**ПОИЗВОДСТВЕНА ПРАКТИКА**

**ДЕЙНОСТ: УЧЕНИЧЕСКИ ПРАКТИКИ**

***2023-2024***

**Тема: „Web-базирано приложение за логистика и транспортиране до адрес“**

**Курсов проект**

*Автори:*

*Йоанна Владимирова Симеонова, клас XI В*

БургасСъдържание

[1 Въведение 3](#_Toc96509792)

[2 Цели и обхват на софтуерното приложение 3](#_Toc96509793)

[3 Анализ на решението 3](#_Toc96509794)

[3.1 Потребителски изисквания и работен процес 3](#_Toc96509795)

[3.2 Примерен потребителски интерфейс 4](#_Toc96509796)

[3.3 Диаграми на анализа 4](#_Toc96509797)

[3.4 Модел на съдържанието / данните 5](#_Toc96509798)

[4 Дизайн 5](#_Toc96509799)

[4.1 Реализация на структура на приложението (3-layer), Разделение на кода според предназначението му 5](#_Toc96509800)

[4.2 Организация и код на заявките към база от данни 5](#_Toc96509801)

[4.3 Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб) 5](#_Toc96509802)

[5 Тестване 6](#_Toc96509803)

[6 Заключение и възможно бъдещо развитие 6](#_Toc96509804)

[7 Използвани литературни източници и Уеб сайтове 6](#_Toc96509805)

[8 Приложения 6](#_Toc96509806)

[9 Критерии и показатели за оценяване 8](#_Toc96509807)

# Въведение

Идеята на проекта е система за изчисляване на график за доставки от склад до адрес, като целта на приложението е да подреди доставките по оптимален начин, за пестене на ресурси като гориво. Проектът ще се ситуира в уебсайт.

Проблема, който се опитва да разреши програмата, е голямото разхищение на ресурси и замърсяване на природата осредством количеството парникови газове отделени от превозните средства, използвани за доставка. Според информация от проучвания, проведени от HIT, „между 20% и 30% от емисиите на въглероден диоксид са произведени от доставки на различни по вид пратки“ (Дейвис, 2020). Приложението ще се справя с този проблем, като предлага следните функционалности:

* За служител в магазин:
  + Форма за въвеждане на поръчка с продукти по EAN номер.
  + Изглед със завършена поръчка и резултат на дата и часови диапазон на доставката.
* За доставчик:
  + Изглед с дневен график за доставки.
  + Функционалност за отмятане на пратка като доставена до зададения адрес.
* Админ панел за евентуална редакция на поръчка.

# Цели и обхват на софтуерното приложение

Приложението е насочено към физически магазини, които предлагат опция за доставка до адрес. Служителите ще имат възможността да генерират поръчки, които запълват графика, създаден от програмата. Доставчиците ще могат да виждат своята програма за деня и в коя част на града трябва да доставят пратки.

Понеже всеки град е разделен на квартали, за целите на приложението ще се използва измислен град със зони. В един часови диапазон ще се поместват пратки, поръчани за една съща зона награда, като за следващия откъс от време могат да се доставят пратки до съседна зона. Така горивото, което се изпозва, намаля значително и ефективността се повишава.

Когато се съставя графикът, ще се използва методът на „първия заварил“, т.е. първата направен поръчка отива на първото свободно място, което удовлетворява целите на апликацията.

Като по-нататъчна цел обхватът на приложението може да се разшири и до фирми доставчици и онлайн магазини.

# Анализ на решението

## Потребителски изисквания и работен процес

### UML Use-Case диаграма

Основните участници в системата са:

* **Потребител:** Всеки, който използва системата.
* **Служител в магазин:** Може да създава поръчки.
* **Доставчик:** Може да вижда графика и да маркира поръчки като доставени.
* **Администратор:** Може да изпълнява CRUD операции върху поръчките.

Най-популярните случаи на използване системата за логистика са следните:

**User:**

* **Log in:** Влиза в системата с предварително зададени потребителско име и парола**.**
  + **Verify account:** Системата сверява достоверността на въведените данни**.**
    - **Display message:** Показва се съобщение за успешно влизане в системата.
    - **Create access token:** Създава се уникален ключ за сесията на текущия потребител.
  + **Display error message:** Изписва се съобщение с грешка при неправилно въведени данни.

**Retailer:**

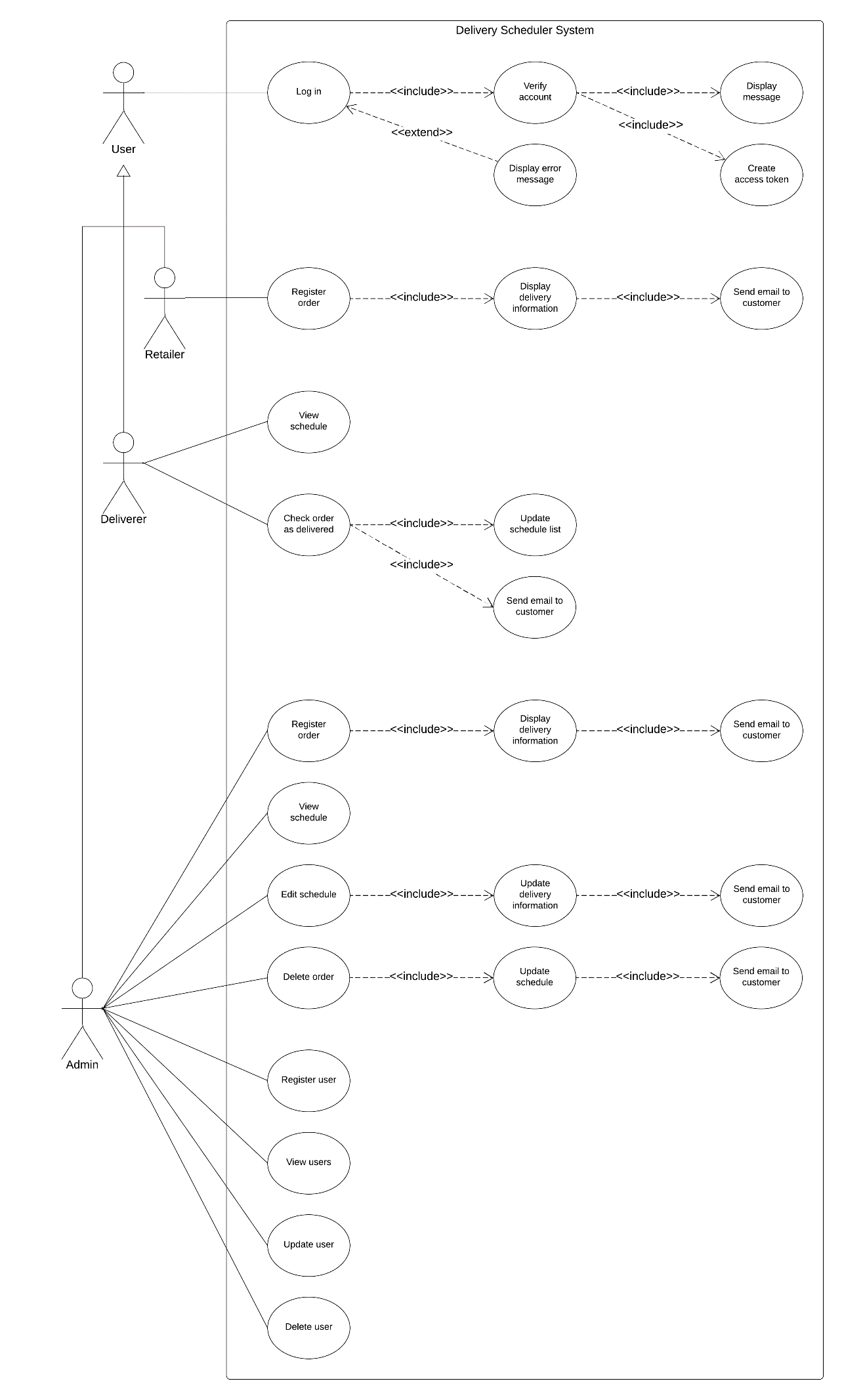
* **Register order:** Въвежда нова поръчка.
  + **Display delivery information:** След задаване на поръчката, системата изчислява къде в графика може да бъде поставена тя и връща дата и часови диапазон.
    - **Send email to customer:** Системата изпраща имейл за потвърждение на клиента.

**Deliverer:**

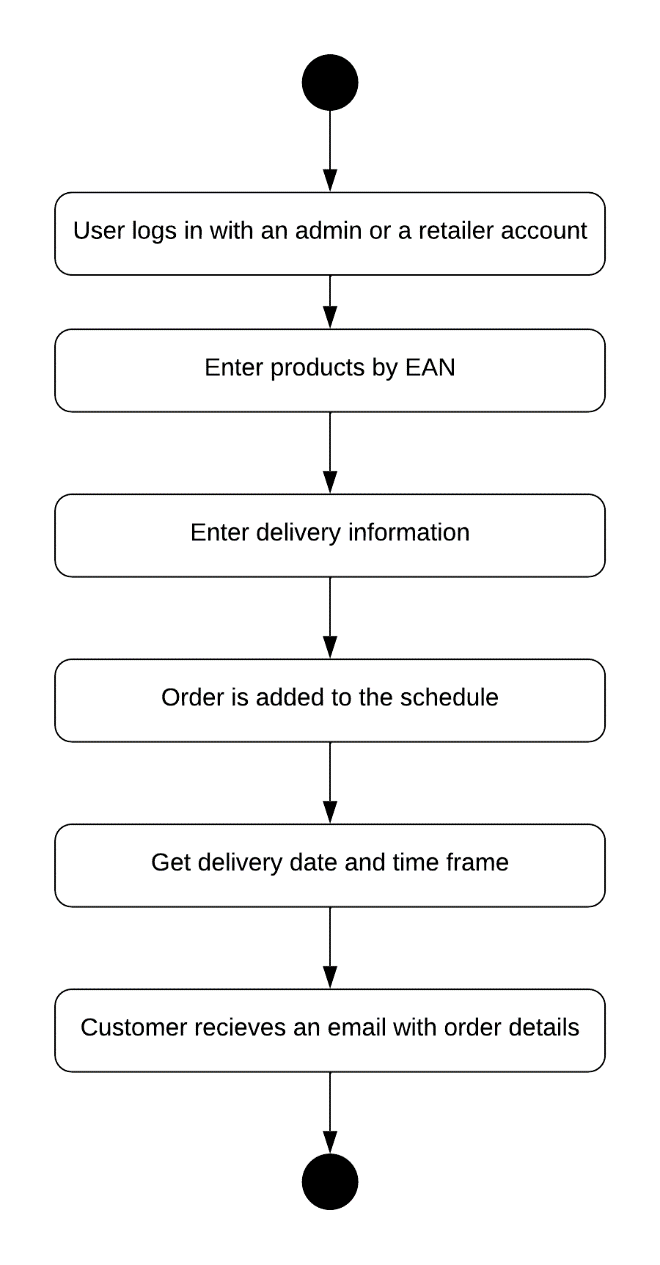
* **View schedule:** Доставчикът може да види дневната си програма за доставки.
* **Check order as delivered:** Задава поръчката като доставена.
  + **Update schedule list:** Премахва поръчката от графика на доставчика.
    - **Send email to customer:** Системата изпраща имейл за потвърждение на доставката на клиента.

**Admin:**

* **Register order:** Въвежда нова поръчка.
  + **Display delivery information:** След задаване на поръчката, системата изчислява къде в графика може да бъде поставена тя и връща дата и часови диапазон.
    - **Send email to customer:** Системата изпраща имейл за потвърждение на клиента.
* **View schedule:** Извежда целия график за доставки по дни.
* **Edit schedule:** Редактира местоположението на поръчка в графика.
  + **Update delivery information:** Обновява данните за поръчката.
    - **Send email to customer:** Системата изпраща имейл с данните за редакцията на клиента.
* **Delete order:** Изтрива поръчка.
  + **Update schedule:** Обновява графика.
    - **Send email to customer:** Системата уведомява клиента за изтриването на поръчката с имейл.
* **Register user:** Регистрира нов потребител.
* **View users:** Вижда списък с всички потребители.
* **Update user:** Редактира потребител.
* **Delete user:** Изтрива потребител.



### UML Activity диаграма

**Създаване на поръчка:** Само администратори и служители на магазина могат да изпълняват тази дейност. Стъпките, които трябва да изпълнят са следните:

## Примерен потребителски интерфейс

Допълнете резултатите от статистически анализа на проблема, описани в секция 3.1, с фигури на примерен графичен интерфейс /създадени или в самата среда заедно с потоците от събития, или извън нея/.

## Диаграми на анализа

Тук опишете резултата от анализа на проблема с UML диаграми

* ER диаграма на базата данни
* използване на клас диаграми на анализа /с класове със стереотипи/ За по-сложните контролни класове представете диаграми на състоянието /евентуално йерархични/
* диаграми на последователността и на комуникацията

*Забележки:*

1. *В зависимост от спецификата на проекта трябва да бъдат разработени съответно необходимите диаграми.*

## Модел на съдържанието / данните

Тук опишете модела на данните/съдържанието - текстово, графично и евентуално аудио/видео съдържание), което ще представите в проекта си. Опишете размера и типа на данните/файловете и начина на кодиране за всеки от ресурсите.

Ако ползвате в проекта си текстово и/или мултимедийно съдържание от различни типове, представете неговата структура, напр. посредством таксономия, типология, онтология или други схеми за представяне на структурата от категории, под-категории, типове и т.н., както и техните взаимовръзки с други категории или типове, напр. географски региони и дялове, области/сегменти от промишлеността, и др.

След структурата опишете и възможните характеристики, атрибути и честота на срещане на всеки един ресурс в съдържанието (категория, тип, екземпляр, връзка/релация и т.н.).

# Дизайн

Тази секция представя дизайна на решението на проблема за проекта ви. Опишете каква софтуерна платформа сте избрали за вашето решение /напр. .NET, java/. Представете схема на софтуерната архитектура на решението /по модули и/или слоеве/ с диаграма на разгръщането, както и диаграми на класовете на дизайна /с ограничения, описани на OCL/, диаграми на времето /за задаване на времена за синхронизация и комуникация в решението/ и компонентни диаграми. Илюстрирайте решението с извадки от генериран сорс код.

## Реализация на структура на приложението (3-layer), Разделение на кода според предназначението му

Описание на слоевете, предназначението им, библиотеки и методи включени в съответния слой.

## Организация и код на заявките към база от данни

Описание на инструментариума за достъп до базата данни от гледна точка на програмния код. Описание на методите за извличане, добавяне и изтриване на обекти в базата данни.

## Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб)

Описание на основните функционалности на интерфейса на приложението.

*Забележка: Няма формално изискване на определен брой диаграми от даден вид, за даден брой проектанти.*

# Тестване

Тук се *включват тестовите случаи* и какви видове тестване предвиждате в реалното изпълнение на проекта, напр. с колко и какви документи, в какви браузъри, с какви приставки, и т.н.

# Заключение и възможно бъдещо развитие

В заключение, обобщете резултатите от работата ви по проекта, както и предимствата и ограничеността на използваните технологии / езици / методи. Укажете какви алтернативи могат да се използват и техните предимства и недостатъци. Опишете каква е използваемостта на подобни решения в практиката и какво бихте предложили като насоки за бъдещо развитие на вашето решение.

# Използвани литературни източници и Уеб сайтове

Използвайте вградената функционалност на Word: References > Citations & Bibliography

1. Уеб сайт на ….., адрес ….
2. Уеб сайт на ….., адрес ….
3. Уеб сайт на ….., адрес ….
4. Уеб сайт на ….., адрес ….
5. Литературен източник 2
6. Литературен източник 3
7. Литературен източник 4
8. Литературен източник 5
9. Уеб сайт на Европейски съюз, адрес: <https://projects.research-and-innovation.ec.europa.eu/en/horizon-magazine/carbon-cost-home-delivery-and-how-avoid-it>

# Приложения

При необходимост можете да добавите и допълнителни секции под формата на апендикси. Таблица с диаграми, таблици и графики

*Забележка:*

1. *Документацията на проекта се предава само в електронен вид в MS Word, чрез качването на архив с документа и останалите файлове по проекта, в задание за предаване на проект, в канала на екипа в Teams.*
2. *Кода на проекта, базата данни и документацията трябва да са налични в репозитори в GitHub, което е копие на заданието генерирано в организацията.*

# Критерии и показатели за оценяване

|  |  |  |
| --- | --- | --- |
| **Показател** | **точки** | **срок** |
| 2. [Цели и обхват на софтуерното приложение](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Цели_и_обхват)  3.1 [Потребителски изисквания и работен процес](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Потребителски_изисквания_и) | 5  5 | 15.03.2024 |
| 3.2 [Примерен потребителски интерфейс](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Примерен_потребителски_интерфейс)  3.3 [Диаграми на анализа](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Диаграми_на_анализа)  3.4 [Модел на съдържанието/данните](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Модел_на_съдържанието) | 5  5  5 | 29.03.2024 |
| 4.1 [Реализация на структура на приложението (3-layer),  Разделение на кода според предназначението му.  Допълване на Class диаграми/3.3/](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 10  10 | 19.04.2024 |
| 4.2 [Организация и код на заявките към база от данни](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 15 | 10.05.2024 |
| 4.3 [Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб)](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 10 | 31.05.2024 |
| 5. [Наличие и организация на автоматизирани тестове](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Тестване). | 15 | 15.06.2024 |
| 6. Организация на проекта в система за контрол на изходния код и употреба на добри практики (merge requests, code reviews, branching strategy) | 10 | 25.06.2024 |
| 1. [Въведение. Ниво на завършеност на проекта](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Въведение) |  | 30.06.2024 |
| Документация на проекта (XML comments, wiki, etc.) | 5 | текущо |
| Презентация на проекта |  | 30.06.2022 |
| Общо | 100 | Финал на първа фаза |