Технически университет Варна

Факултет по изчислителна техника и автоматизация

Катедра „Софтуерни и интернет технологии“

Специалност „Софтуерни и интернет технологии“

**КУРСОВ ПРОЕКТ**

**по**

**Обектно ориентирано**

**програмиране – втора част**

**на тема**

**СКЛАДОВЕ**

Изготвили:

Мухаммед Февзи ФН:17621775, група 5

Синан Бехчет ФН:17621802, група 5

Специалност СИТ Трети курс Проверил:

(доц. Х. Ненов)

1. **Пълно условие на заданието на проекта**

**III. Складове**

Да се разработи информационна система, предоставяща услуга склад. Програма съхранява и обработва данни за складови помещения. Системата позволява множествен достъп.

Системата поддържа два вида потребители администратор и клиенти(складов агент, собственик) с различни роли за достъп до функционалностите в системата.

Oперации за работа с потребители :

* + Създаване на собственици от администратор;
  + Създаване на складов агент от администратор;
  + Създаване, редактиране на склад(ове) с избор на един или списък складови агента от собственик за отдаване на складовете под наем
  + Поддържане на профили с характеристики на собственици и складови агенти (хонорар и др.)
  + Рейтинговане на складови агенти .

Системата поддържа операции за работа със складови помещения:

* + Добавяне на ново помещение от собственик (размери, климатични условия, вид на съхраняваната стока, категория склад и др.);
  + Отдаване на складово помещение от складов агент, създаване на формуляр за наем (информация за наемателя, период и цена);

Системата поддържа справки по произволен период за:

* + Складов агент (сключени договори за наем, данните на уредник, рейтинг);
  + Налични складове за отдаване под наем(дата, статус, адрес, вид на склада);
  + Складове (на даден собственик);

Собствениците на складове достъпват справки само за складове, на който е собственик.Собственика достъпва справки за всички притежаващи складове.Складови агенти има право на справки за налични складове и история на наемателите.

Системата поддържа Известия за събития:

* + Новопостъпила заявка за отдаване под наем (в профила на складовия агент);
  + Отдаден(и) склад(ове) (в профила на собственика);
  + Изтичащ договор (собственик, складов агент);

1. **Анализ на проблема**

Анализирахме подробно условието на заданието на проекта по отношение на три аспекта:

1. Функционални изисквания;
2. Структура на проекта;
3. Дефиниция на модулите на системата.

Във всеки един от аспектите направихме съответните изводи:

1. Функционалните изисквания, заложени в условието на заданието, са следните:  
   1. Необходимо е да се изгради информационна система, която да осигурява автоматизация на работния процес, предоставящи системата Складове.
   2. Информационната система трябва да обхваща всички звена (нива) в йерархията на всеки склад:
      1. Администратор(управленско ниво);
      2. Собственик и Агент (изпълнително ниво).

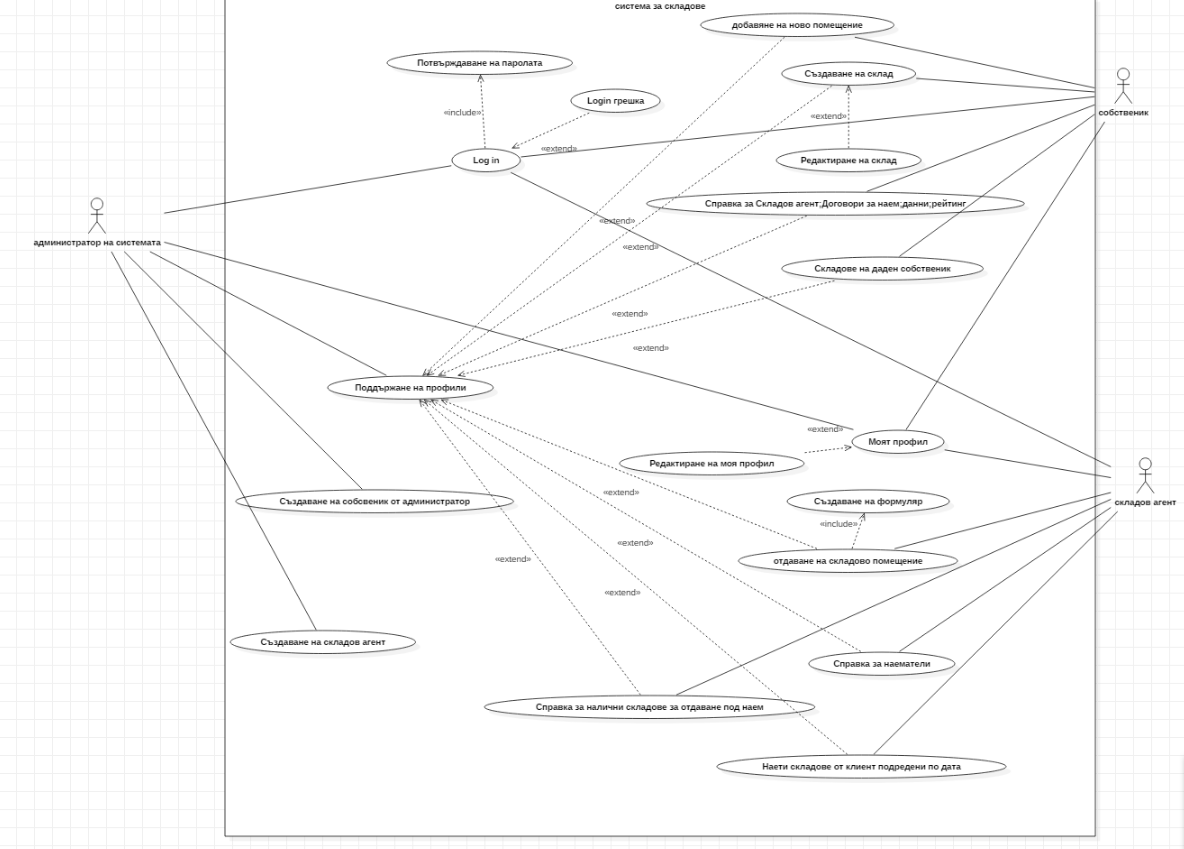
* 1. Основни потребители на системата са Админ,Собственик и Агент , тъй като те поддържат складове. Администраторите са главната позиция над собствениците и агентите, те вече имат и права за създаване и др.
  2. Потребителите на системата от различните йерархични звена, както и администраторите на системата притежават правото да се ползват от собствен профил, който автоматизира функционалностите на тяхната работна длъжност, които съответстват на тяхното йерархично ниво в компанията.
  3. Всички операции с потребители, които системата поддържа, се инициализират от администраторите или се извършват автоматично от системата, като:  
     1. Инициирани от администратор биват следните операции с потребители:  
        1. Операцията за създаване на собственици
        2. Операция за създаване на складов агент
        3. Поддържане на профили с характеристики на собственици и складови агенти (хонорар и др.)
        4. Рейтинговане на складови агенти .

* + 1. Справки, извършвани от администратор и складов агент:
       1. Всички справки биват извършвани от складов агент собственик и от администратори
       2. В зависимост от това дали да се иска да се видят всички резултати или само за период от време.

1. Относно структурата на проекта решихме да се придържаме към следните принципи:  
   1. Проектът трябва да се изгради посредством реализация на трите отделни модула или слоеве на системата, които да взаимодействат помежду си;
   2. Разработването на системата трябва да се извърши посредством последователно следване на логически свързани етапи. Добра практика за такъв вид проекти е да разделим извършената разработка на четири етапа на работа:
      1. Анализ на функционалните изисквания;
      2. Съставяне на план за разработка;
      3. Проектиране на системата;
      4. Реализация на системата.
2. Дефиниция на модулите на системата:  
   1. Презентационен слой: Модул/слой, отговарящ за графичния потребителски интерфейс (GUI), или за начина на представяне на системата пред крайния потребител, по който се осъществява взаимодействието между потребителя и системата;
   2. Слой Бизнес логика: Модул/слой, отговарящ за бизнес логиката на приложението или за същинската част на приложението, където се решава, по какъв начин ще се възприемат новопостъпилите данни в системата и как ще се направляват до базата данни, както и по какъв начин ще се извличат данни от базата данни и как ще се направляват до крайния потребител;
   3. Даннов слой: Модул/слой, отговарящ за устойчивостта (persistence) на начина, по който се внедряват и извличат данни към и от базата данни, с цел точно съответствие между записите в базата данни и обектите в бизнес логиката;
   4. Слой Бази данни: Модул/слой, отговарящ за съхранението на всички данни на системата.
3. **Проектиране на системата**

1. Общо описание на проектирането.

Проектирането на системата се извърши посредством изграждането на диаграми и модели от различни естество и структура, които имат отношение към всеки един от модулите в рамките на проекта.

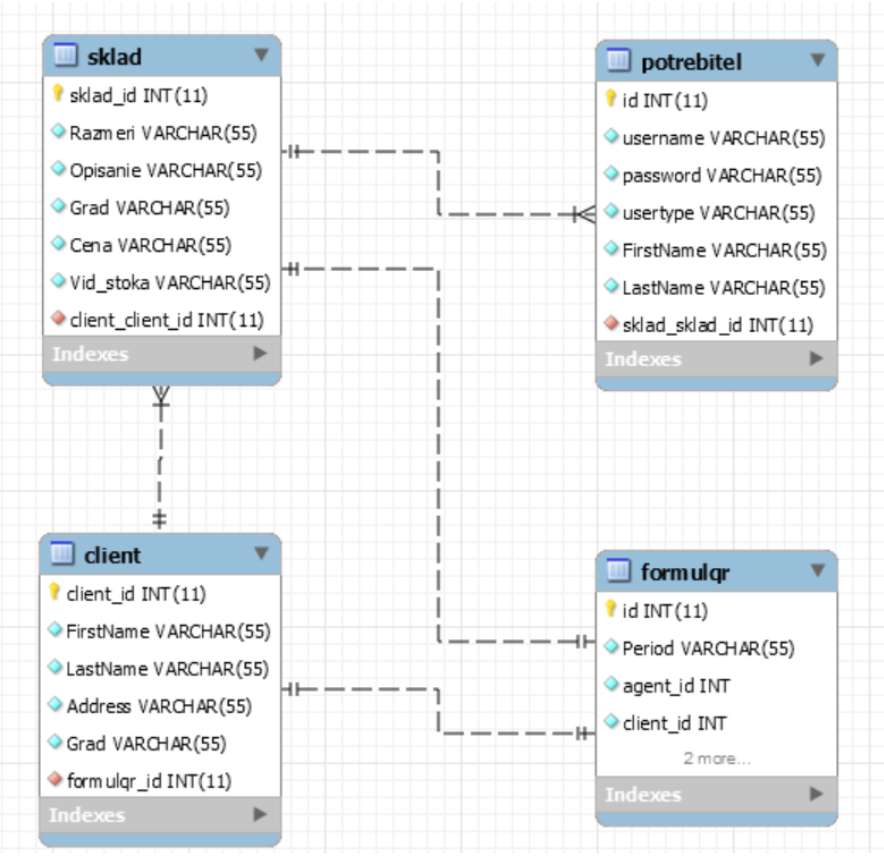


Фиг.1. Първа идея за проекта

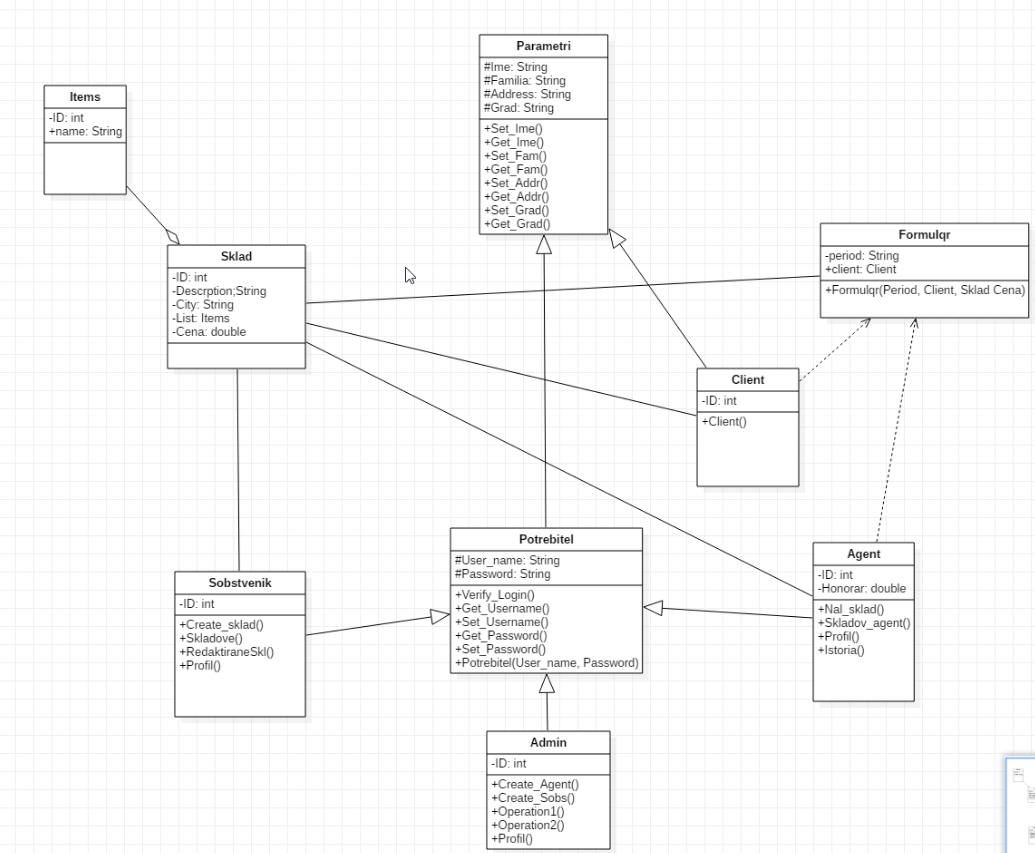
Преди да започнем с изграждането на модели от утвърдени стандарти, си създадохме обща начална представа за начина, по който искаме да изглежда завършената система. За тази цел начертахме схема на свободен принцип. На изображението горе е показана първоначалната идея за крайния вид и структура на графичния потребителски интерфейс на системата. Тази идея ни послужи за ориентир за по-нататъшната разработка на проекта. В първоначалния ни поглед за системата, описан от това изображение, подобно на блок диаграма, може да се види какви прозорци ще се визуализират при интеграция на всеки вид потребител с контролите от неговия профил. При тази структура и компанията е потребител със собствен профил, което по-късно променихме от съображения за придържане към функционалните изисквания и простота на реализацията. Големите правоъгълници в схемата представят опростен изглед на прозорците в съответния профил. При начално зареждане на приложението се показва прозорецът с форма за въвеждане на креденциали на потребител (Login форма). След успешна автентикация на потребител на екрана се извежда основния прозорец за съответния потребител според ролята му в системата.

1.1. След това в началото на проектирането на системата изградихме Диаграма на случаите на използване (Use Case Diagram).Тя се използва за поглед върху информационната система от най-високо ниво, тъй като прави възможно обстойното и изчерпателно онагледяване на всички функционали (Use Cases), които системата осигурява, и посочва каква е връзката между тях и външните за системата фактори (актьори), които обичайно инициират или пасивно констатират изпълнението на даден функционал.

1.2. По отношение на проектирането на модула с базата данни съставихме модел на Чен и Entity Relationship (ER) Diagram. По същество моделът на Чен и ЕR диаграмата онагледяват един и същи проблем, а именно кои са основните единици (същности (Entities)), които ще изграждат системата, какви са техните свойства и какви са връзките между тях. Разликата е, че моделът на Чен добавя някои специфики, като например възможност за по-ясно обособяване на връзките между същностите в отделен графичен елемент (ромб), което позволява на самите връзки между същностите да се обявят свойства, подобно тези на самите същности.

1.3. Проектирането на слоя от бизнес логика за настоящото приложение – информационна система - реализирахме чрез моделиране на Class диаграми. Те се използват за онагледяване на изолирани един от друг обобщени обекти и отношенията между тях. свойства и операции в рамките на един клас.





1. **Реализация на системата**

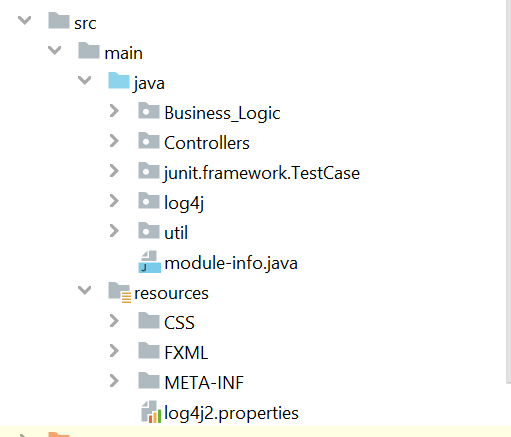
1. Реализация на слоя за работа с базата данни

Слоя за работа с базата от данни се реализира чрез Java Database Connectivity  
  
  
3. Реализация на бизнес логика и графичен интерфейс – Java, JavaFX

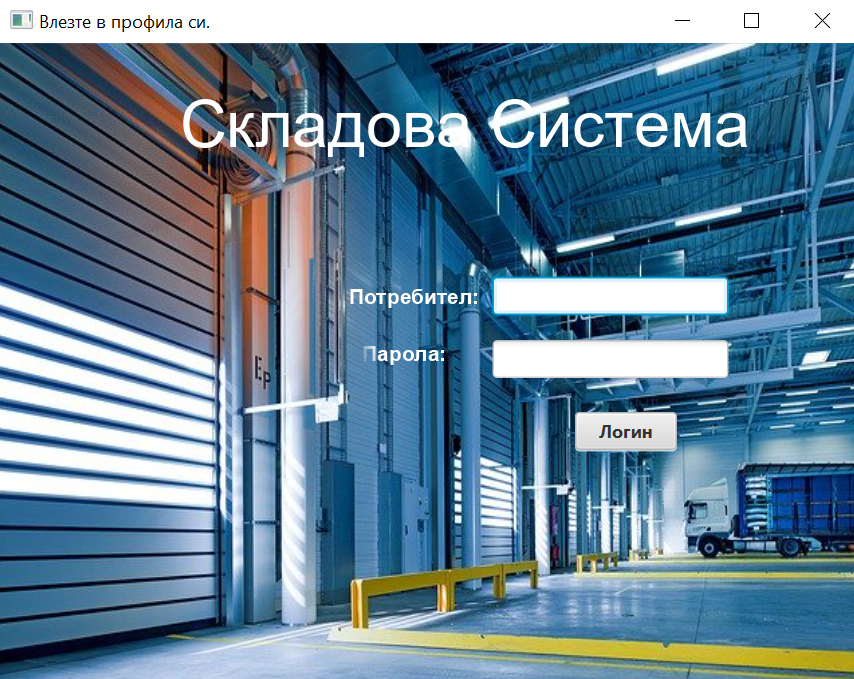
В реализацията на бизнес логиката сме използвали класове, структурирани в три пакета: Controllers, Business\_Logic и Util. Конфигурирали сме проекта си като Маven Project.

За графичния интерфейс сме използвали JavaFX библиотеките и SceneBuilder.

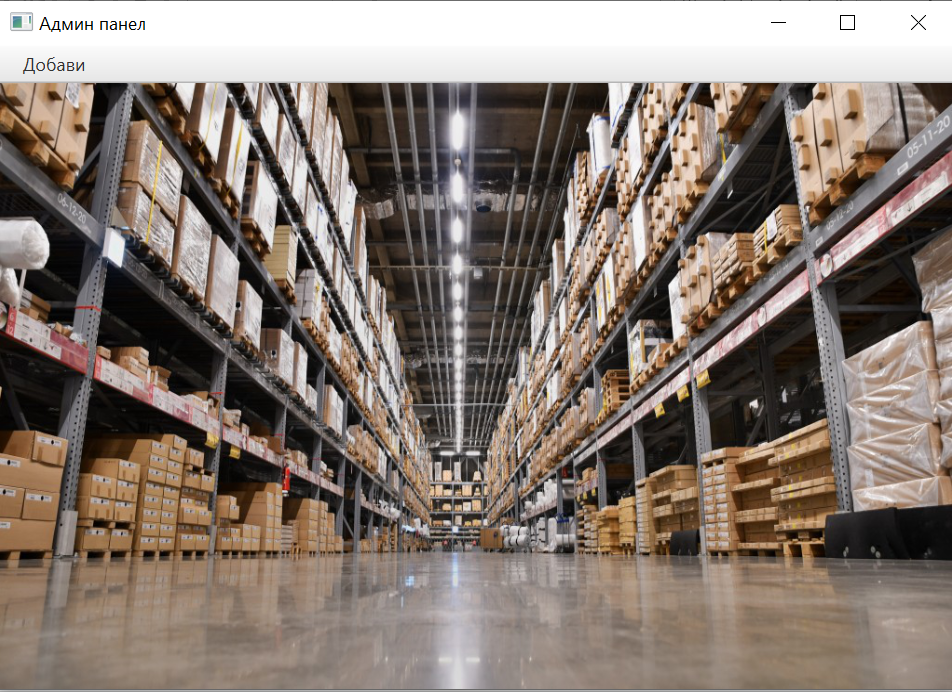
# 



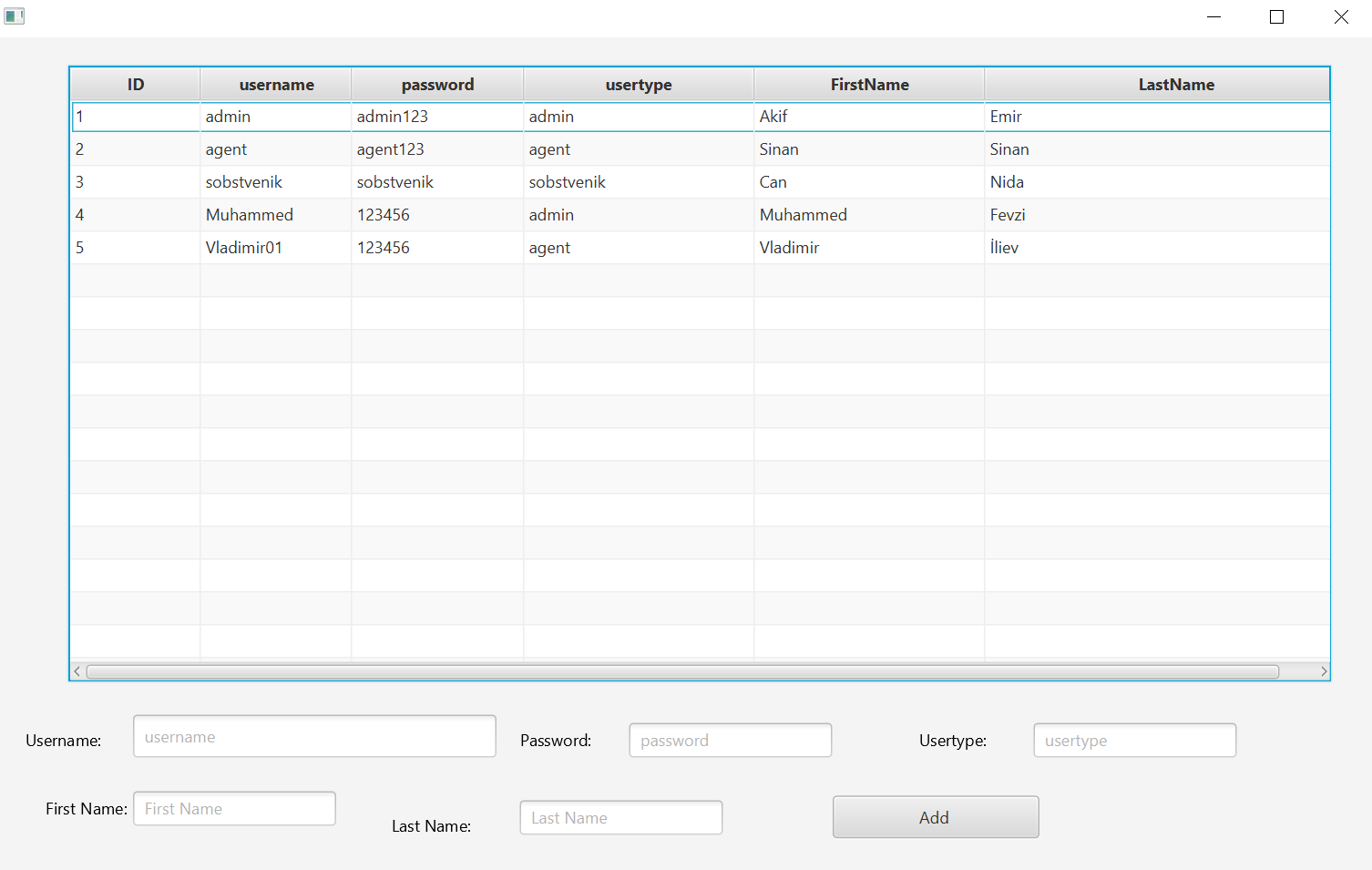
Фиг. Структурна схема на проекта



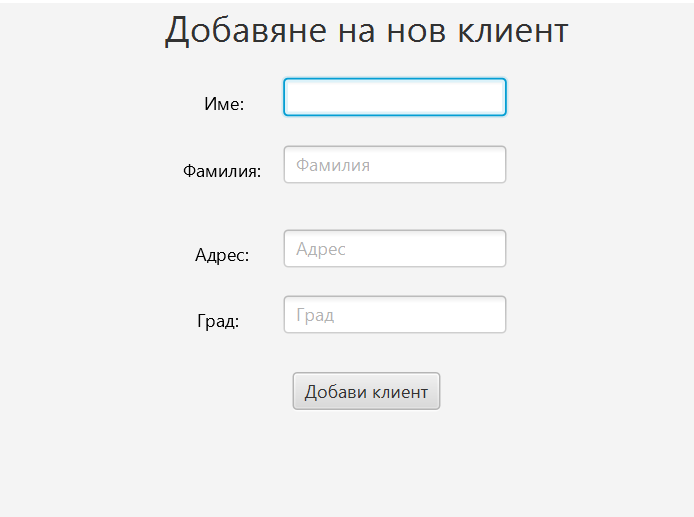
Фиг. Екран за влизане в програмата като админ,складов агент или собственик



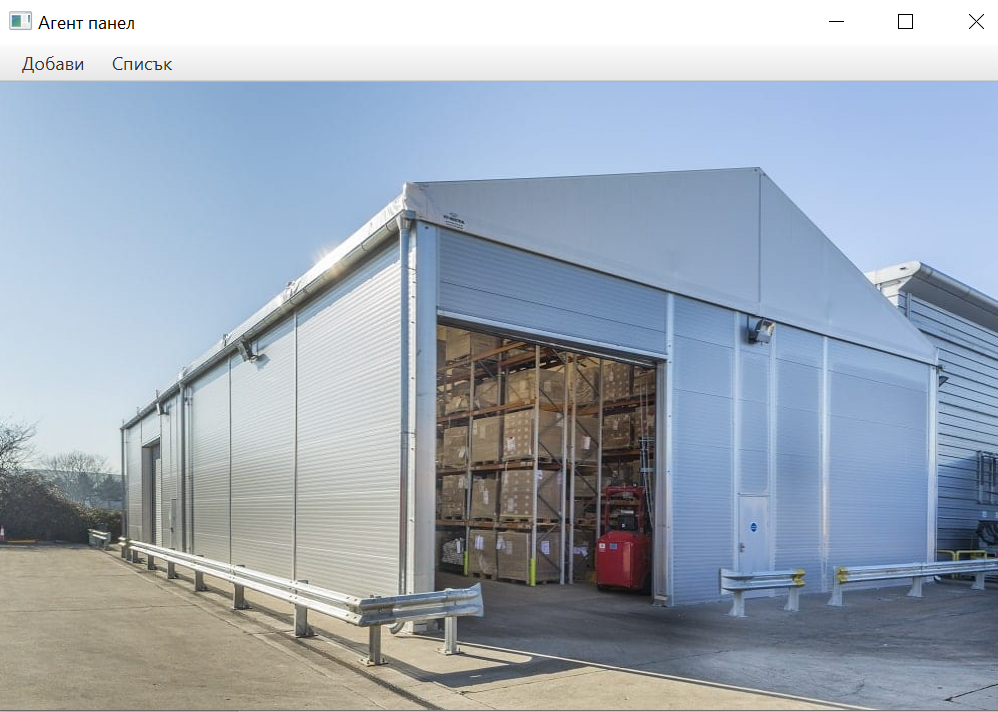
Фиг. Админ Панел



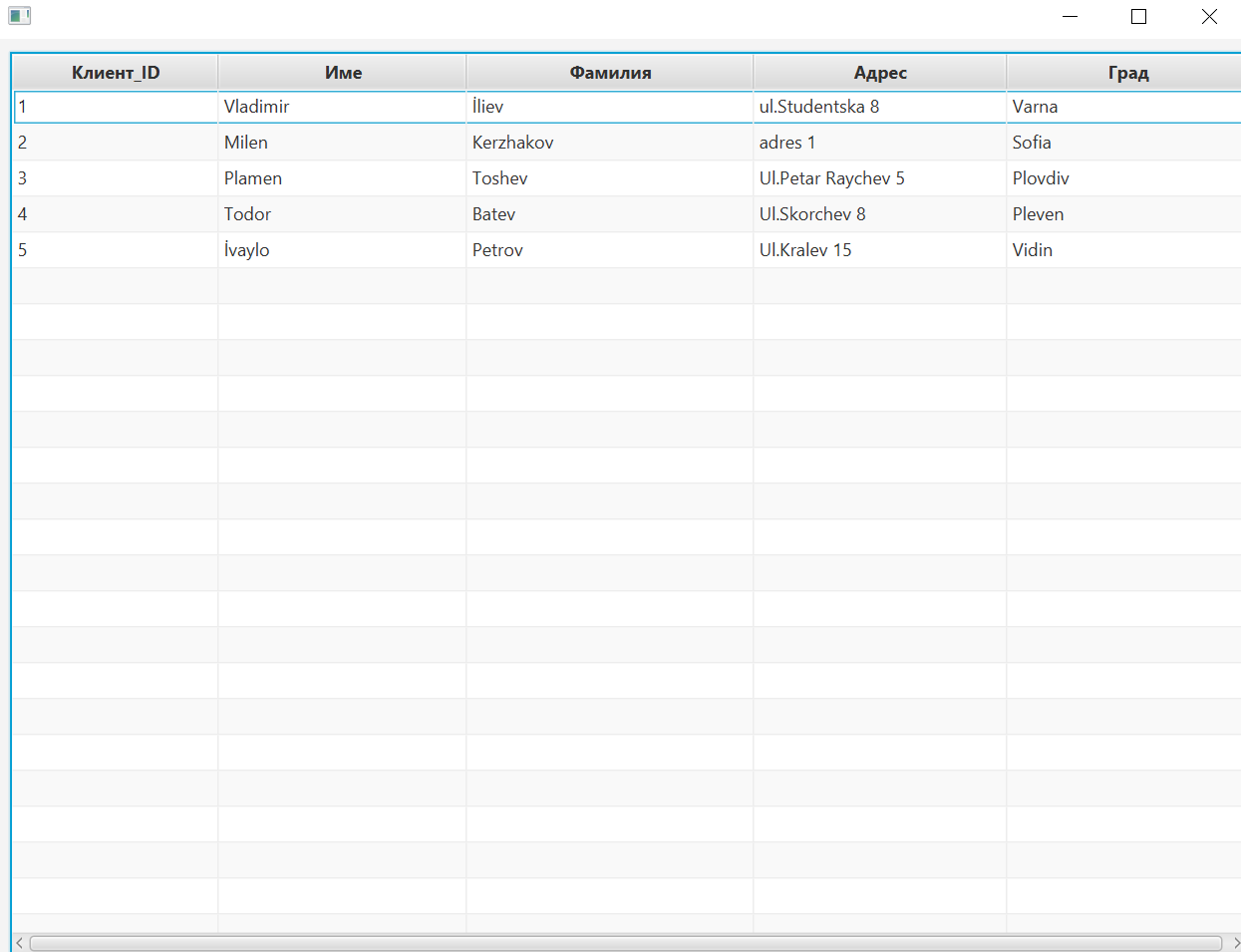
Фиг. Добавяне на потребител



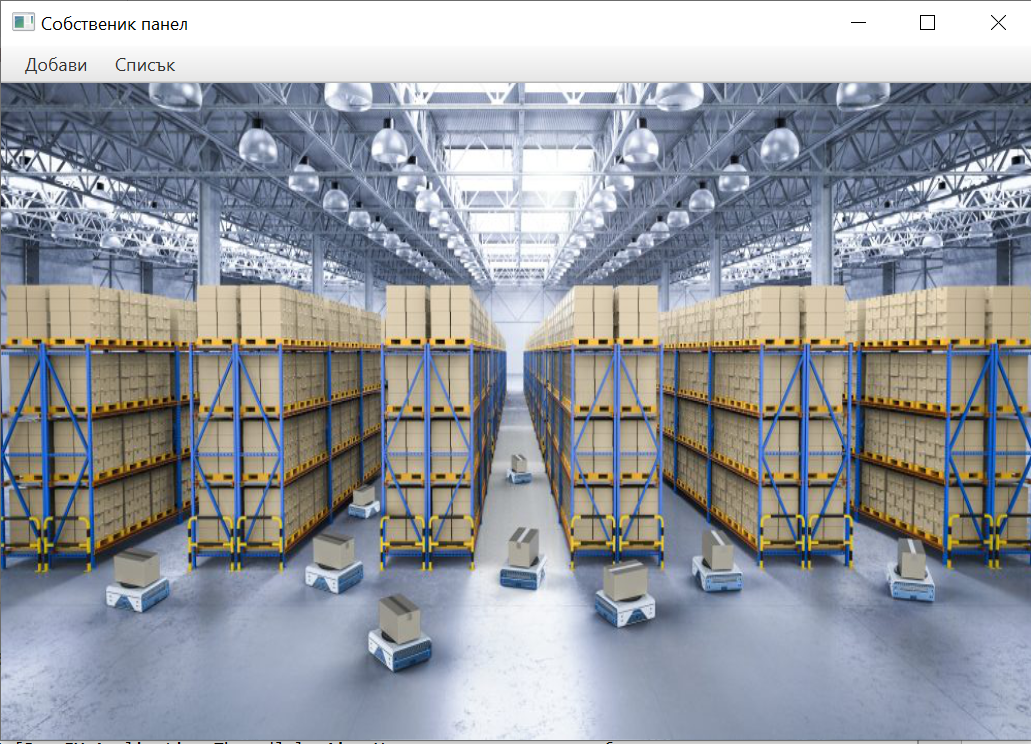
Фиг. Добавяне на клиент



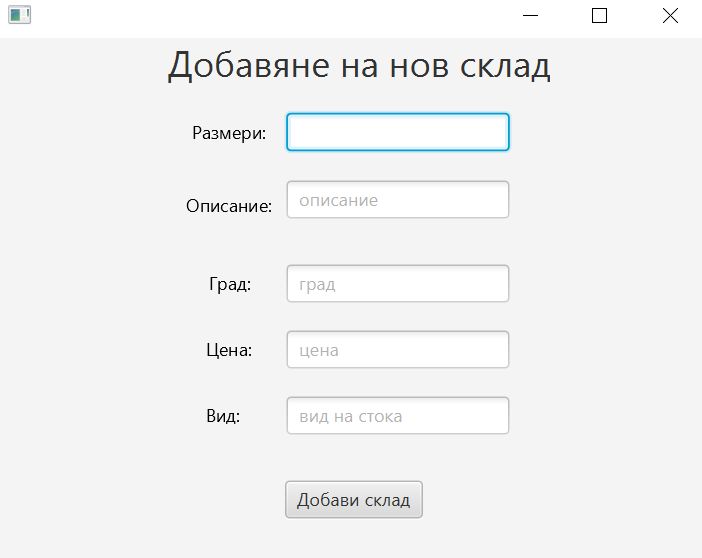
Фиг. Агент Панел



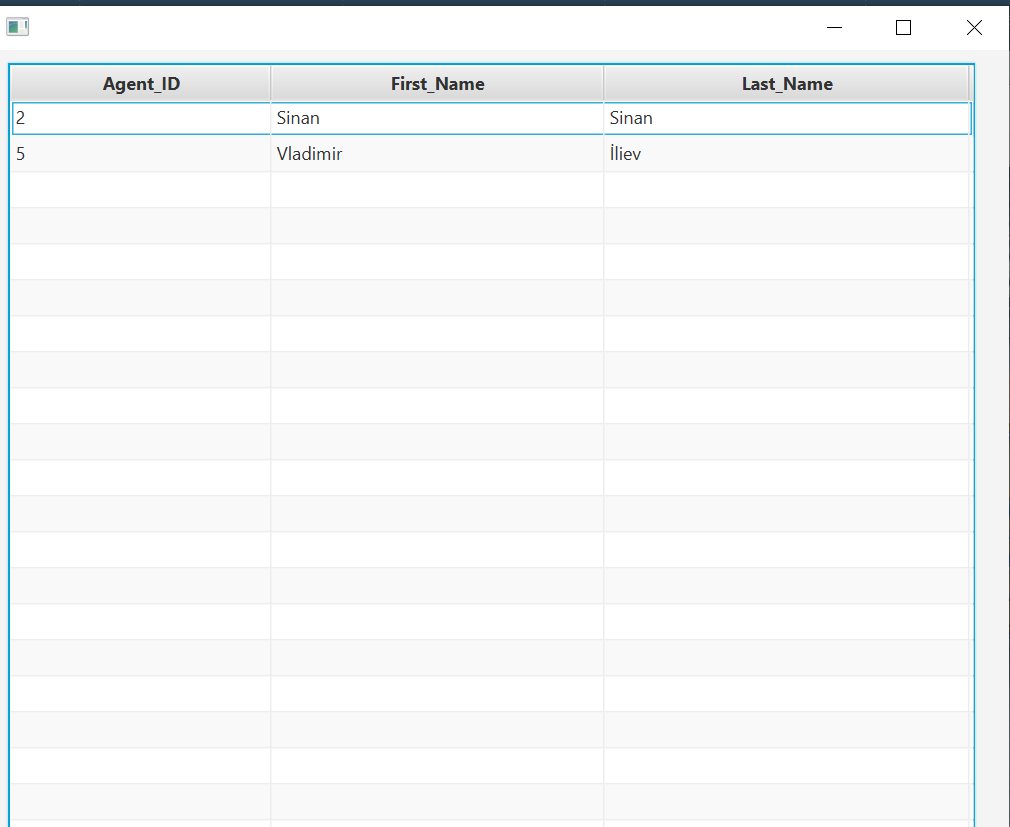
Фиг. Списък с клиенти



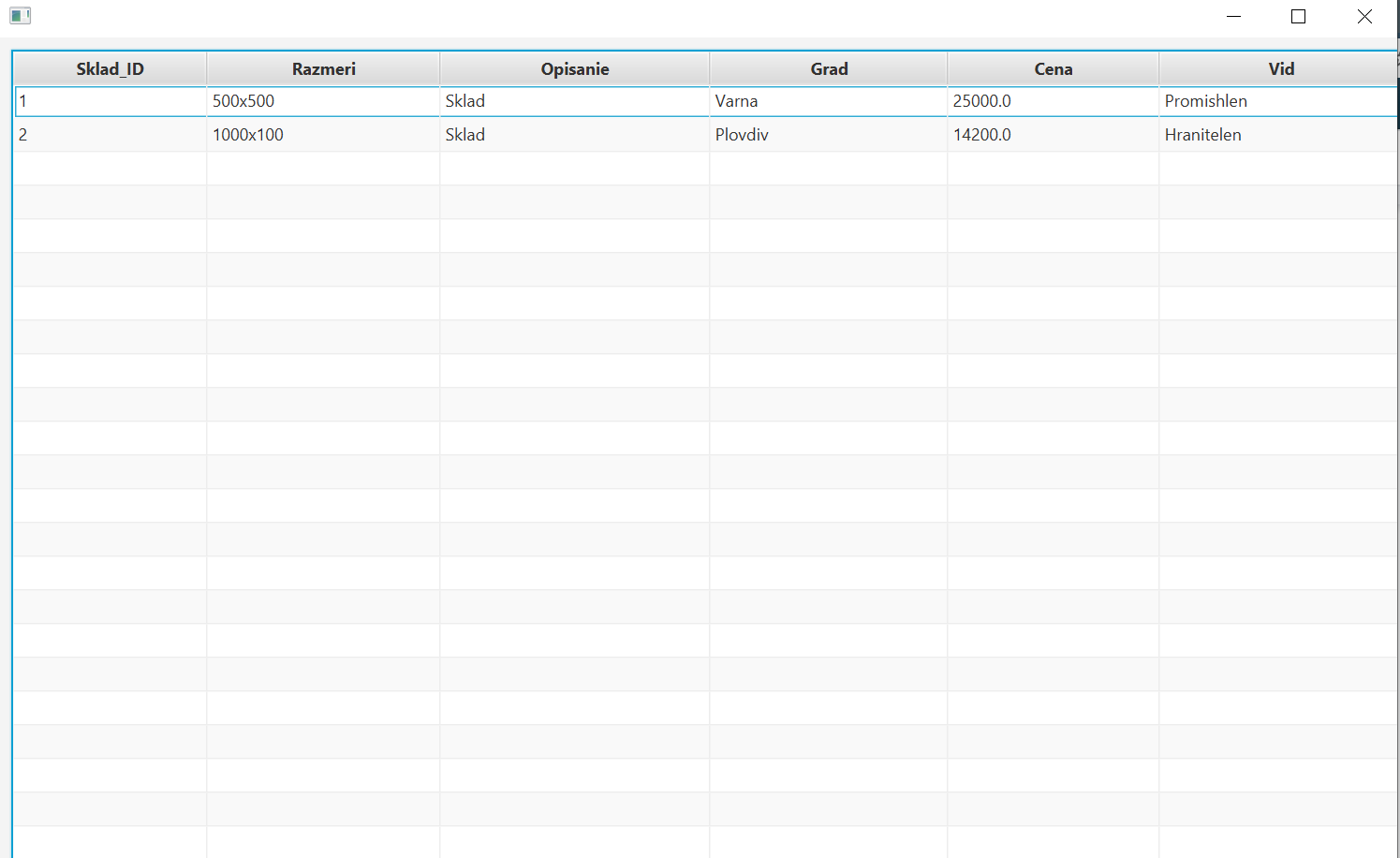
Фиг. Собственик Панел



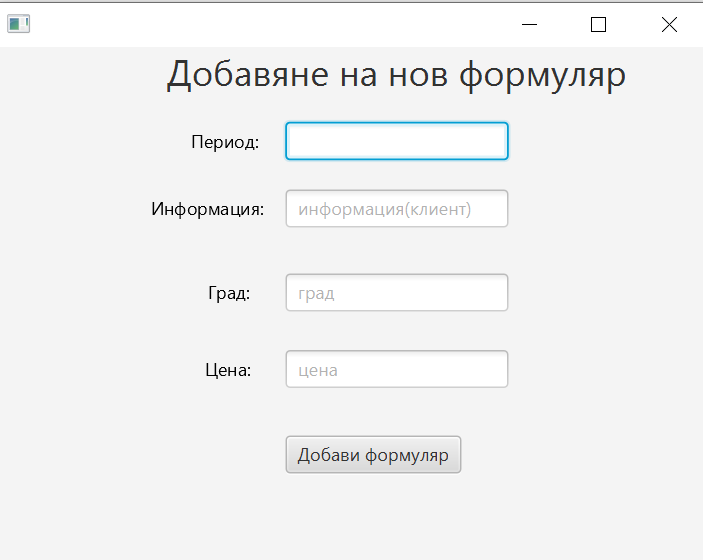
Фиг. Добавяне на склад



Фиг. Списък с агенти



Фиг. Списък със складове



Фиг. Добавяне на формуляр