**SMP**

|  |  |  |  |
| --- | --- | --- | --- |
| **Начало мероприятия** | **Окончания мероприятия** | **Длительность** | **Мероприятие** |
| 9:50:00 | 10:00:00 | 00:10:00 | Регистрация участников |
| 10:00:00 | 10:20:00 | 00:20:00 | Инструктаж участников по охране труда и технике безопасности, правилам проведения демонстрационного экзамена |
| 10:20:00 | 10:25:00 | 00:05:00 | Распределение рабочих мест (жеребьевка) |
| 10:25:00 | 11:00:00 | 00:35:00 | Ознакомление с рабочими местами |
| 11:00:00 | 11:15:00 | 00:15:00 | Ознакомление с заданием |
| 11:15:00 | 12:45:00 | 01:30:00 | Выполнение задания |
| 12:45:00 | 13:00:00 | 00:15:00 | Перерыв |
| 13:00:00 | 14:30:00 | 01:30:00 | Выполнение задания |
| 14:30:00 | 14:45:00 | 00:15:00 | Перерыв |
| 14:45:00 | 15:45:00 | 01:00:00 | Выполнение задания |
| 15:45:00 | 20:30:00 | 04:45:00 | Проверка работ |

**Структура задания**

|  |  |  |
| --- | --- | --- |
| **Номер сессии** | **Описание задания** | **Время выполнения** |
| Сессия 1 | ***Разработка баз данных, объектов баз данных и импорт***  Реализация базы данных в выбранной СУБД: создание таблиц, связей между ними, полей в таблицах на основании ERD или при помощи скрипта. Приведение исходных файлов данных к виду, подходящему для импорта. Импорт исходных данных разного формата. Создание объектов базы данных, таких как представления (Views), триггеры (Triggers), хранимые процедуры (Stored Procedures), функции (User Defined Functions) или аналоги | 00:30:00 |
| ***Разработка desktop-приложений***  Создание настольного приложения: окон, таблиц, списков, форм для заполнения, работа с базой данных, работа с изображениями. Разработка библиотеки классов. | 03:00:00 |
| ***Тестирование***  Разработка тест-кейсов, тестовых наборов, модульных тестов, реализация интеграционного тестирования. | 00:00:00 |

# **Стиль гид**

Шрифт - Serifiqo 4F Free Capitals

Цвет основной - #FFFFFF

Цвет для дополнительного фона - #FEFFB7

Цвет для кнопок - #03DAC5

Цвет для акцента - #001AFF

**Сессия 1**

|  |  |  |
| --- | --- | --- |
| **Наименование репозитория** | **Наименование ветки** | **Раздел задания** |
| Desktop | S\_1 | Основное приложение |
| Library | S\_1 | Библиотека классов |
| Unit-tests | S\_1 | Модульные тесты |

**Восстановление базы данных из скрипта**

Для восстановления таблиц в созданную базы данных воспользуйтесь предоставленным скриптом. В процессе разработки приложения **Вы можете изменять базу данных** на свое усмотрение.

**Импорт данных**

Заказчик системы предоставил файлы с данными для переноса в новую систему. Подготовьте данные файлов для импорта и загрузите в разработанную базу данных.

**Разработка Desktop-приложения**

При запуске приложения окно входа – первое, что видит пользователь (рисунок 1). На ней пользователю предлагается ввести свой логин и пароль. При вводе пароль пользователя должен быть скрыт. При этом необходимо предусмотреть возможность просмотра пароля при наведении на него.

После первой неуспешной попытки входа, система выдает сообщение о неуспешной авторизации, сбрасывает пароль пользователя, а затем, помимо ввода пароля, просит ввести captcha, состоящую из 6 символов (цифры и буквы латинского алфавита) и графического шума.

После неудачной попытки авторизации с вводом captcha, система блокирует возможность входа на 10 секунд и сбрасывает пароль пользователя.

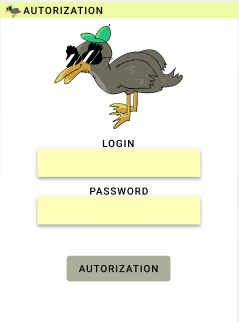


Рисунок 1 - Окно «Авторизация»

При успешном входе появляется всплывающее сообщение на 5 секунд о успешной авторизации (рисунок 2), которое содержит в себе: фотографию, роль и ФИО.

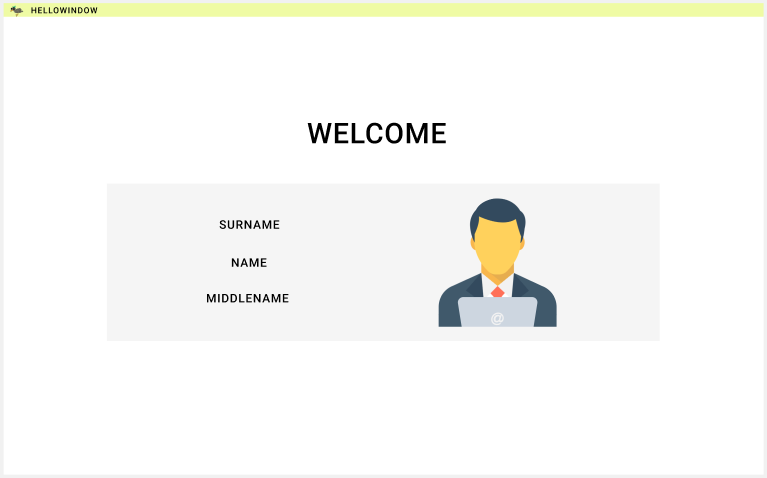


Рисунок 2 - Окно «Приветствия»

В базе данных пароль сохранен в зашифрованном виде с использованием rc2 метода шифрования (с использованием ключа шифрования «ILoveU»), для удобства снизу предоставлена таблица, в которой присутствует незашифрованный пароль и роль к которой она относится.

|  |  |  |  |
| --- | --- | --- | --- |
| **Логин** | **Пароль** | **Пароль (RC2)** | **Роль** |
| KorolevMM | Korolev | 9sJdtdFB6yA= | Менеджер |
| ZiminaMM | Zimina | F8OT1bvkuvM= | Кладовщик |
| SemyonovaVA | Semyonova | npslAobwdN30gQtvT2VuBg== | Продавец |

Если пользователь вошел под ролью «Продавец» то перед ним открывается окно (рисунок 3), которое содержит в себе следующие данные:

* статус пользователя (роль, ФИО, продолжительность сеанса, изменяется в реальном времени каждые 5 секунд);
* ассортимент товара магазина и их фактическое наличие в магазине (цифрой указывается количество на складе при магазине);
* поиск по артикулу товара, наименование товара, характеристикам товара;
* фильтрация по категориям товара;
* просмотр информации о товаре, с указанием наличия товара на складах и возможности оформления товара со склада.
* Корзину с товарами с указанием с какого склада будет доставлен товар с возможность удаления товара из корзины.

Как только покупатель определится с товаром он может его приобрести для этого ему необходимо нажать на кнопку «PAY» и перед ним откроется окно продажи (рисунок 4).

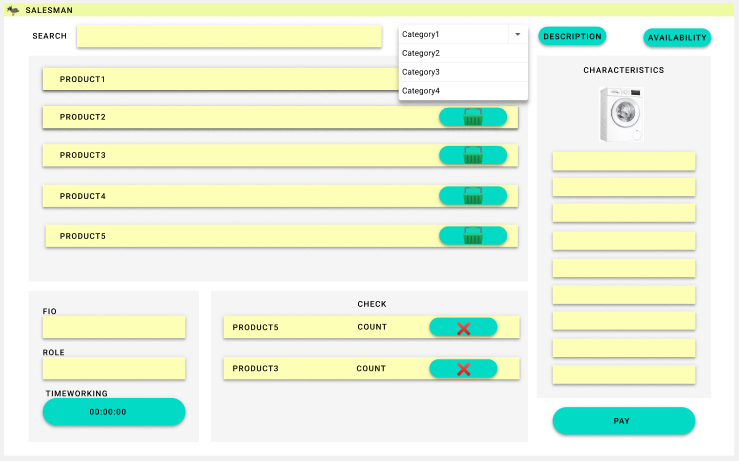


Рисунок 3 - Окно «Продавец»

В нем он может выбрать способ оплаты, а также ввести бонусную карту клиента.

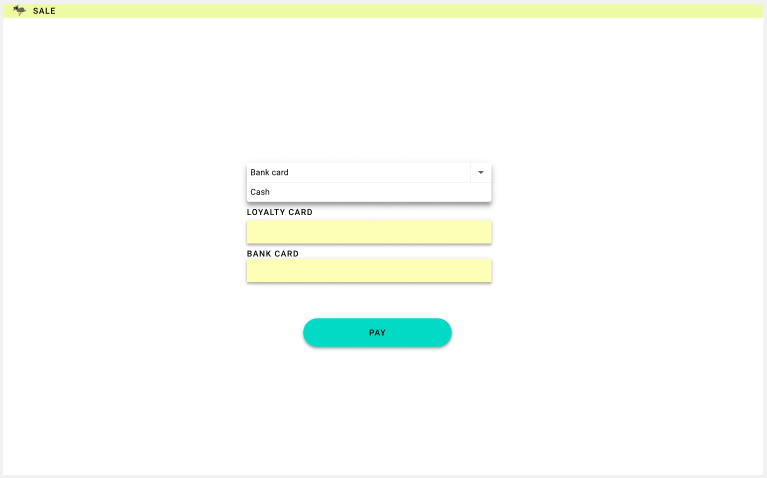


Рисунок 4 - Окно «Продажи»

После успешного ввода данных должен формироваться чек, с возможность сохранения в pdf (рисунок 5).

Если пользователь вошел под ролью «Менеджер» то перед ним открывается окно менеджера (рисунок 6), в нем отображается весь ассортимент товара и количество товара которое присутствуют на складе. Изменить склад менеджер может в выпадающем списке.

Для данной таблицы будет использоваться следующее цветовое выделение:

Серым - нет товара, подпись нет в наличии.

Красным - мало товара, количество товара от 1 до 3, подпись указывается количество товара на складе.

Желтым – умеренное количество товара, количество товара от 4 до 7, подпись указывается количество товара на складе.

Зеленый – достаточное количество товара от 8 до 10, подпись указывается количество товара на складе; если товара больше 10, то подпись меняется на много.

При нажатии на кнопку CSV, делается выгрузка товара, которое находится на складе, в формате csv.

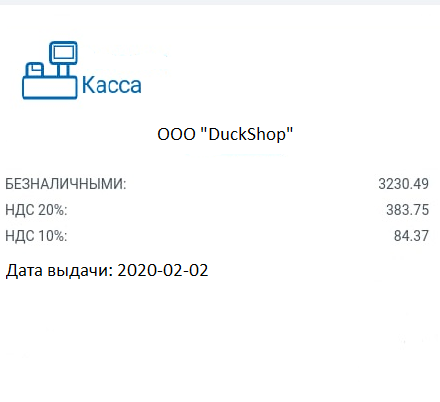


Рисунок 5 - Окно «Чек»

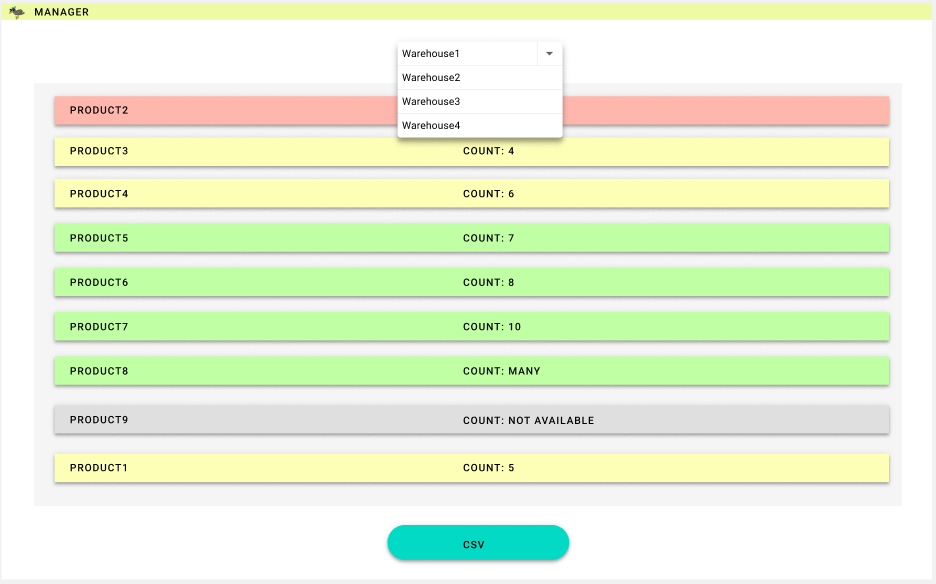


Рисунок 6 - Окно «Менеджер»

Если пользователь вошел под ролью «Кладовщик» то перед ним открывается окно кладовщика (рисунок 7), в нем отображается весь ассортимент товара, который не присутствует ни на одном складе. При нажатии на кнопку «FORM AN APPLICATION» должна формироваться заявка на закупку товара, который был выбран из списка отсутствующих товаров, т.е. формироваться запись в таблицах «Application» (id(autoincrement, datetime, user), ApplicationProduct (id(autoincrement,application,product).

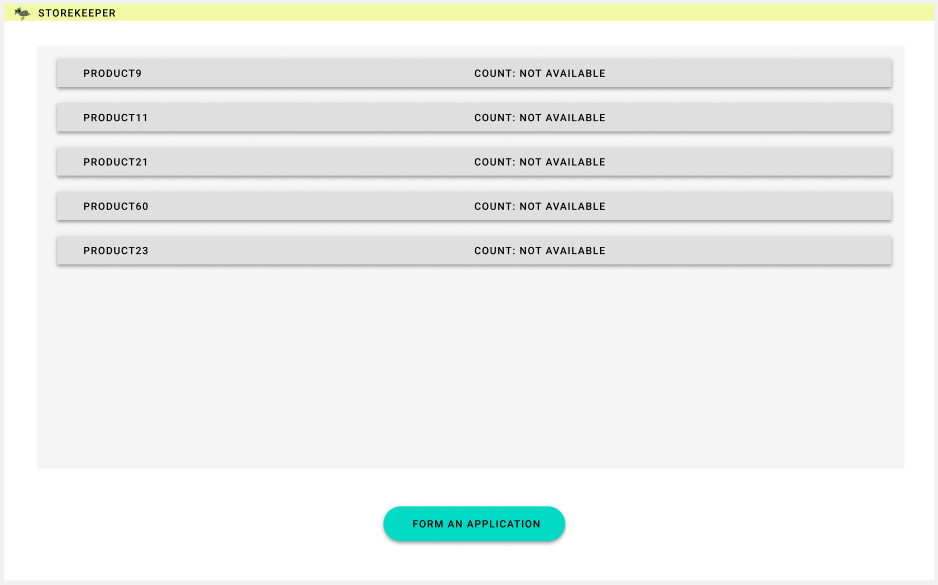


Рисунок 7 - Окно «Кладовщик»

**Разработка библиотеки классов**

**Класс аналитики**

Вам необходимо создать класс с названием Analytics, который будет позволять проводить аналитику различных процессов нашей компании.

Реализуйте метод, который принимает в себя список объектов даты и времени по совершенным покупкам, типа оплаты, номер бонусных карт, суммы покупки, а возвращает список месяцев, отсортированный в порядке уменьшения частоты заказов.

Например, вам поступили следующие данные: 2021-12-12 14:43, 2019-12-01 15:05, 2019-11-04 09:01, а, значит, самый популярный месяц - декабрь. Вам необходимо вернуть следующие данные: декабрь, ноябрь. В случае совпадения характеристик популярности сперва нужно вывести более ранние месяцы.

Для проверки работы метода вам предоставлены данные по продажам за 2020 год. В ресурсах вам предлагается файл (sale(for library).xlsx) для тестирование метода библиотеки.

**Разработка модульных тестов**

Для выполнения процедуры тестирования созданного вами метода библиотеки CompanyCoreLib,возвращающего упорядоченный список популярных месяцев, вам необходимо создать отдельный проект модульных тестов.

В рамках проекта разработайте тесты, максимально полно покрывающие функционал метода. Ничего страшного, если ваш метод работает не совсем идеально и тесты могут быть не пройдены в связи с этим в данном модуле это не так важно.

Обратите внимание, что имена тестов должны отражать их суть, т.е. вместо TestMethod1() тест следует назвать, например, PopularMonths\_NullList() для тестирования случая передачи пустой коллекции дат.

Необходимо разработать 5 модульные тестов.