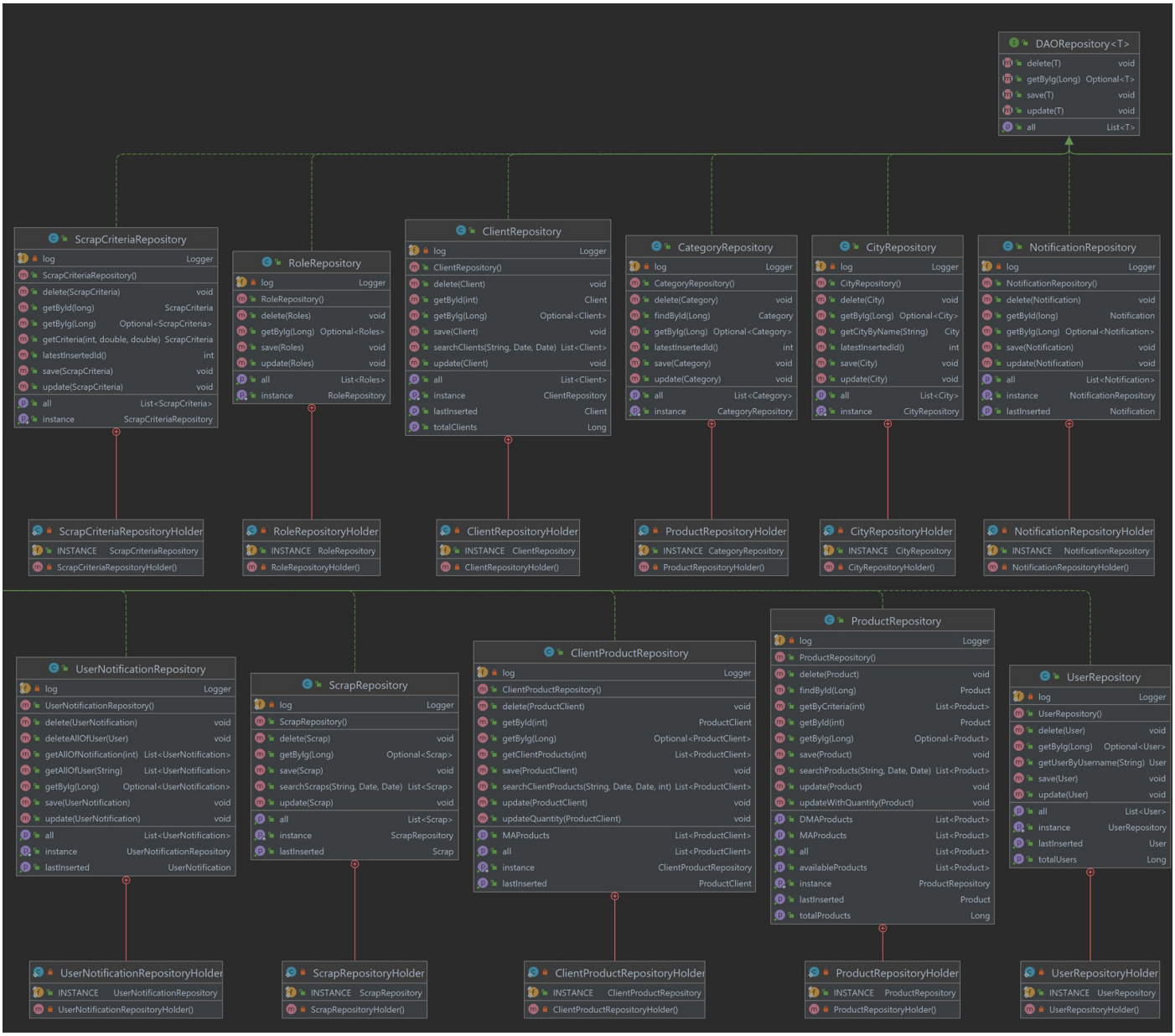
3.2 Class диаграма

Диаграмата на класове се използва за описване на статичната структура на дадена система. Показани са връзките между класовете.

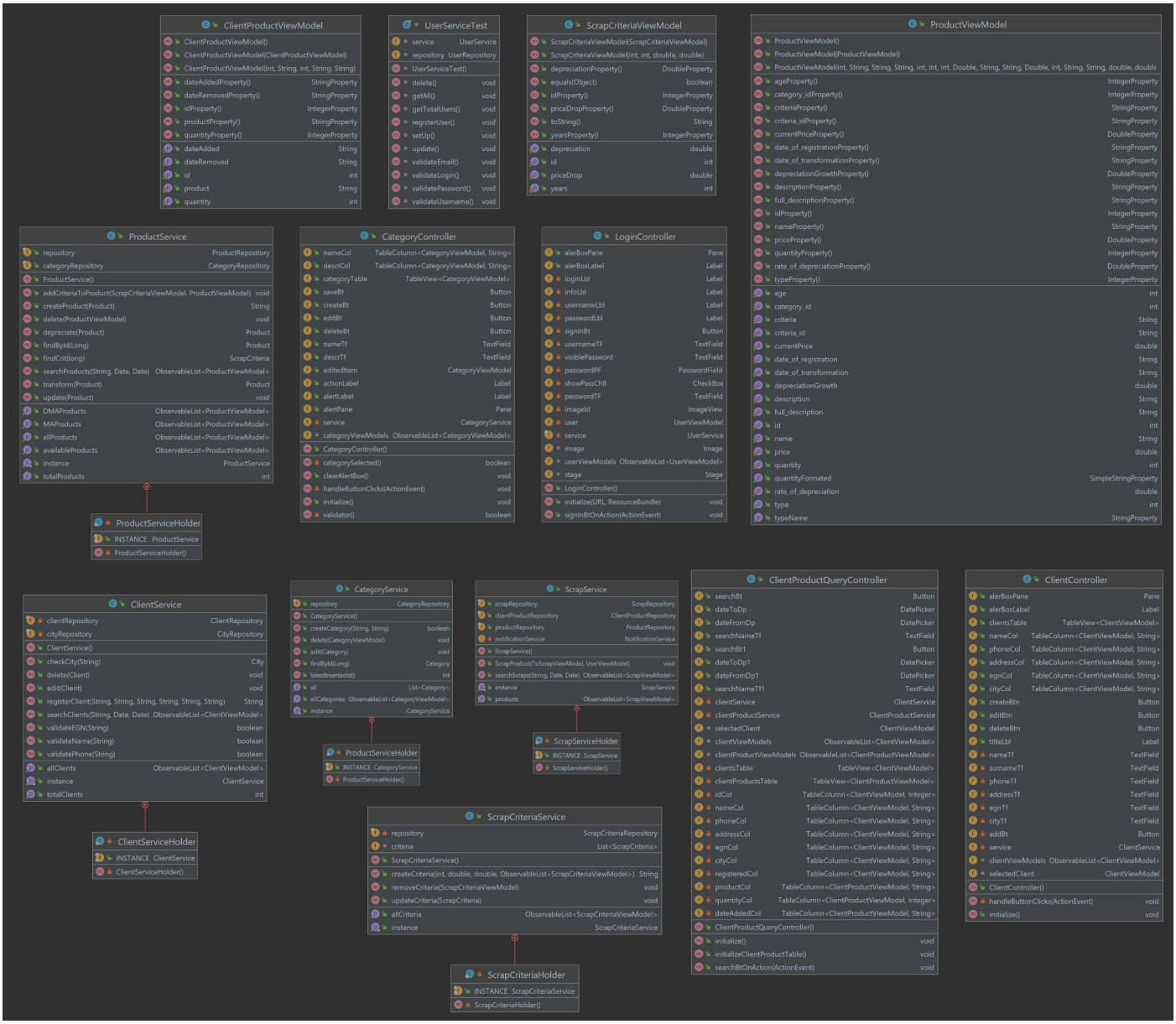
Class диаграма на контролерите:



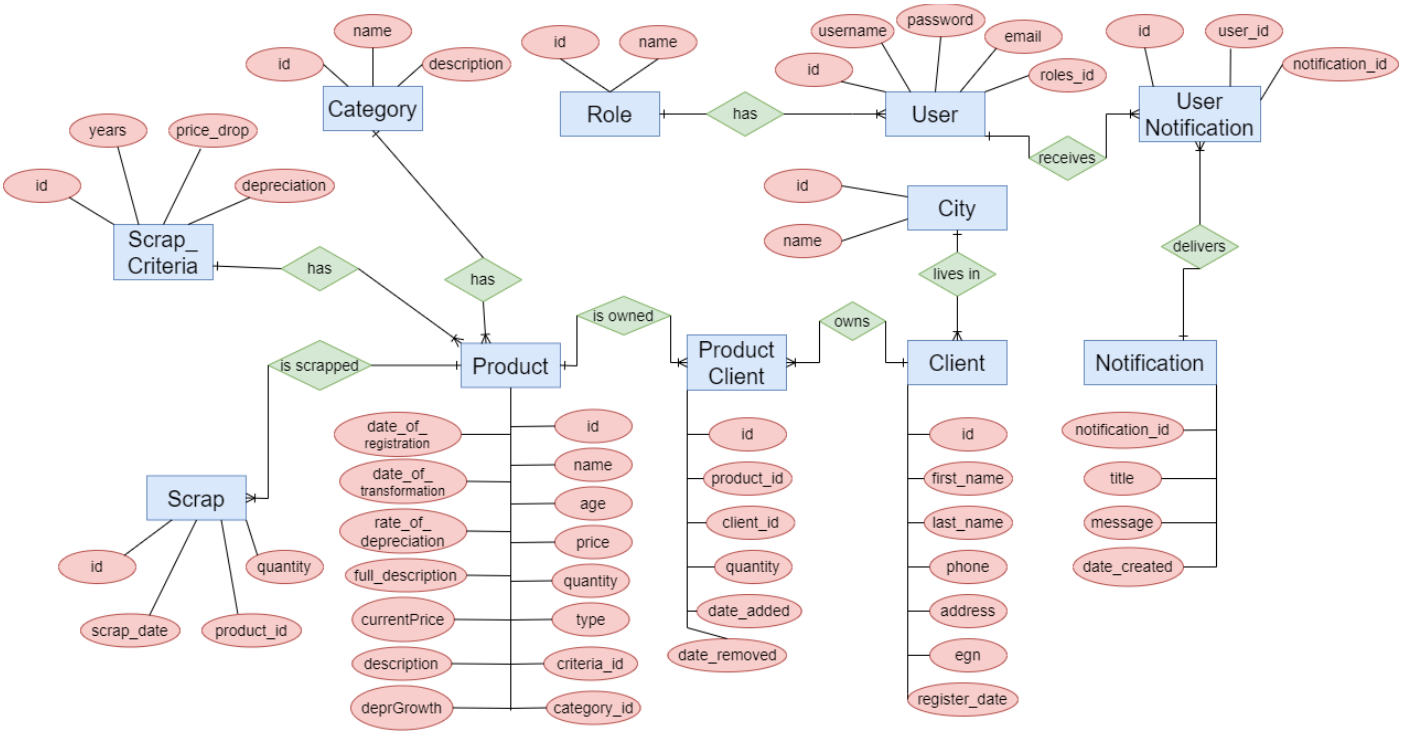
Class диаграма на Repositories:



Class диаграма на други класове:



3.3 Модел на Чен



4. Реализация на системата

4.1 Реализация на базата от данни

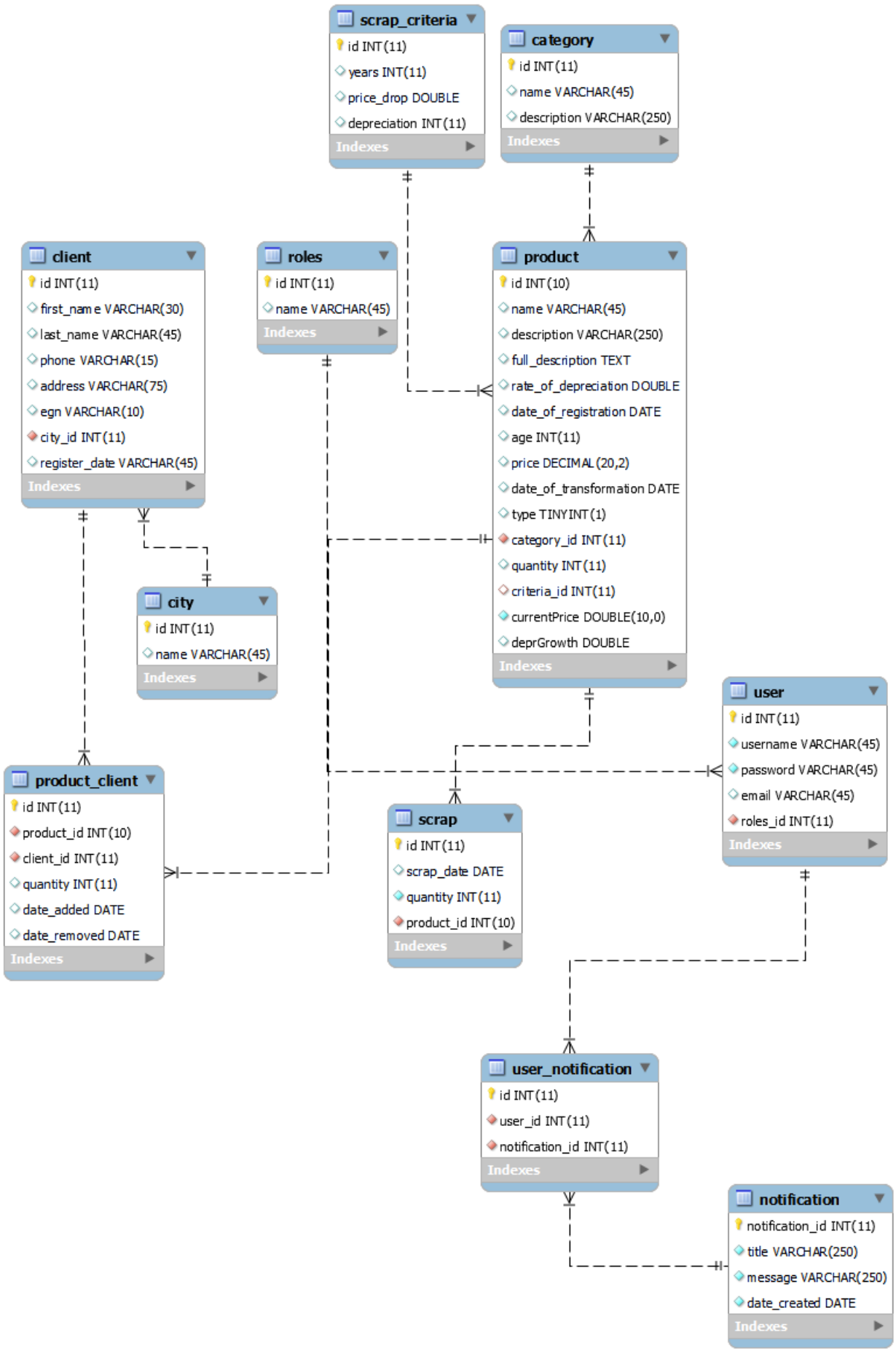


Таблица Roles

Данните в тази таблица не могат да бъдат редактирани или изтривани през приложението. Също така не е позволено добавянето на нови роли. Има два записа, които отговарят за правата на достъп на потребителя. Таблицата се състои от id и name и е свързана с таблицата User. Когато ролята на User е 1, той се води като администратор и разполага с всички функционалности на програмата. Когато ролята е 2, той е материално отговорно лице или накратко МОЛ.

Таблица User

User-ът се състои от потребителско име, парола, имейл и роля, която вече споменахме. Типът на данните е Varchar, като са позволени до 45 символа. Потребителското име трябва да бъде уникално, както е идентификационният номер. ИД на ролята е тип INT също като ID на потребителя.

Таблица Notification

Известията притежават колони за наименование, съобщение и дата на създаване. Наименованието не може да бъде повече от 45 символа, а съобщението не по-дълго от 100. Използвани са типовете Varchar и Date за датата.

Таблица User_Notification

Това е междинната таблица на User и Notification. Тя реализира връзката много към много и в нея се съхраняват ID-тата на потребителя и известието. Чрез нея се осъществява изпращането на всяко известие към всеки потребител.

Таблица City

Таблицата съхранява градове на регистрираните клиенти. Тази таблица е с цел нормализиране на данните. За име на граз се използва varchar и не са предвидени градове с дължина над 45 символа.

Таблица Client

Клиентите се характеризират с име, фамилия, телефонен номер, адрес на местоживеене, ЕГН, дата на регистрация и информация за града. Имената са с ограничена дължина. Телефонният номер е до 15 цифри, а ЕГН-то до 10. В програмата е предвидено въвеждане на точно 10 цифри за ЕГН-то на клиента. Налична е връзка едно към много с таблицата за град.

Таблица Product

Таблицата за продуктите е най-богата на полета. Един продукт се характеризира с:

- Име не по-голямо от 45 символа
- Описание, което може да достигне до 250 символа
- Пълно описание с тип TEXT, тоест могат да се използват максималният брой разрешени символи за типа
- Степен на амортизация и степен на нарастване на амортизацията Double
- Дата на регистрация и дата на трансформация Date
- Възраст на продукта в години и количество INT
- Начална цена и цена след амортизацията, които са с ограничение до 20 цифри, 2 след запетаята. Цената след амортизацията по подразбиране е 0.
- Тип на продукта TINYINT, който е или 1, или 0. 1 е за ДМА, а 0 за МА. По подразбиране е единица.
- Категория и критерия, които се извличат от другите таблици

Таблица Product_Client

Това е свързващата таблица между продукт и клиент. Играе ролята на картон и съхранява информация за количеството от даден актив, което е заето от клиента. Налични са дата на добавяне и на премахване на продукт от картона на клиент. По подразбиране датата на премахване е NULL.

Таблица Scrap_Criteria

Критерията има полета за възраст, която се записва като години INT, спад на цената и степен на амортизация, които се измерват като проценти и са реализирани с тип Double. Таблицата е свързана с продукта.

Таблица Category

Категорията се характеризира с име и описание. Налична е връзка 1 към много с таблицата на продукта.

Таблица Scrap

В тази таблица се записват бракуваните активи. В нея се взема продукта от таблицата за продукти и са налични полета за количестовото, което е бракувано, и датата.

4.2 Реализация на слоя за работа с базата данни

Приложението използва Hibernate за реализация на връзката с базата данни.

Hibernate предоставя рамка за съпоставяне на обектно-ориентиран модел към релационна база данни и заменя директния, постоянен достъп до база данни с функции за обработка на обекти от високо ниво.

4.3 Реализация на бизнеслогика и графичен интерфейс

За реализацията на графичният интерфейс се използва JavaFX.

JavaFX е софтуерна платформа за създаване на настолни приложения, както и уеб приложения, които могат да работят на голямо разнообразие от устройства. JavaFX има поддръжка за настолни компютри и уеб браузъри на Microsoft Windows, Linux и macOS, както и за мобилни устройства с iOS и Android.

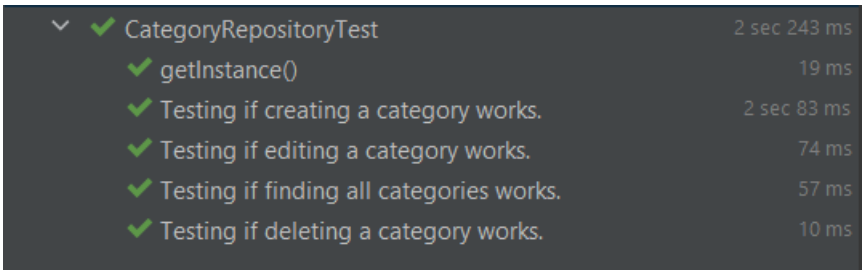
4.4 Реализация на модул за регистриране на събития в системата

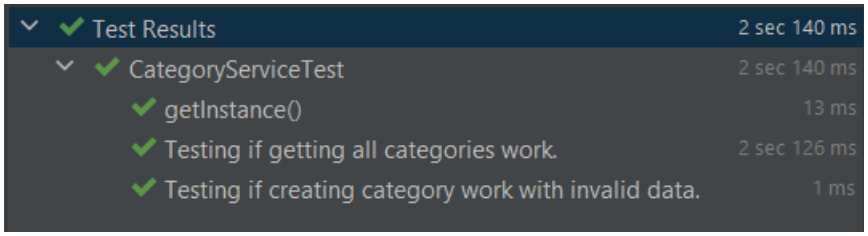
Приложението използва log4J, което е помощна програма за регистриране на събития.

5. Тестове

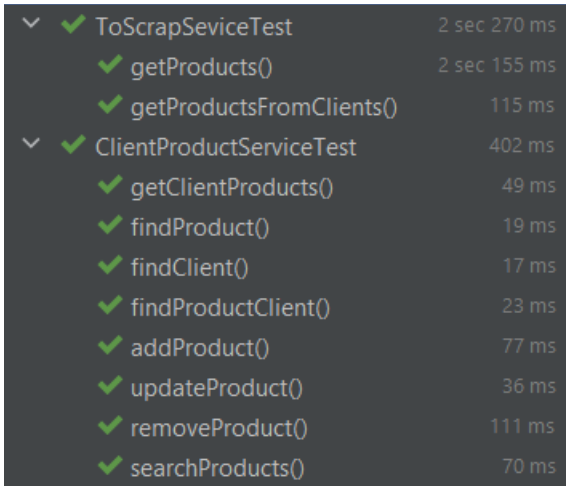
Има написани общо 64 теста за най-важните части на приложението като услугите и репозиторитата.

Тестове за функционалността на категориите

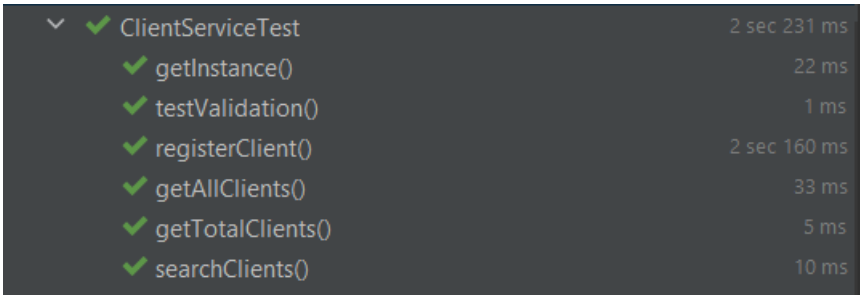




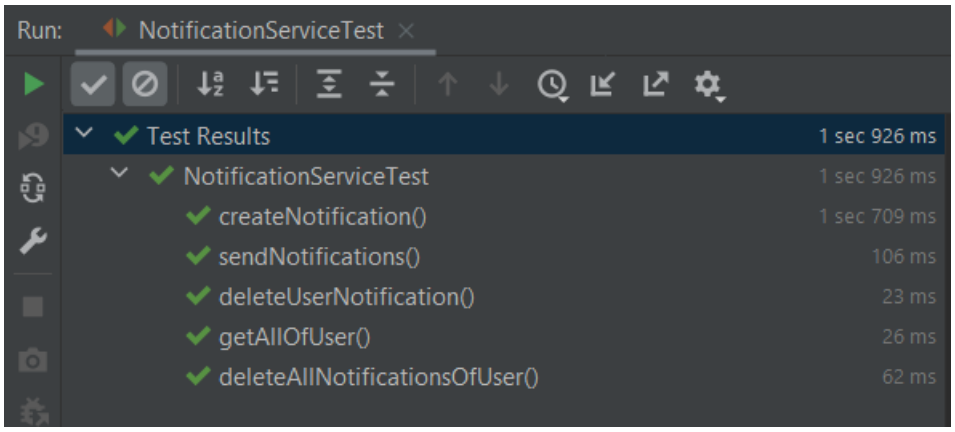
Тест на услугата за управление на картона на клиентите:



Тест на услугата за клиентите

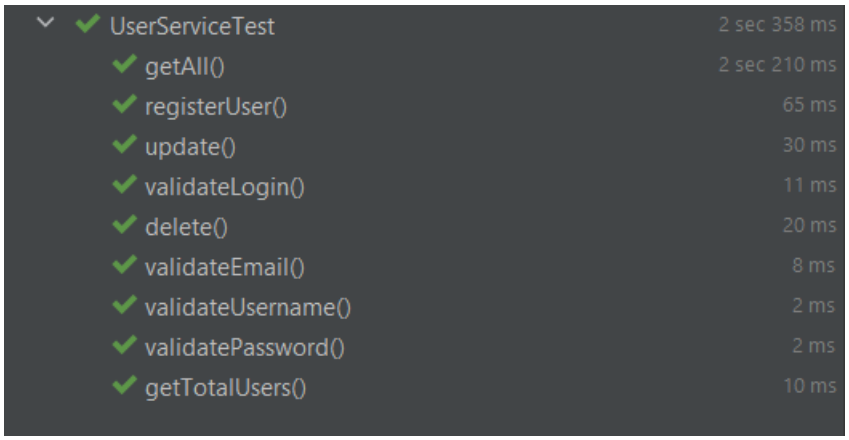


Тест на известията



Тест на услугата за бракуване на продуктите

Тест на услугата за потребители



Тест на модел

Тест на Repository за градове