

**ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – ВАРНА**

**ФАКУЛТЕТ ПО ИЗЧИСЛИТЕЛНА ТЕХНИКА И АВТОМАТИЗАЦИЯ**

**КАТЕДРА „СОФТУЕРНИ И ИНТЕРНЕТ ТЕХНОЛОГИИ“**

**Специалност „Софтуерни и интернет технологии“**

**Проект по Обектно – ориентирано програмиране част 2**

**Тема 8**

**Производство на вина**

**Изготвили:**

Владимир Андреев и Александър Живков

(Спец. СИТ, гр. 4б, Фак. №: 19621703, 19621702)

**ВАРНА**

**2021**

**Условие на проекта**

**VIII. Производство на вина**   
  
Да се разработи информационна система, предоставяща възможност за организиране на производството на различни типове вино.

Системата поддържа три вида потребители администратор,оператори   
(винопроизводители) и домакин на склада с различни роли за достъп до   
функционалностите в системата.   
  
Операции за работа с потребители:

• създаване на оператори от администратор;   
• създаване на домакини от администратор;

Системата поддържа операции за винопроизводство:

• регистриране на сорт грозде и количеството му в склада(сортовете се делят на   
две категории - бели и черни);   
• дефиниране на количеството вино което може да се получи от килограм грозде   
от даден сорт;   
• регистриране на видове и количество бутилки за бутилиране на вино(750, 375,   
200 и 187ml);   
• всеки тип вино се състои от един или няколко сорта и за направата му е   
необходимо определено количество от тях;   
• дефиниране на необходимите количества грозде за производството на даден   
вид вино;   
• бутилиране на виното в бутилки (бонус – системата автоматично да изчислява   
оптимално запълване на бутилки);

Системата поддържа Справки по произволен период за:

• наличности на сортове грозде;   
• наличност на видове бутилки;   
• наличности на видовете бутилирани вина;

Системата поддържа Известия за:

• критичен минимум и липса на даден сорт грозде;   
• критичен минимум и липса на даден вид бутилки;

Анализ на проблема

Системата трябва да поддържа три вида потребители, които изпълняват различни роли и имат различни нива на достъп до различни функции в програмата. Всеки потребител ще има уникални данни за идентификация.

Потребител от тип „Администратор“ ще има достъп до всички функционалности на системата, а именно:

* създаване на нови потребители от всеки един тип;
* регистриране на вид бутилка и нейното количество;
* регистриране на цветове грозде;
* регистриране на сортове грозде, тяхното количество в склада и количеството вино което се произвежда от един килограм от тях;
* регистриране на ново вино от един или няколко сорта грозде;
* регистриране на цикъл на производство за съответно избраното вино;
* проверка на налични количества бутилки;
* проверка на налични количества грозде;
* проверка на налични количества вино;
* проверка на вече завършени производсвени цикли;
* изтриване на потребители (от всяко ниво);
* изтриване на бутилки;
* изтриване на цветове грозде;
* изтриване на сортове грозде;
* изтриване на вина;
* изтриване на производствени цикли;
* обновяване на количество бутилки;
* обновяване на количесво грозде;
* обновяване на количесво вино;
* проверка за критична наличност на бутилки;
* проверка за критична наличност на сорт грозде;

Потребител от тип „Оператор“ ще има достъп до следните функционалности

на системата:

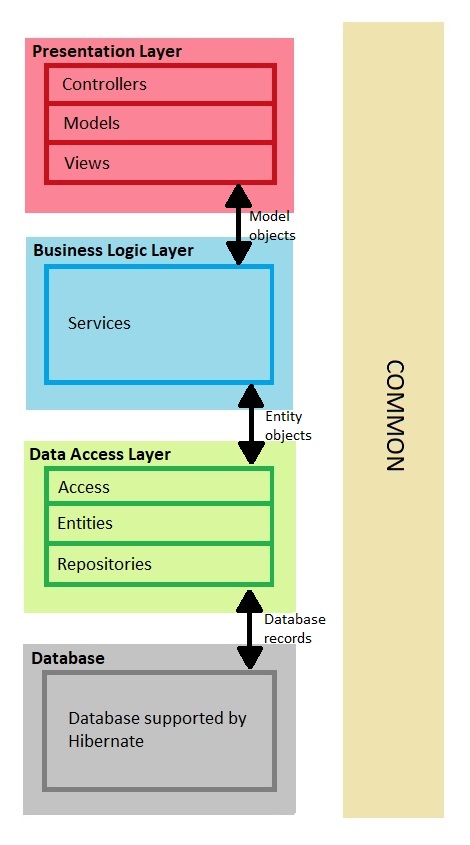
* регистриране на ново вино от един или няколко сорта грозде;
* регистриране на цикъл на производство за съответно избраното вино;
* проверка на налични количества бутилки;
* проверка на налични количества грозде;
* проверка на налични количества вино;
* проверка на вече завършени производсвени цикли;
* изтриване на вина;
* изтриване на производствени цикли;
* обновяване на количесво вино;
* проверка за критична наличност на бутилки;
* проверка за критична наличност на сорт грозде;

Потребител от тип „Домакин на склад“ ще има достъп до следните функционалности на системата:

* регистриране на вид бутилка и нейното количество;
* регистриране на цветове грозде;
* регистриране на сортове грозде, тяхното количество в склада и количеството вино което се произвежда от един килограм от тях;
* проверка на налични количества бутилки;
* проверка на налични количества грозде;
* проверка на налични количества вино;
* изтриване на бутилки;
* изтриване на цветове грозде;
* изтриване на сортове грозде;
* изтриване на вина;
* обновяване на количество бутилки;
* обновяване на количесво грозде;
* проверка за критична наличност на бутилки;
* проверка за критична наличност на сорт грозде;

Системата ще следи и извежда конретния потребител, който се е вписал. Също така ще извежда известия при достигане на критичен минимум или липса на бутилка от даден вид, както и на даден сорт грозде. При вписване на потребител ще се извършва проверка за коректност на данните и при липса на съвпадение ще се извежда известие, и ще бъде отказан достъп. Потребителите ще имат достъп до функционалностите на системата чрез модерен и интуитивен потребителски интерфейс.

Структура и модули на проекта



За реализацията на възложения проект е избрана многослойна архитектура, представена от слоевете посочени на горната снимка, предоставената от нея възможност за ясното разграничение на нейните модули и по-лесна имплементация на нови функционалности и корекции по системата.

Презентационният слой е изграден от три модула – Controllers, Models и Views. В модул views се намират javafx файлове, които изграждат потребителския интерфейс и изгледа на системата. Те съдържат основните компоненти на прозорците като диалогови кутии, бутони и т.н. В модул controllers се намират класовете, които осъществяват връзката между въведените от потребителя данни и тяхната обработка в бизнес логиката. Също така те управляват отделните елементи в javafx файловете. В модул models се намират обектите в които се записва въведената от потребителя информация, която се предава на бизнес логиката. Чрез тях се постига индиректната връзка между данните от потребителя и записите в базата данни.

Слоят бизнес логика обработва информацията и реализира отделните функционални изисквания на системата. Той преобразува моделите в entity обекти, които се използват от слоя за връзка с базата данни.

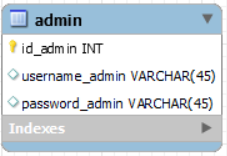
Слоят за връзка с базата данни се разделя на три модула – Access, Entities и Repositories. В модул access се осъществява същинската връзка с базата данни, чрез която се достъпват отделните записи. Модул entities съдържа класове съответстващи на таблиците от базата данни и връзките между тях. Във всеки един клас са реализирани методи, които позволяват достъпването, сравнението и визуализацията на отделните записи от съответната таблица. В модул repositories се намират класове в които се имплементират различните заявки към базата данни.

Връзката с базата данни е реализирана чрез Hibernate. Избран е този подход поради автоматичното генериране на заявки от Hibernate, както и сравнително лесната му имплементация. За реализация на базата данни е избран MySQL, поради широката му разпространеност и лесното му проектиране.

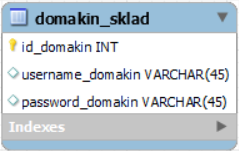
В слоя common са поместени класовете, които се използват за зареждането на различните страници от потребителския интерфейс, класът изпозван за указване на пътя до конфигурацията на логъра за събития, както и класът който се използва за следете на конкретния потребител използващ системата.

Проектиране и реализация на системата

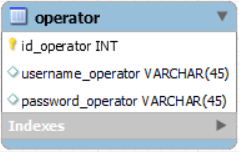
Проектиране на базата данни:



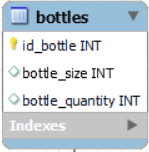
В таблица admin се записват данните на потребител от тип Аdmin. Те са име на потребителя и парола представени съответно чрез username\_admin и password\_admin, които са от тип varchar с дължина 45, което дава по-голяма гъвкавост в избора на данни. За идентификация на отделните потребители се използва id\_admin, което е от тип int, защото се инкрементира автоматично чрез auto increment.



В таблица domakin\_sklad се записват данните на потребител от тип Warehouse\_host. Те са име на потребителя и парола представени съответно чрез username\_ domakin и password\_ domakin, които са от тип varchar с дължина 45, което дава по-голяма гъвкавост в избора на данни. За идентификация на отделните потребители се използва id\_ domakin, което е от тип int, защото се инкрементира автоматично чрез auto increment.

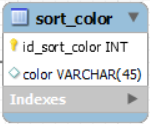


В таблица operator се записват данните на потребител от тип Operator. Те са име на потребителя и парола представени съответно чрез username\_operator и password\_ operator, които са от тип varchar с дължина 45, което дава по-голяма гъвкавост в избора на данни. За идентификация на отделните потребители се използва id\_ operator, което е от тип int, защото се инкрементира автоматично чрез auto increment.

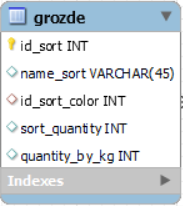


В таблица bottles се записват данните за определена бутилка, които могат да бъдат размер и количество и са представени съответно чрез bottle\_size и bottle\_quantity. Те са от тип int поради фактът че количесвото трябва да бъде цяло число, както и размерът. За идентификация на отделните бутилки се използва id\_ bottle, което е от тип int, защото се инкрементира автоматично чрез auto increment.

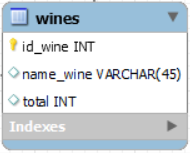
Таблицата се свързва с таблица production с връзка едно към много.



В таблица sort\_color се записват различните цветове, които могат да бъде дадено грозде. Данните се записват в поле color, което е от тип varchar понеже е текст. За идентификация на отделните цветове се използва id\_sort\_color, което е от тип int, защото се инкрементира автоматично чрез auto increment. Таблицата се свързва с таблица grozde с връзка едно към много.



В таблица grozde се съдържа информация за дадено грозде. Тази информация се състои от неговото име (name\_sort), от тип varchar, цвета му (id\_sort\_color), който се избира от таблица sort\_color, наличното на склад количеството в килограми (sort\_quantity), което е от тип int, количеството вино което може да се произведе от един килограм от даденото грозде в милилитри (quantity\_by\_kg), от тип int. За идентификация на отделните сортове грозде се използва id\_sort, което е от тип int, защото се инкрементира автоматично чрез auto increment. Осъществена е връзка с междинна таблица с grozde\_wine от тип едно към много.



В таблица wines се съдържа информация за дадено вино. Тази информация се състои от неговото име (name\_wine), от тип varchar с дължина 45 символа, общото му количесво в скада в килограми (total), което се състои количеството вино което може да се произведе от един килограм от гроздата от които се състои даденото вино, умножено по необходимото количество грозде за направата на това вино, разделено на хиляда (за да се преобразува в килограми). За идентификация на отделните вина се използва id\_wine, което е от тип int, защото се инкрементира автоматично чрез auto increment. Осъществена е връзка с междинна таблица с grozde\_wine от тип едно към много.

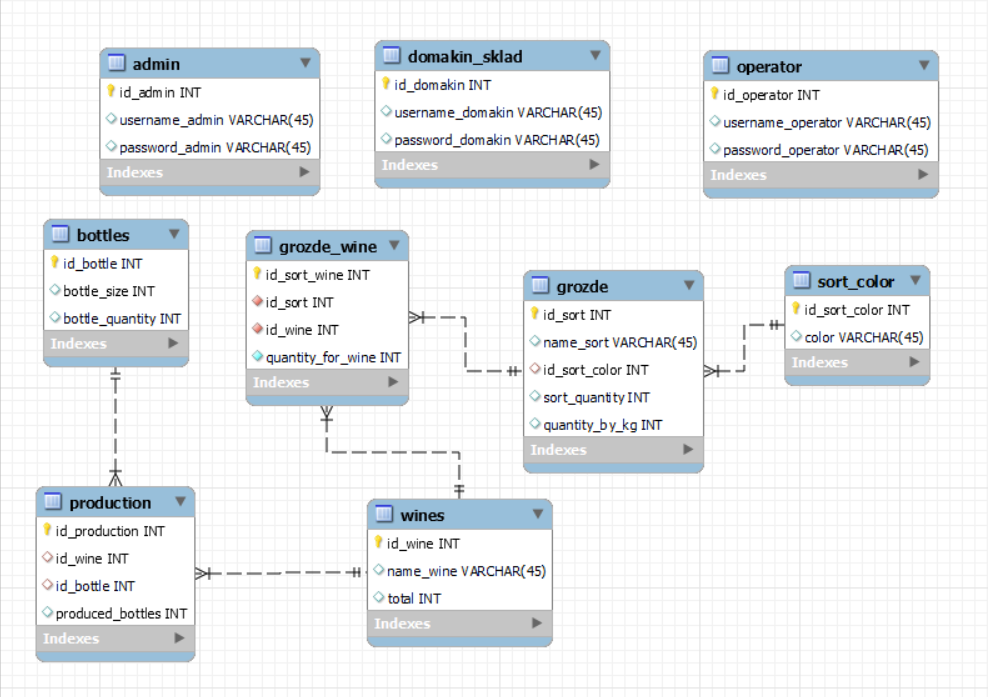


Таблицата grozde\_wine е междинна таблица в която се описват сортовете грозде, съставящи едно вино, както и необходимото количество от даден сорт за неговото производство. Всеки един запис се следи чрез id\_sort\_wine, от тип int, защото се инкрементира автоматично чрез auto increment. Тя има осъществени връзки с таблиците grozde и wines от тип много към едно.



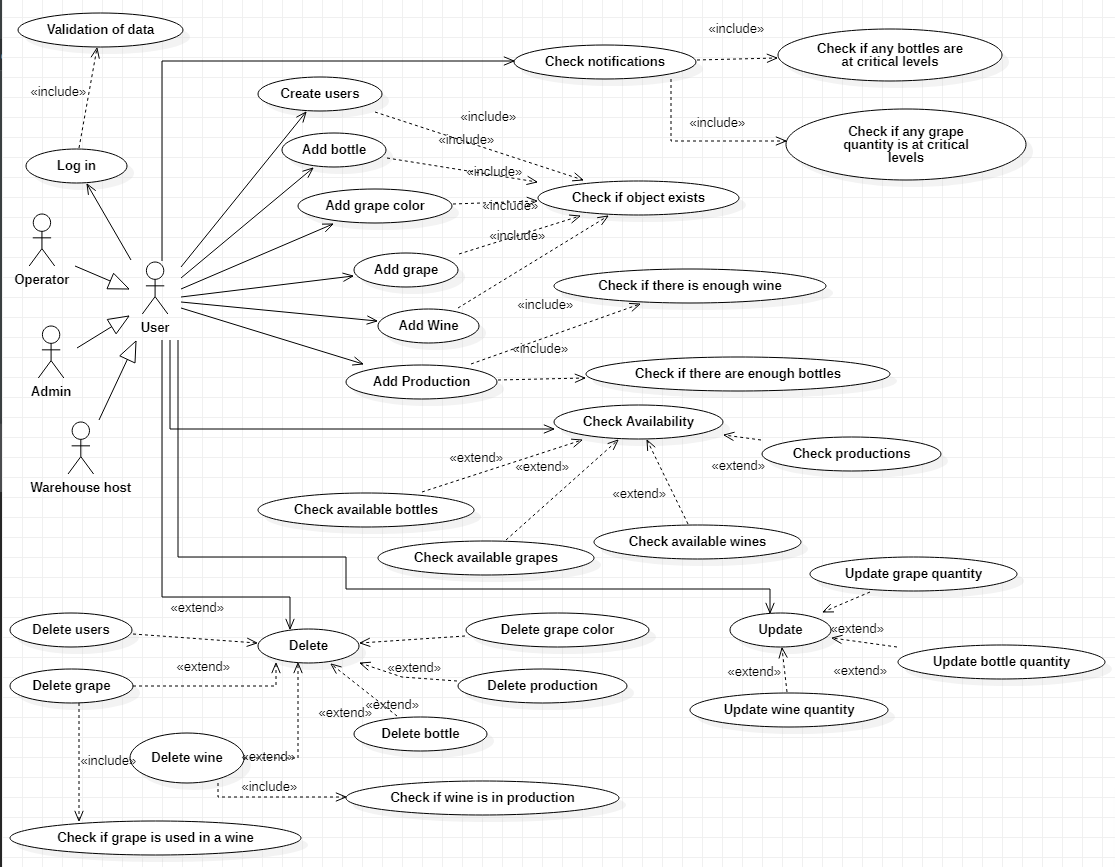
В таблицата production се записват данните за всеки производствен цикъл. Той се състои от дадено вино (id\_wine), избран вид бутилка (id\_bottle) и брой бутилки които ще се произведат (produced\_bottles), който е от тип int. Всеки производствен цикъл има id\_production , което е от тип int, за да може да се идентифицира. Таблицата има връзки към bottles и wines, които са от тип много към едно.

**ER диаграма**

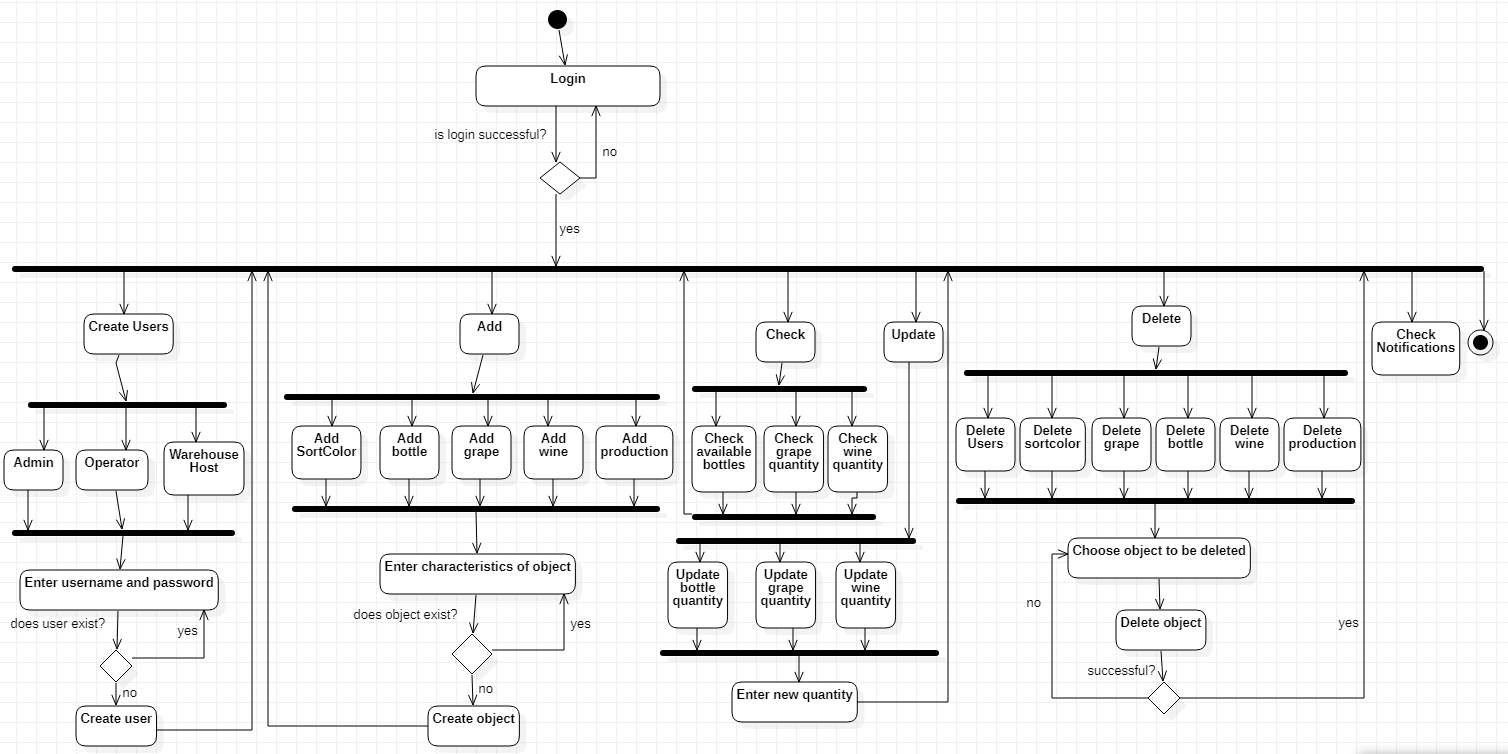


**UML диаграми**

**Use-case диаграма:**

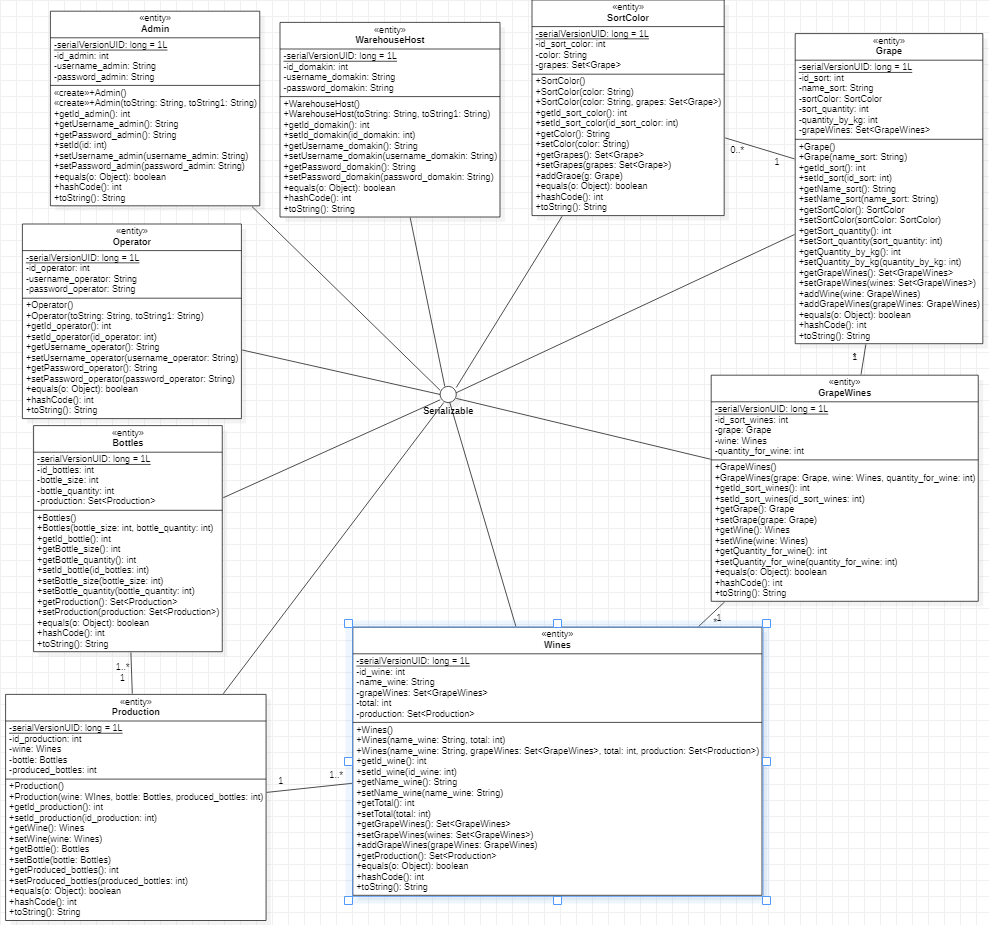


**Activity диаграма:**

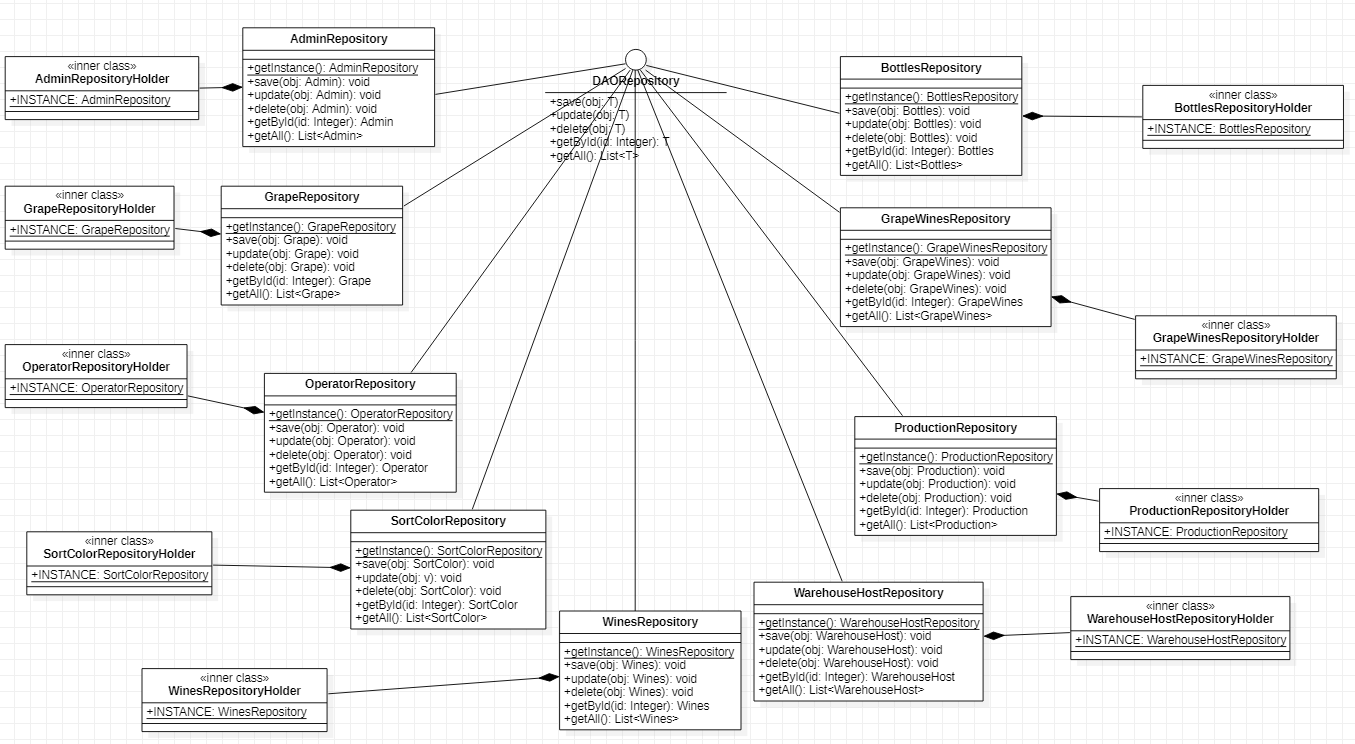


**Class диаграми:**

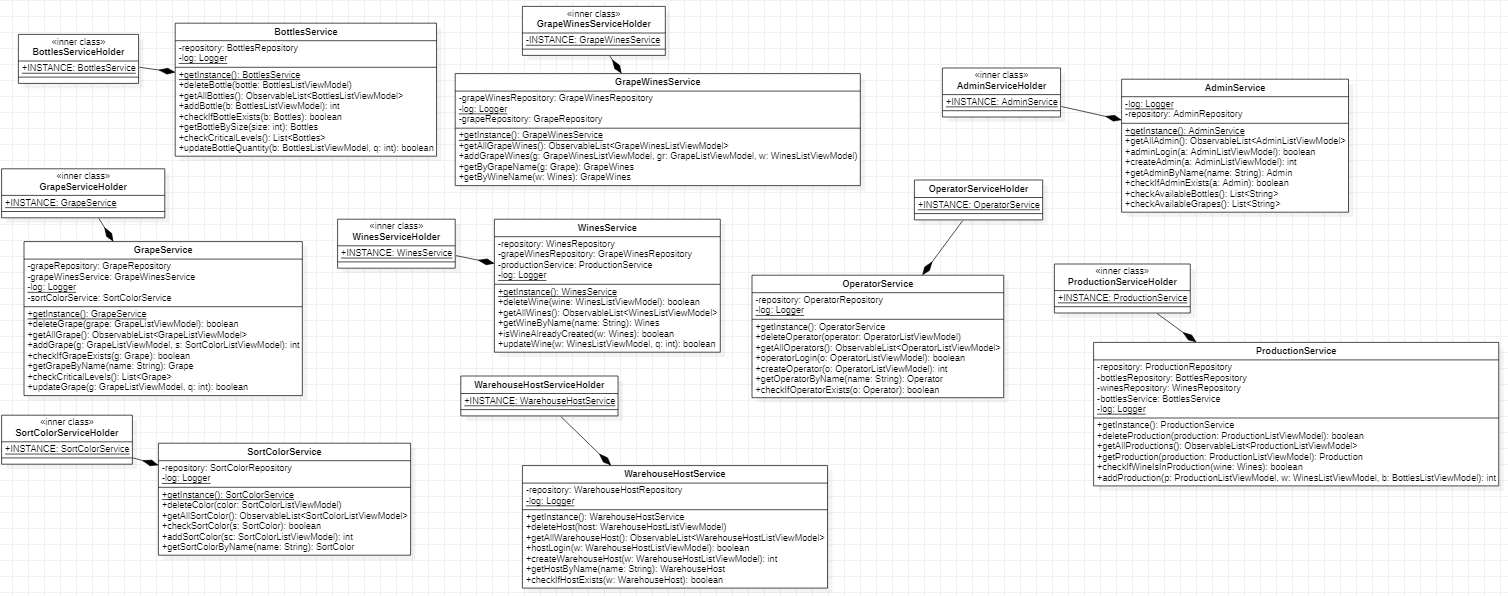
**-Entities**



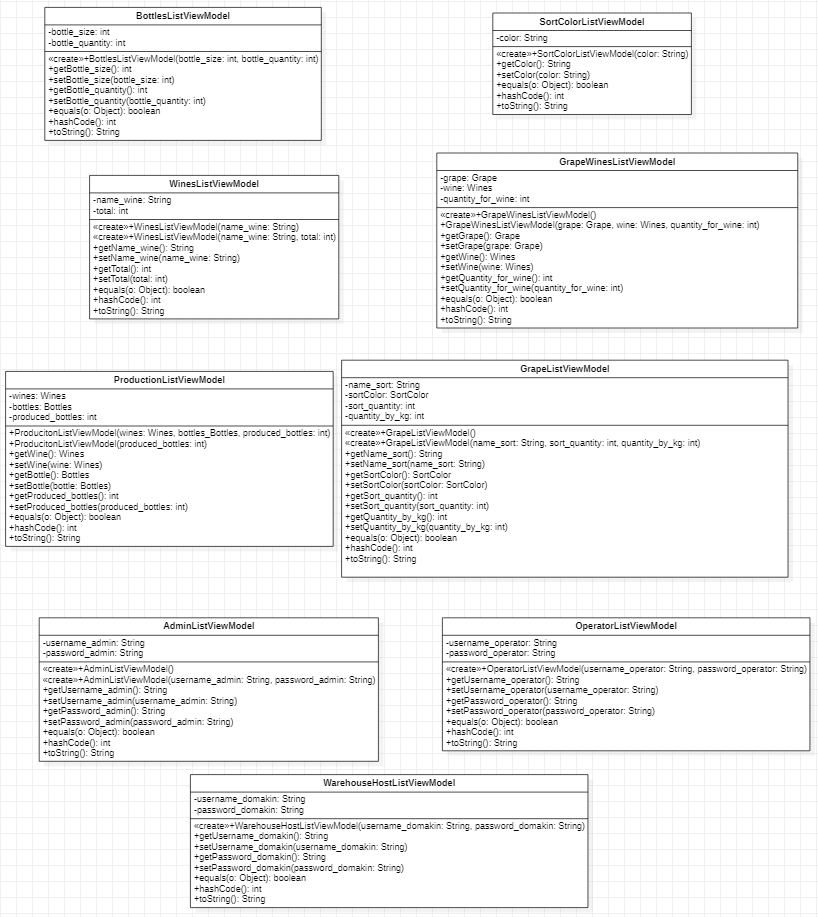
**-Repositories**



**-Business services**

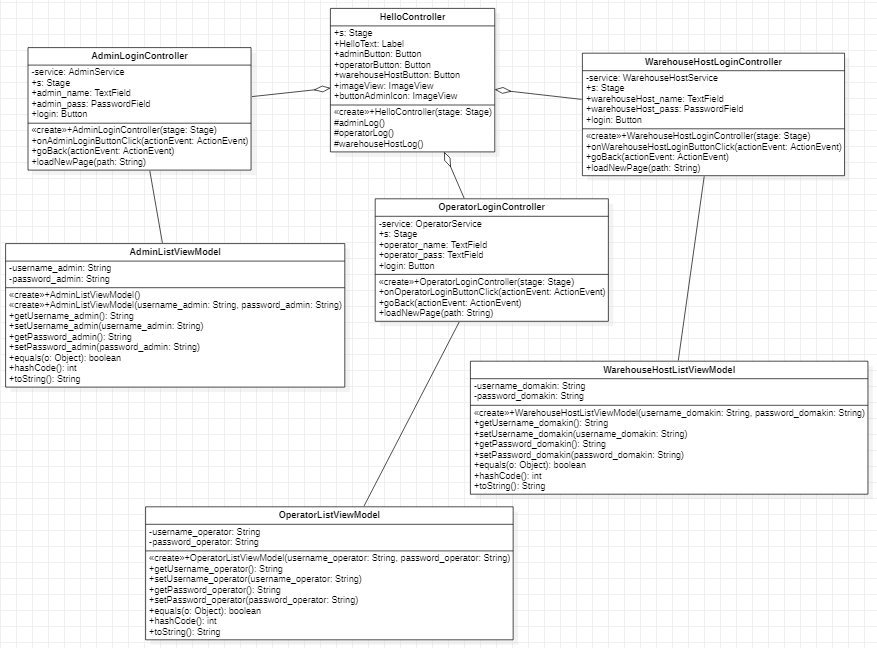


**-ListViewModels**

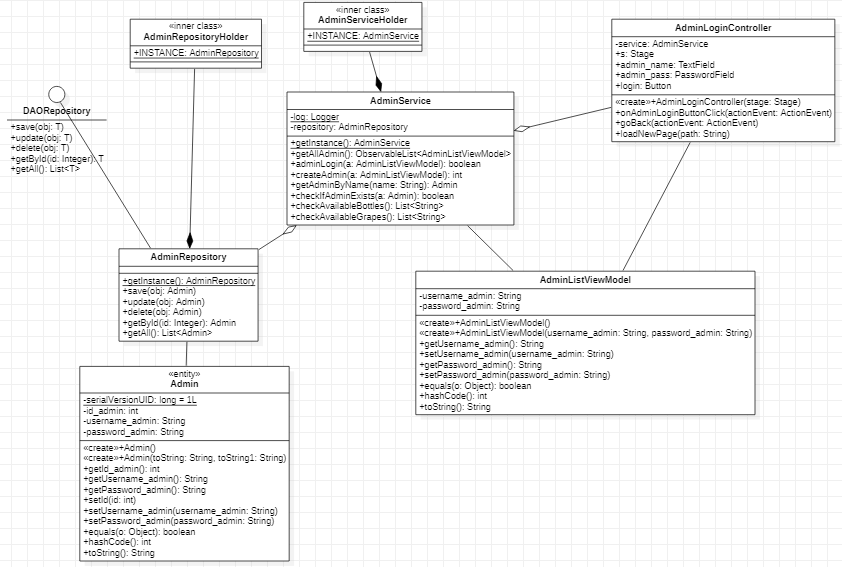


**Клас диаграми за функционални изисквания**

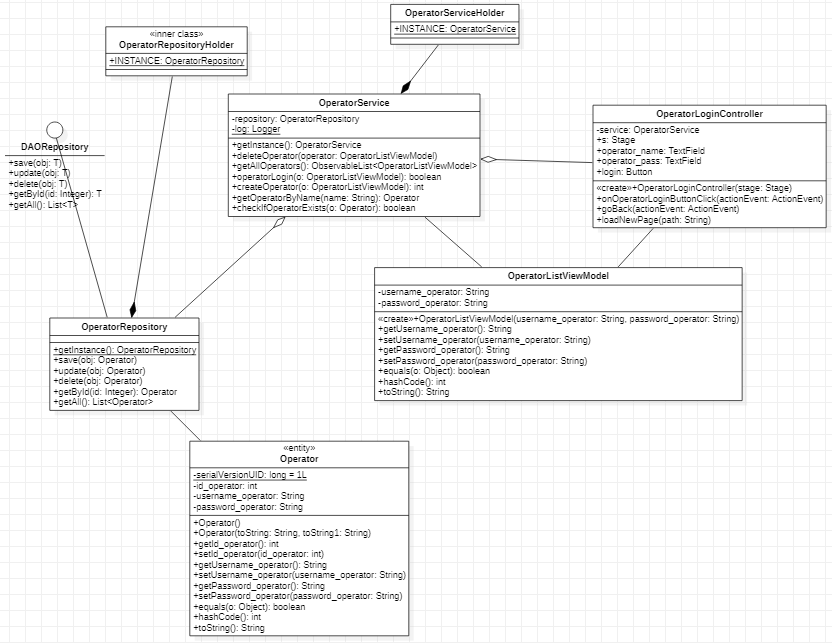
**-Начална страница**



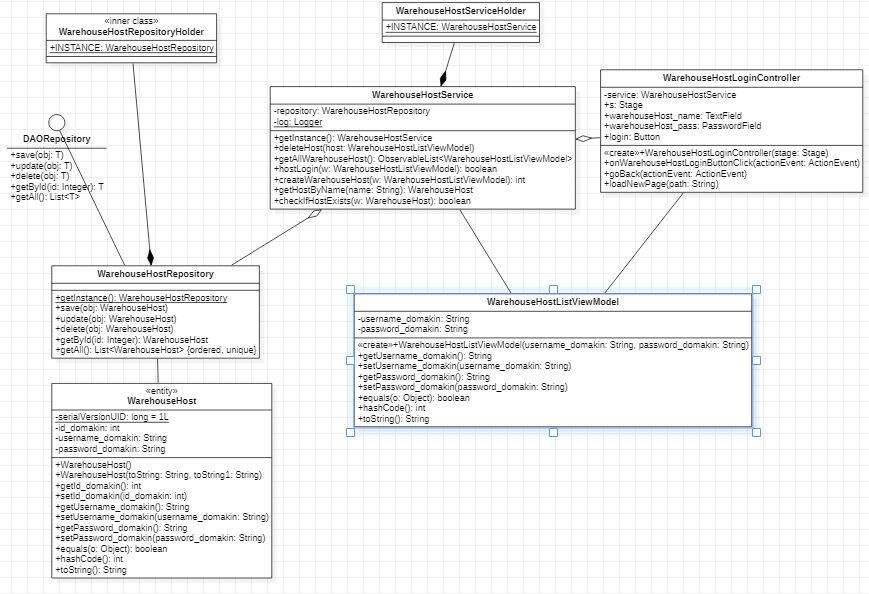
**-Логване като админ**



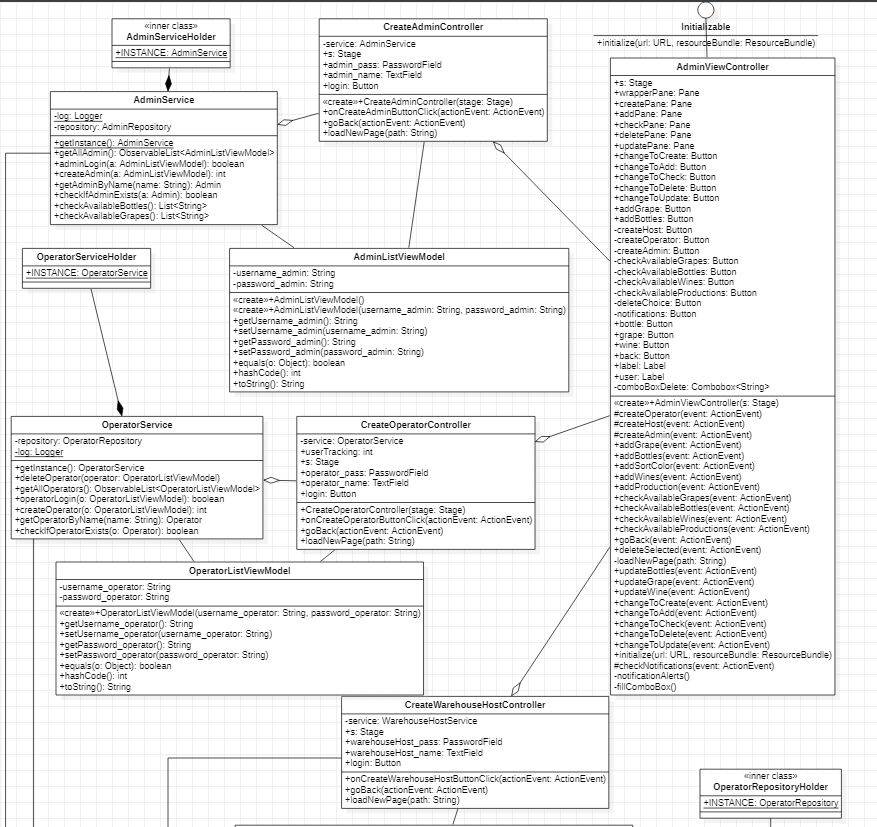
**-Логване като оператор**

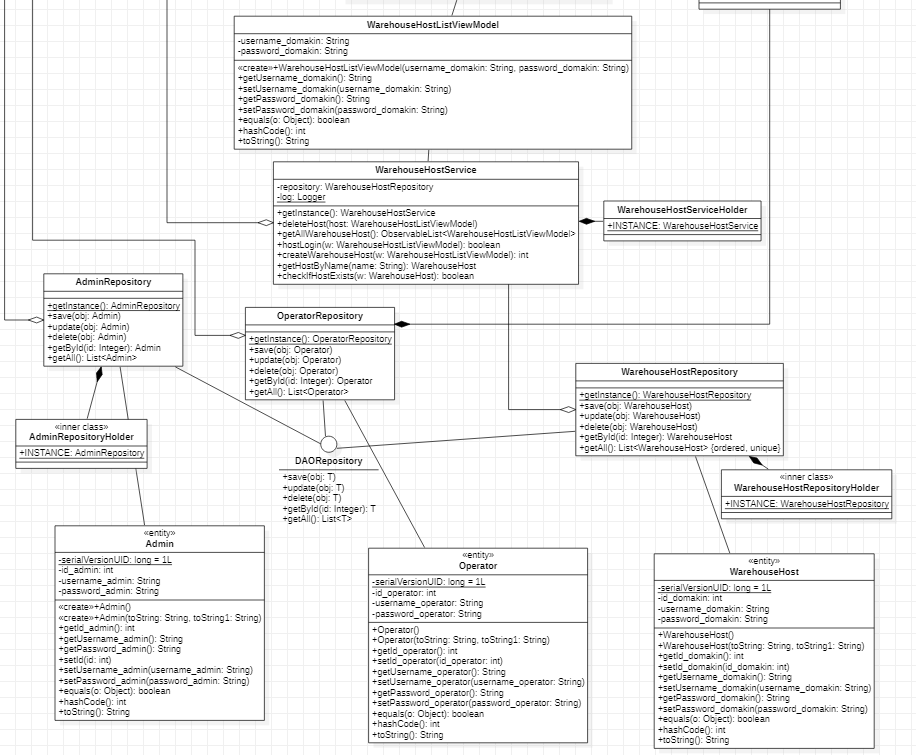


**-Логване като домакин на склад**

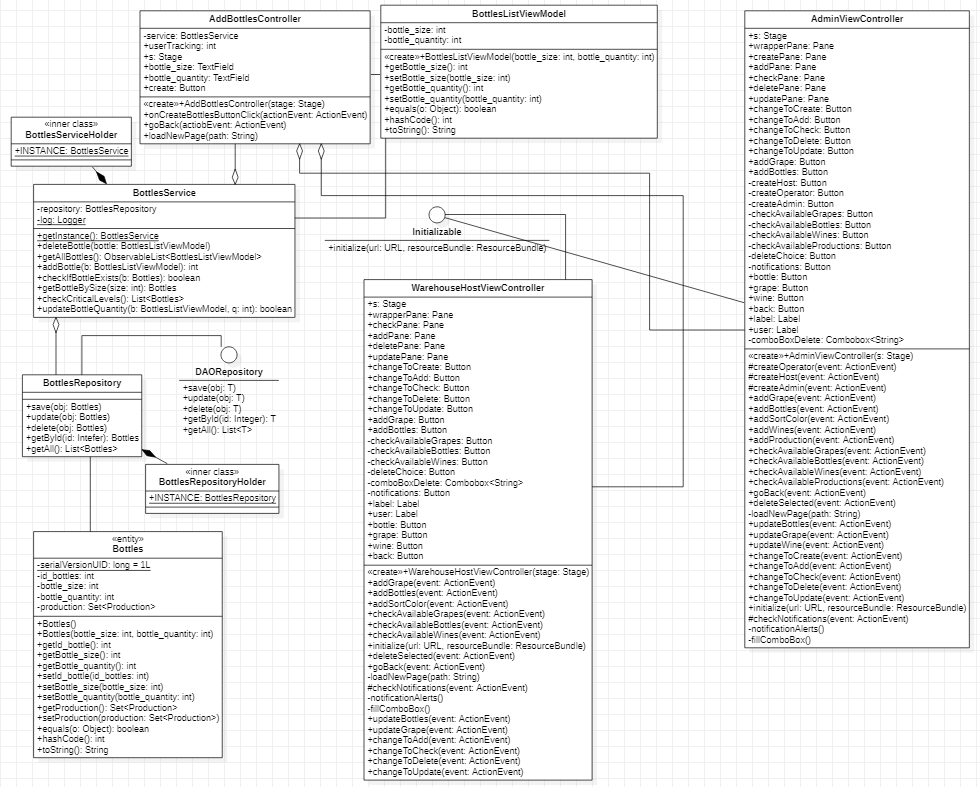


**-Създаване на потребители**

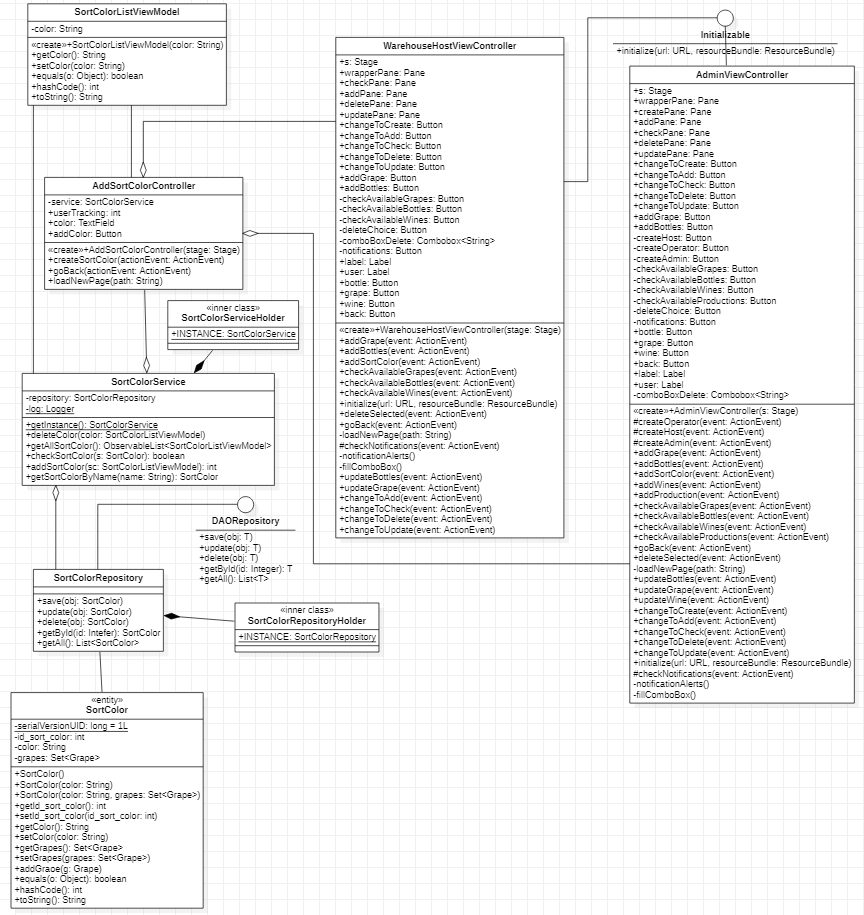




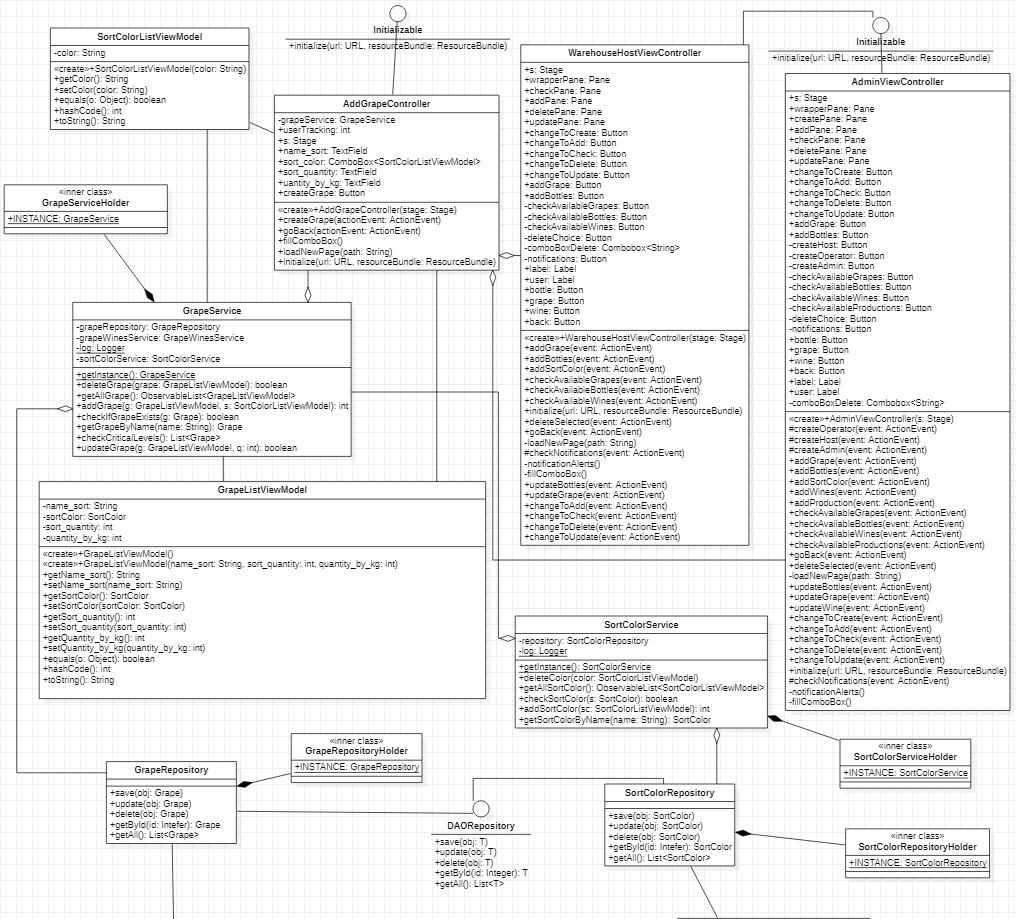
**-Създаване на бутилка**

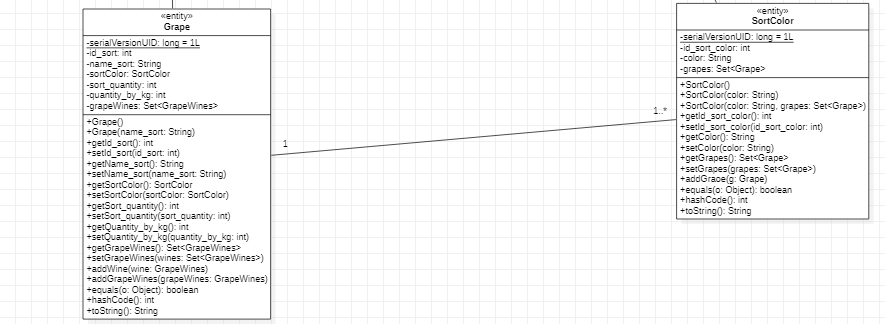


**-Създаване на цвят грозде**

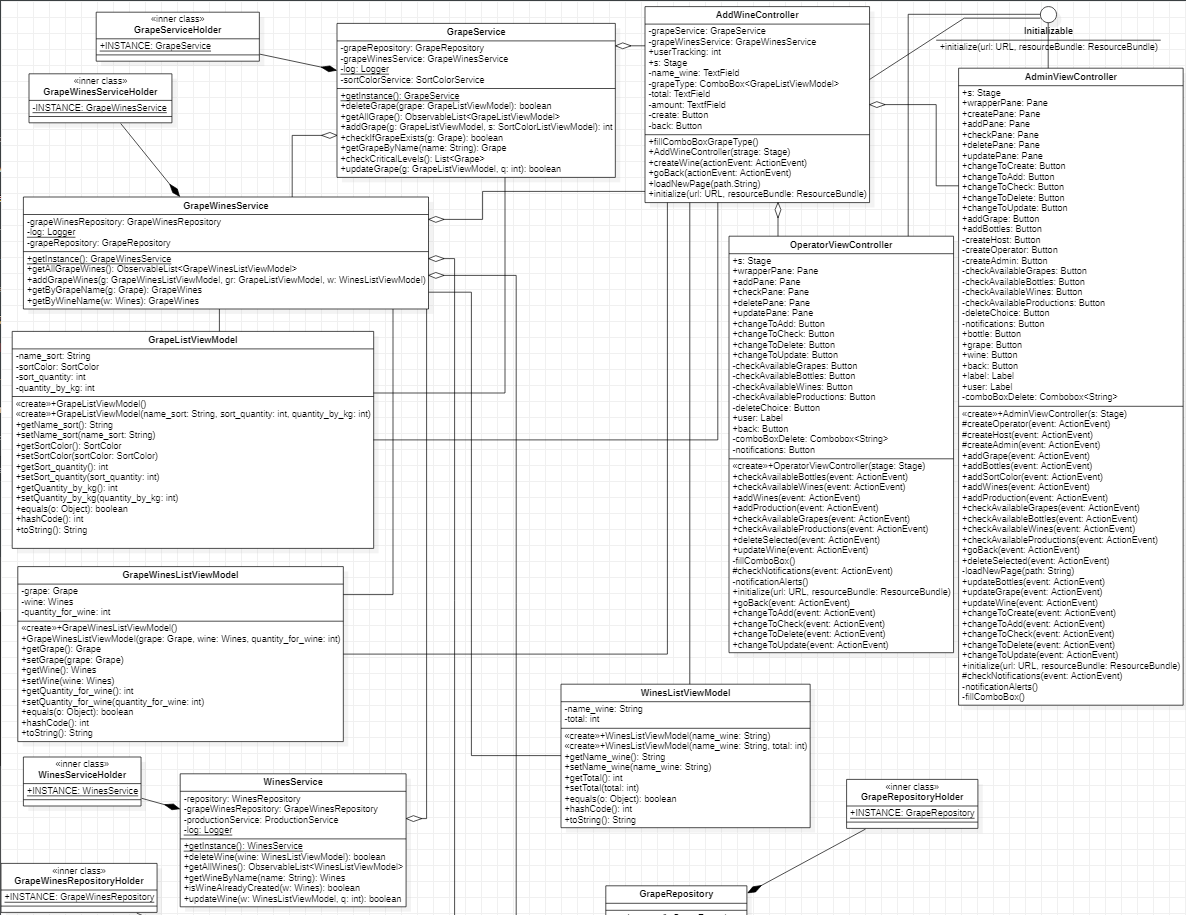


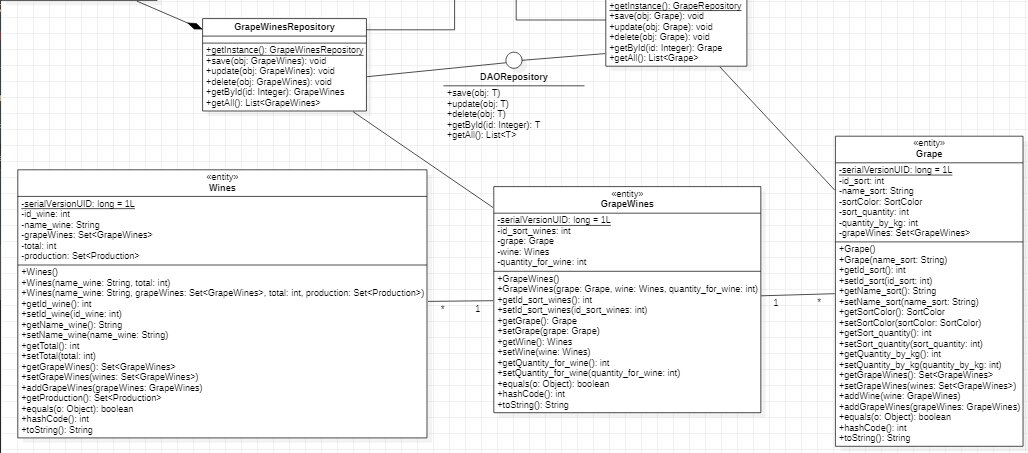
**-Създаване на грозде**





**-Създаване на вино**





**-Създаване на производствен цикъл**

