

**ФАКУЛТЕТ ПО ПРИЛОЖНА МАТЕМАТИКА И ИНФОРМАТИКА**

*ТЕХНИЧЕСКИ УНИВЕРСИТЕТ СОФИЯ*

*София*

*2022г.*

Изготвила:

Савина Вълчанова

Фак. номер: 471220071

II курс 77 гр.

КУРСОВ ПРОЕКТ

ПО

СОФТУЕРНИ АРХИТЕКТУРИ

**Задание: Online магазин за дрехи**

**Съдържание:**

1. Речник на термини..............................................................................3стр.
2. Въведение............................................................................................3стр.
3. Цели, обхват и очаквани резултати от изпълнението на проекта..3стр.
   1. Общи и специфични цели на проекта....................................3стр.
   2. Целева група.............................................................................4стр.
   3. Очаквани резултати..................................................................4стр.
4. Функционалност.................................................................................5стр.
   1. Структура на потребители.......................................................5стр.
   2. Роли в приложението...............................................................6стр.
   3. Логически изглед за програмистите.......................................9стр.
   4. Процес на поръчка..................................................................10стр.
5. Функционални изисквания..............................................................10стр.
6. Нефункционални изисквания..........................................................11стр.
   1. Общи нефункционални изисквания.....................................11стр.
   2. Изграждане и поддръжка на множество среди...................12стр.
7. **Речник на термини**

|  |  |
| --- | --- |
| Термин | Описание |
| 1. Система бази данни | Набор от компютърни програми, контролиращи изграждането, поддръжката и използването на бази от данни. |
| 1. Use-case (сценарий) | Техника за определяне на функционалните изисквания на една система. |
| 1. Логически изглед | Представяне на системата от гледна точка на фукционалността. |
| 1. Верификация | Установяване, удостоверяване на истината. |
| 1. Централизация | Вид организация, където всички важни права и правомощия са в ръцете на висшето ръководство. |
| 1. Интерфейс | Осигурява логика на програмиране. |
| 1. HTTP/2 | Протокол за пренос на хипертекст. Пренася информация в компютърни мрежи. |
| 1. Continuous integration | Обединяване на всички копия, по които програмистите работят, в споделена главна линия няколко пъти на ден. |

1. **Въведение**

Целта на настоящия документ е да опише софтуерните изисквания за изпълнението на примерна обществена поръчка на тема: „Online магазин за дрехи“.

В настоящото техническо задание са описани и изискванията към проектната реализация и документация.

1. **Цели, обхват и очаквани резултати от изпълнението на проекта**
   1. **Общи и специфични цели на проекта**

Реализиране на система за пазаруване в онлайн среда с предназначение улесление на потребителите.

Постигането на общата цел ще бъде реализиранa чрез следните специфични цели, съотвестващи на планираните по проекта дейности:

* Повишаване на информираността на потребителите относно нови продукти и наличности;
* Улесление на пазаруването чрез онлайн среда;
* Система от бази данни*1* с направени поръчки и контакти на потребителите за бъдещи поръчки;
* Гарантирана сигурност на данните на потребителите;
* Бърз и лесен достъп до всяка страница;
  1. **Целева група**

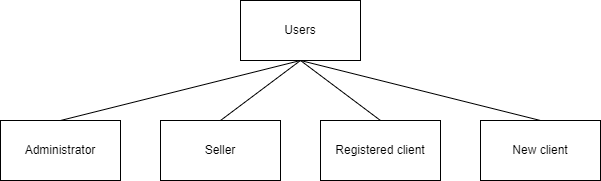
Целевата група, към която е насочен проектът, обхваща потребители/клиенти;

* 1. **Очаквани резултати**

Очакваните резултати от изпълнението на настоящия проект са:

* Работоспособен софтуерен продукт за създаване на поръчки;
* Обработка на лични данни с цел доставяне на продукти от заявената поръчка;
* Лесен и бърз достъп до магазина от десктоп компютър или мобилно приложение;
* Подготвени каталози с избор на много различни продукти;
* Регулярна актуализация на каталози;
* Леснодостъпно попълване на лични данни;
* Регистрация на клиенти;
* Логване на клиенти, продавачи и администратор;
* Добавяне на продукти в количката;
* Възможност за разплащане с банкова карта или PayPal;
* Публикуване на ревю след получаване на продуктите;
* Достъп до историята с направените поръчки;

1. **Функционалност**
   1. **Структура на потребители**

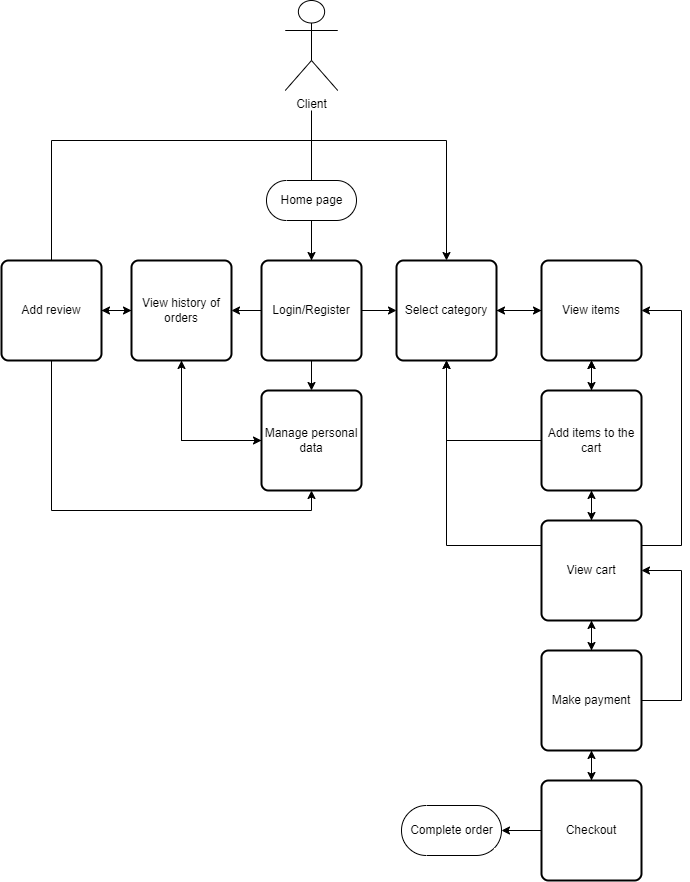
****

*Фиг.1*

Всички потребители на системата са обощение чрез Users (*фиг.1*), а те се делят на администратор, продавач, регистриран и не-регистриран клиент.

* 1. **Роли в приложението**

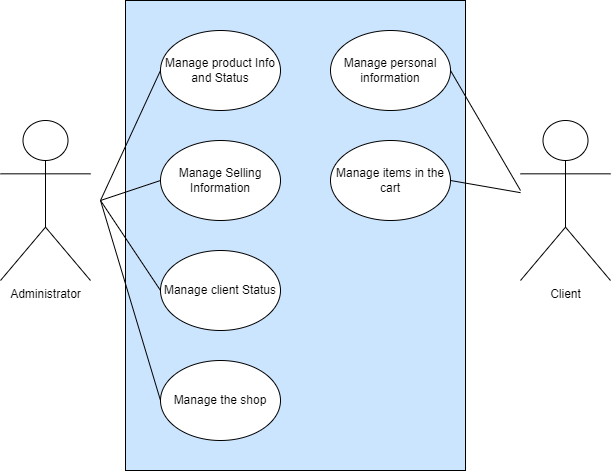
Диаграмите показват основните use-case (сценарии)*2* :



*Фиг.2*

Диаграмата (*Фиг.2*) показва постъпково процесите, през които клиента е възможно да премине като винаги се започва от началната страница на магазина.

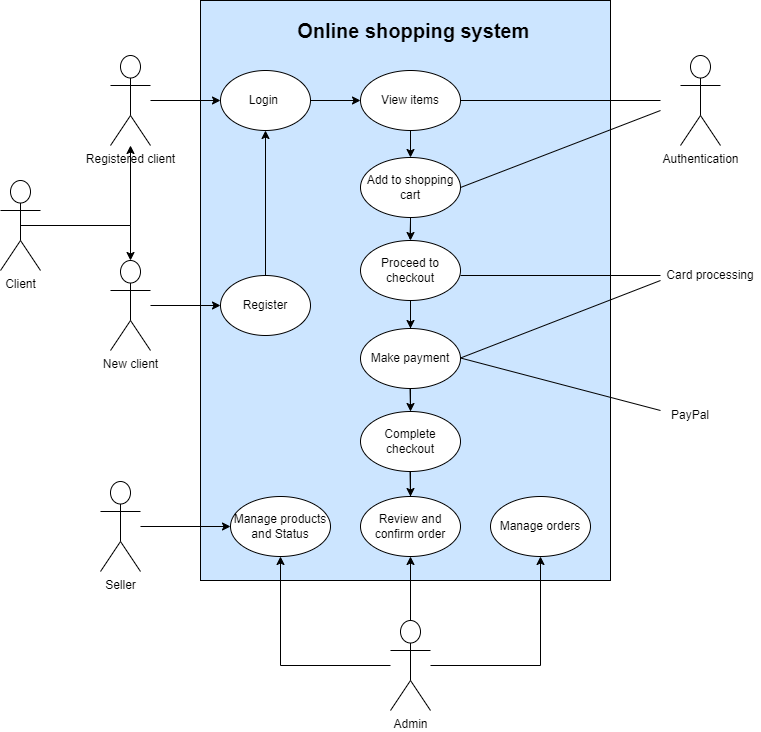
* От там може да се логне или регистрира (ако не се е). Потребителят има възможност да си погледне историята на поръчките, да редактира личните си дании и да избере категория от началната страница.
* При историята с поръчки може да остави ревю.
* След разглеждането на продуктите се добавят в количката, която по всяко време може да се преглеждат вече добавените продукти.
* Въвеждат се банковите данни и се предоставя възможност за цялостно разглеждане на поръчка.
* Потвърждава се и бива изпратена към базата данни със останалите заявки за разглеждане от администратора.
* Крайната цел в този случай е заявена поръчка, като тя не е задължителна, т.е. потребителят може само да разгледа продуките спрямо категориите.



*Фиг.3*

Диаграмата *(Фиг.3)* показва опциите по редакция на информацията, като:

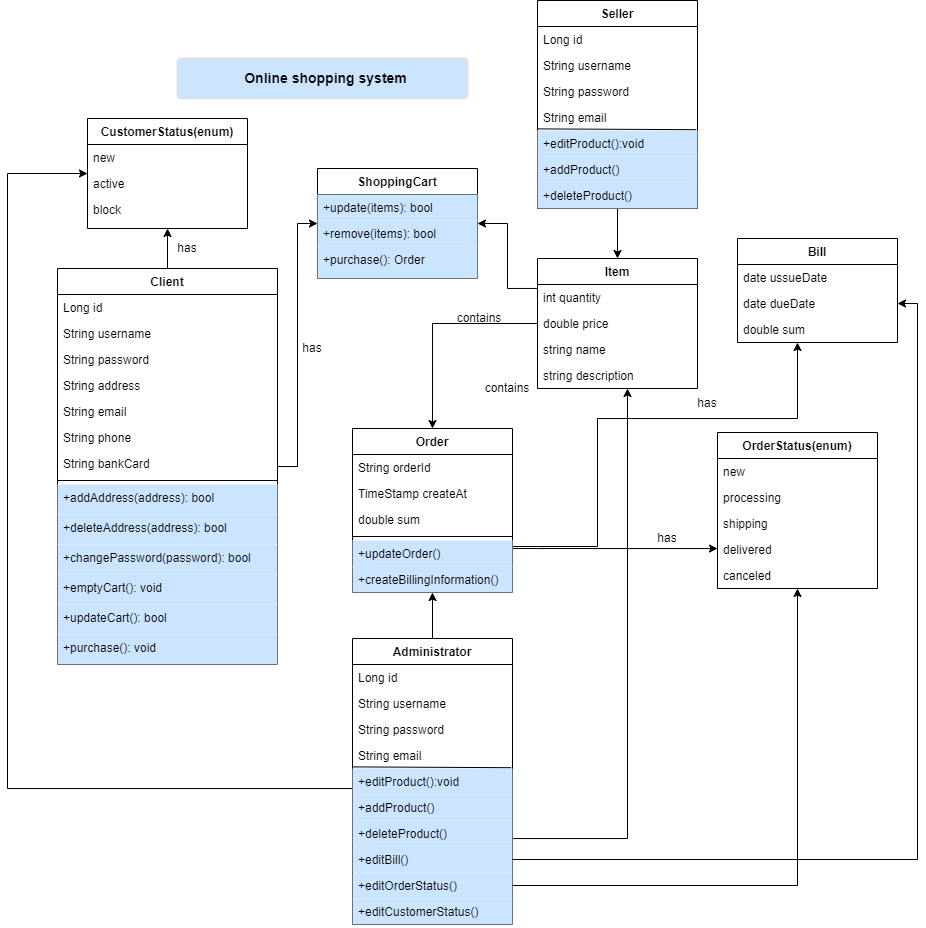
* администраторът има пълния контрол върху магазина и статусът на потребителя;
* клиентът може да коригира личните си данни и да добави/премахне продукти в количката;



*Фиг.4*

Диаграмата (*Фиг.4*) изобразява най-общо пълната система от всички потребители и процеси, които се очакват от използването на софутерния продукт.

1. Клиентите се регистрират и/или логват;
2. Разглеждат налчините продукти;
3. Добавят в количката (през това време се идентифицират);
4. Въвеждат данните си за доставка и добавят банкова карта, ако решат да плащат през PayPal;
5. Разглеждат поръчката и потвърджават;
6. Администраторът преглежда направената заявка;
7. Администраторът контролира заявките за поръчки и информацията за продуктите;
   1. **Логически изглед*3* за програмистите**

****

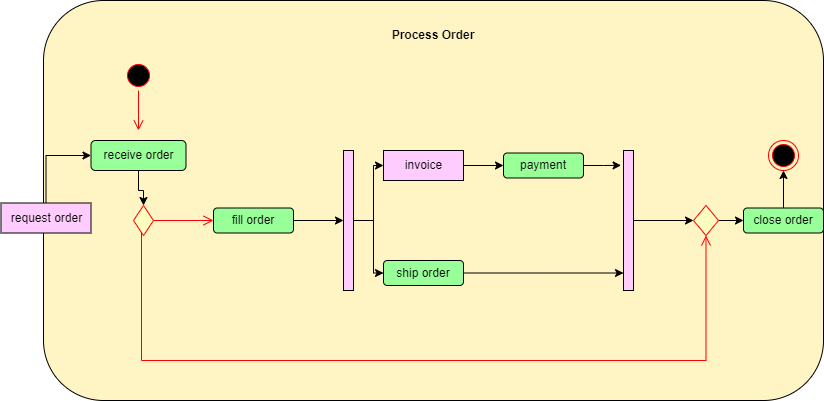
*Фиг.5*

Основните класове са:

* Администатор;
* Продавач;
* Клиент;
* Продукт;
* Количка;
* Поръчка;
* Сметка;

Допълнителни класове за статусът на клиента и поръчката (*фиг.5*).

* 1. **Процес на поръчка**

****

*Фиг.6*

Диаграмата (*фиг.6*) е процес на поръчка чрез попълване на лична информация.

1. **Функционални изисквания**

Системата поддържа следните видове потребители:

* Администратор – разполага с пълен достъп до всяка част от системата.
* Продавач – разполага с достъп до продуктите. Има разрешение да добавя, премахва или редактира продуктите. Според ситуацията може да има и времените права на администратора.
* Регистриран потребител – да създава поръчка, да се разплаща електронно, да достъпва историята с поръчки (ако има такава), да променя информация като адрес и разплащателни средства.
* Не-регистриран потребител – да разглежда продуктите и да се регистрира в системата.

Описание на данните на клиентите:

* Име, презиме, фамилия;
* Адрес;
* Идентификационен номер ID - вижда се само от адмнистратора и продавача;
* Потребителско име (username) и парола;
* Имейл;
* Мобилен телефон;
* Електронна карта;

Модулите в системата са:

* За вход/регистрация на клиента;
* Добавяне и премахване на продукти в количката;
* Редакция на продукти от продавача;
* Пълен контрол на системата от администратора;
* Създаване на поръчка с данни на клиента;
* Добавяне или премахване на платежни средства;
* Потвърждаване на поръчка;
* Изпращане на ревю;
* Верификация на клиент;
* За фактуриране;
* Запазване на история с поръчки;

*Бележка:* Системата ще има ограничение за данните на потребителите – личните данни да не са публични и при нарушаване на верификация*4* профилът на потребителя ще бъде временно спрян.

1. **Нефункционални изисквания**
   1. **Общи нефункционални изисквания:**

* Да бъде централизирана*5*;
* Да работи с бази данни;
* Да позволява многопотребителска работа;
* Да поддържа историята на данните и промените;
* Интерфейсът*6* да позволява бъдещо разширение и подобрение;
* Системата да позволява работа с различни браузъри;
* Системата да се използва както в уеб така и в мобилно приложение;
* Системата да осигурява непрекъснат 24/7 достъп;
* Системата да извършва обработка на потребителски заявки в нормално възприето време (2 секунди), дори и при голям брой едновременно работище с нея потребители;
* Да преизползва определени ресурси, за които това е допустимо;
* Да позволява лесно преконфигуриране на отделнинте модули при нарастване на потребителските заявки с цел запазване на заложената производителност;
* Да осигури достатъчно високо ниво на защита на личните данни на потребителите;
* Да поддръжа разширяване на каталози с продукти;
* Да поддържа адекватно ниво на самонаблюдение на процесите и функционирането си;
* Да бъде реализирана със стандартни технологии и да поддържа общоприети комуникационни стандарти;
* Да бъде разгърната върху съответните среди (тестове);
* Да бъде създаден административен интерфейс, чрез който може да бъде извършвана конфигурацията на софтуера;
* Всеки потребител да има уникален идентификационен номер в системата;
* Използване на HTTP/2*7*;

При подобряване на версии на използваните проекти се извършва анализ на влиянието върху настоящата система. В случаите, при които се оптимизира използвана функционалност, отстраняват се пропуски в сигурността, стабилността или бързодействието, новата версия се извлича и използва след успешното изпълнение на интеграционните тестове.

* 1. **Изграждане и поддръжка на множество среди**

|  |  |
| --- | --- |
| Среда | Описание |
| Development/  разработка | чрез средата за разработка се осигурява работата по разработката, усъвършенстването и развитието на системата. В тази среда са налични и допълнителните софтуерни системи и инсталации, необходими за управление на разработката – continuous integration*8* средства, системи за автоматизирано тестване и др. |
| Testing | чрез тестването се извършват тестове, преди разгръщане на нова версия от средата за разработка върху крайното представяне. В нея се извършват всички интеграционни тестове, както и тестовете за натоварване. Всички, които трябва да се интегрират към системата могат да тестват в нея интеграцията си, без да застрашават работата на продукцията. |
| Production | това е средата, която е публично достъпна за реална експлоатация и интеграция със съответните външни системи и услуги. |