**Анотація:** в курсовій роботі розроблений сервіс для тренування пам’яті «Memory.pro». Для розробки програмного забезпечення було використано мову програмування Java, бібліотеку JavaFX (з використанням FXML та SASS), JPA фреймворк EclipseLink та СУБД MYSQL.

**Аннотация:** в курсовой работе разработан сервис для тренировки памяти «Memory.pro». Для разработки программного продукта был использован язык программирования Java, библиотека JavaFX (с использование FXML и SASS), JPA фреймворк EclipseLink и СУБД MYSQL.

**Annotation:** a software product for memory training named «Memory.pro» is developed in this coursework. Java programing language, library JavaFX (with FXML and SASS), JPA framework EclipseLink and DBMS MYSQL have been used in the coursework.

**Вступ**

Автоматизація ведення обліку підприємств є поширеним явищем сьогодні. У даній курсовій роботі розроблюється програмне забезпечення для автоматизації ведення обліку та продажу пального на АЗС, автоматизації роботи програми лояльності клієнтів АЗС.

Метою роботи є змоделювати базу даних, що будет відтворювати структуру модулів АЗС, які відповідають за облік та реалізацію пального на АЗС та роботу програми лояльності АЗС. Розробити програмне забезпечення, що буде автоматиувати облік та реалізацію пального на АЗС, роботу програми лояльністі АЗС.

Віднопідно для досягнення мети необіхдно вирішити наступні задачі:

* Провести аналіз предметної області та вже існуючих аналогів програмного забезпечення.
* Розробити проект програмного забезпечення, який містить в собі:

1. Розробку, концептуальної, логічної, фізичнї моделі бази даних.
2. Розробку специфікації програмних модулів.
3. Опис алгоритмів програми та розробку інтерфейса користувача.
4. Опис СУБД та засобів програмної реалізації.
5. Аналіз та опис предметної області
   1. **Дослідження та опис предметної област:**

Автозаправні станції (АЗС) призначені для заправки транспортних засобів нафтопродуктами. Головним призначенням АЗС є заправка пальним автомототранспорту. На окремих АЗС організовується продаж масел, консистентних мастил, запасних частин, аксесуарів до автомобілів та інших транспортних засобів, приймання від власників індивідуального транспорту відпрацьованих масел і дрібної тари з-під нефтепродуктів, а також технічне обслуговування та мийку автомобілів.

АЗС поділяються на стаціонарні, пересувні (ПАЗС), контейнерні (КАЗС і блок-пункти). Стаціонарні АЗС розташовуються в містах та населених пунктах, а також на автодорогах. Модульні АЗС характеризується надземним розташуванням резервуарів та рознесенням ТРК і контейнера зберігання пального. У дані роботі будемо розробляти підсистему для автоматизації роботи стаціонарної АЗС.

Зберігання нафтопродуктів на АЗС здійснюється в металевих горизонтальних надземних або заглиблених резервуарах.

Найбільш поширені АЗС заправляють автотранспорт традиційними сортами вуглеводневого пального - бензином і дизельним паливом (бензозаправні станції). Менш поширеними є Автомобільна Газонаповнювальна Компресорна станція (АГНКС) - заправка стисненим природним газом (CNG) і Автомобільна Газозаправна Станція (АГЗС) - заправка скрапленим нафтовим газом (LPG). Є також кілька типів водневої заправної станції.

* 1. **Аналіз вже існуючого програмного забезпечення**

На сьогоднішній день існує достатньо програмних продуктів, що дозволяють автоматизувати роботу АЗС, більшість з яких націлені на автоматизацію системи, у той час коли програмний продукт, що розробляється у межах даної роботи має на меті автоматизувати оберт пального, продаж пального та роботу програм лояльності АЗС. Основним недоліком вже існуючого ПО є його складність в цілому, так як він охоплю роботу усіх систем АЗС, а також висока вартість програмних продукта.

* 1. **Постановка задачі**

Необхідно розробити інформаційну систему для автоматизації роботи менеджера по роботі з постачальниками та оператора стаціонарних автозаправних комплексів.

Менеджер по роботі з постачальниками відповідальний за внесення даних про пальне у БД системи та роботу з формами звітності. Вхідні дані він будет отримувати з прихідних накладних та контрактів з постачальниками. Менеджер по роботі з постачальниками буде стоворювати наступні форми звітності:

* Звіт про залишок пального;
* Звіт про оберт (переміщення) пального за певний період;
* Звіт про постачальників;
* Видаткова накладна.

Оператор АЗС відповідальний за створення облікових записів програми лояльності, роботу з формами звітності програми лояльності. Дані для створення облкового запису програми лояльності оператор АЗС будет отримувати безпосередньо від клієнтів АЗС. У випадку успішного створення облікового запису у програмі лояльності АЗС, клієнт отримуватиме картку програми лояльності, що надає право на знижку при покупці пального відповідно до умов програми лояльності. Відповідно оператор АЗС будет створюват наступні форми звітності:

* Звіт про клієнтів, що є учасниками програми лояльності;
* Звіт по карткам програми лояльності.

Підсистема, що відповідає за ведення обліку пального на автозаправному комплексі характеризує пальне наступними параметрами:

1. унікальний номер (ідентифікатор) пального;
2. тип пального;
3. виробник;
4. дата поставки пального;
5. загальний об’єм завезеного пального;
6. наявний об’єм пального;
7. ціна поставки (за 1л);

Відповідно для отримання повної інформації про постачальника, дані про постачальника слід деталізувати, винесши їх до окремої таблиці, що буде містити наступну информацію:

1. унікальний номер (ідентифікатор) постачальника;
2. назва компанії;
3. контактний телефон;
4. контракт;
5. дата укладення;
6. дата закінчення дії контракту;
7. контана особа;
8. адреса.

Автозаправний комплекс має передбачати можливість роботи програми лояльності для своїх кліентів. Для роботи програми лояльності необхідні наступні дані про клієнта:

1. унікальний номер (ідентификатор) клієнта;
2. прізвище, ім’я, по-батькові;
3. дата народження;
4. контактний телефон;
5. e-mail;
6. номер карти програми лояльності;
7. дата видачі;
8. марка автомобіля;
9. тип пального;
10. загальний об’єм придбаного пального;
11. відсоток знижи.

Підсистема, що відповідає за облік пального на АЗС має давати змогу створювати видаткові накладні, які у даній системі будуть відповідати лише за рух пального між різними АЗС. Відповідно для такої накладної необхідно мати наступні дані про отримувача:

1. унікальний номер (ідентифікатор) АЗС;
2. адреса АЗС;
3. контактний телефон АЗС;
4. контактна особа;
5. тип пального, що найдійшо на АЗС;
6. загальний об’єм пального;
7. об’єм проданого пального;
8. дата поставки;
9. ціна поставки (за 1л).

Для коректної роботи системи слід передбачити наступні умови:

1. Учасником програми лояльності може бути особа немолодше 18 років.
2. Клієнт автозаправного комплексу може мати лише один обліковий запис у програмі лояльності.

З даною інформаційною системою будуть працювати наступні групи користувачів:

* Манеджер по роботі з постачальниками;
* Оператор АЗС.

При роботі з підсистемою, що відповідає за ведення обліку пального на автозаправному комплексі менеджер по роботі з постачальниками повинен мати можливість виконувати наступні задачі:

1. Вносити дані про пальне.
2. Корегувати дані про пальне.
3. Видаляти записи про пальне для помилкових записів;
4. Виконувати операцію пошуку за певним критерієм (атрибутом) запису;
5. Переглядати усі записи;
6. Формувати звітність;
7. Вести облік пального на автозаправному комплексі.

При роботі з підсистемою, що відповідає за ведення програми лояльності кліентів оператор АЗС повинен мати можливість виконувати наступні задачі:

1. Вносити дані про кліента;
2. Корегувати дані про кліета;
3. Перевіряти актуальність клієнта;
4. Виконувати операцію пошуку за певним критерієм (атрибутом) запису;
5. Переглядати усі записи;
6. Формувати звітність.
7. Продаж пального та формування чека.

Відповідно система в цілому має виконувати наступні функції:

1. Створення запису;
2. Вибір певного запису;
3. Видалення запису;
4. Перегляд записів;
5. Пошук записів;
6. Робота з формами звітності.

Розглядаючи підсистему, що відповідає за ведення обліку пального на автозаправному комплексі виділимо наступні типа даних, які будуть характеризувати пальне:

1. Ціле число для унікального ідентифікатора (номера) пального;
2. Дійсне число для загального об’єму завезеного пального, наявного об’єму пального, ціни закупівлі.
3. Рядкове представлення для типу пального, виробника.
4. Дата для дати поставки пального.

Розглядаючи підсистему, що відповідає за ведення обліку пального на автозаправному комплексі виділимо наступні типа даних, які будуть характеризувати постачальника:

1. Ціле число для уникального ідентифікатора (номера) пального;
2. Рядкове представлення для назві компанії, адреси, контактного телефону, контактної особи, контракту.
3. Дата для дати укладення контракту та дати закінчення дії контракту.

Розглядаючи підсистему, що відповідає за ведення обліку пального на автозаправному комплексі виділимо наступні типа даних, які будуть характеризувати отримувача, для якого буде створено видаткову накладну:

1. Ціле число для унікального ідентифікатора (номера) АЗС.
2. Рядкове представлення для адреси АЗС, контактного телефону, контактної особи, типу пального.
3. Дійсне число для загального об’єм пального отриманого пального, обє’м проданого пального, ціни продажу;
4. Дата для дати поставки.

Розглядаючи підсистему, що відповідає за ведення обліку клієнтів, які беруть участь у програмі лояльності на автозаправному комплексі, виділимо наступні типа даних, які будуть характеризувати клієнта:

1. Ціле число для номера картки програми лояльності, унікального ідетифікатора клієнта;
2. Дійсне число для відсотка знижки;
3. Рядкове представлення для ПІБ, контактного телефону, e-mail, марки автомобіля, типу пального.
4. Дата для дати народження.

На основі отриманих даних, які зберігаються та оброблюються, програмне заберзпечення буде формувати такую документацію як звіти, видаткова накладна, формування заявки.

Вхідними документами для системи буде прихідна накладна, контракт з постачальником, анкета кліента.

* 1. **Вибір засобів розробки**

Для розробки програмного забезпечення у цій роботи ми будемо викорстовувати мову програмування Java. Для розробки графічного інтерфейсу користувача використаємо бібліотеку JavaFX, яка задоволняя основні вимоги до інтерфейсу. Для роботи з базами даних використаємо СУБД MariaDB, а для безспосередньої роботи з програмних модулів – фреймворк Persistence API.