

Система для учета магазина - shopmate (рабочее название)

15/04/2015

1 Описание проекта

Система shopmate позволяет вести оперативную деятельность (продуктового) магазина, начиная с оформления поставки и заканчивая фактической продажей. Shopmate делится на модули, каждый из которых может работать автономно, но раскрывает весь свой потенциал только при взаимодействии с другими модулями.

2 Технические уточнения

2.1 Термины

- Блок - некий визуальный элемент, выделяющийся либо графически (в виде рамок, очертаний), либо по смыслу (совокупность похожих элементов)
- Компонент - часть содержания, имеющего закрытое визуальное представление. Одна страница сайта может состоять из нескольких компонентов.
- Фронтэнд - для пользователя видимая оболочка веб-страницы
- Бэкэнд - невидимые для пользователя математические алгоритмы
- CMS - все работы происходят на основе системы управления содержанием - CMS 1C Bitrix (1С Битрикс)
- Модуль - является описанием общего функционала, который не может быть классифицирован как привязанный к определенной странице. Он может встречаться на любой странице в любом месте. Модуль может состоять из нескольких компонентов. Также модуль может содержать в себе части логики фронтэнда и бэкэнда.
- Хэдер - верхняя часть сайта, обладающая определенной структурой, которая видна сквозняком на всех или почти всех страницах сайта. Также используется обозначение "шапка".
- Футер - нижняя часть сайта. Функционал аналогичен хэдеру. Также используется обозначение "подвал".
- Тетрадка - рабочее и финальное название для кабинета салона красоты.
- Зеркало - рабочее название для сайта, который доступен клиенту (позже этот сайт будет по-умолчанию доступен по адресу getbam.ru)

2.2 Технические требования к сайту

Сайт должен быть адаптивным, поддерживать IE 8+ и использовать технологию композита. Тут ещё больше технического бла-бла.

3 Модули

Каждый модуль должен обладать индивидуальными правами доступа.

3.1 Товары/Склад

Данный модуль позволяет заносить новые товары, редактировать существующие, удалять (т.е. стандартные CRUD действия). Занесение и просмотр товаров вынесены отдельно, так как подразумевают специфичную логику. Например занесение должно в более поздних релизах работать через считывание штрих-кодов. Для хранения данных о товаре должен использоваться стандартный компонент товаров и складской системы Битрикс-магазина для того, что бы в любой момент можно было к системе shopmate подключить интернет-продажу (тем более оффлайн-продажа отличается только одной опцией от интернет-магазина - оплатой наличными). Важно разделять занесение нового товара (данная часть программы представляет собой справочник) и оприходывание товара (поступает новая партия товара). Т.е. если при занесении товара я ввожу его базовые характеристики (такие как название например), то при оприходывании я заносу параметры конкретной партии (поставщик, закупочную цену и т.д.).

3.1.1 Занесение новых товаров

Данная страница позволяет занести новый товар в виде справочника (т.е. не поступившую партию, а именно товар как сущность). Пользователь заносит товар обычно один раз и очень редко редактирует. Параметры товара:

- Название товара
- Штрихкод (у одного и того же товара может быть несколько штрихкодов, для этого предусмотрено добавление дополнительного штрихкода)
- Единица измерения
- Описание товара

3.1.2 Оприходывание новых товаров

3.1.2.1 Релиз Альфа Если приходит новая партия товара, то пользователь заносит этот товар через накладную, одна накладная может содержать одну или несколько позиций товаров с определенными параметрами. Накладных может быть неограниченно количество и создаваться они будут часто.

Первично пользователь должен внести название накладной (добавляем маску, что бы он понимал как заносить данные) и дату (по умолчанию подгружаем текущую дату, но также

даем возможность поменять (опять же, с маской)). Далее пользователь выбирает поставщика (текстовое поле с саджестом). После этого в каждой строке он может занести одну партию товара. Каждая строка состоит из определенных параметров товара.

Параметры товара:

- Штрихкод (можно занести вручную, а также с помощью сканнера). Если человек занес штрихкод, то базовые параметры (название) должны подгружаться автоматически. Если он занес штрихкод, но система не нашла такого товара, то должна предложить ему создать новый товар (открываем форму в попапе).
- Название - хранятся в справочнике (см. пункт 3.1.1), при занесении должен работать саджест (или подгружаем уже по штрихкоду)
- Кол-во по накладной (должно соответствовать в точности бумажному документу)
- Кол-во по факту (если пришло меньше (или больше), то нельзя это фиксировать в поле "кол-во по накладной нужно обязательно отдельно вынести)
- Цена на продажу (см. отдельный пункт ниже)
- Закупочная цена
- Дату истечения срока годности, если товар скоропортящийся

При занесении цены у пользователя должен быть следующий выбор:

- Использовать старую цену (если такой товар уже есть) - стандартный функционал.
- Ввести новую цену и обновить эту цену для всех уже существующих товаров. При этом новую цену можно ввести в виде двух параметров (на выбор) - в качестве наценки процентов на закупочную цену или в свободной форме

При этом дефолтно просто загружается текущая цена. Но также у пользователя есть возможность ввести новую цену в абсолютном значении или в процентуальном прибавлении/убавлении. Для сохранения первого сценария нужно просто оставить поле с дефолтным значением и ничего не делать.

Для изменения цены пользователь может сразу в обзоре позиций накладной занести абсолютное значение, либо нажав на "+" открыть всплывающее окно, которое дает несколько вариантов на выбор:

- Оставить все как есть (если пользователь нажал по ошибке)
- Накинуть процент - вводим значение процента...
- ...и выбираем к чему накинуть процент:
- на закупочную цену текущий поставки
- на самую дорогую закупку
- на среднезакупочную цену

Для каждого товара также необходимо предусмотреть индивидуальные прайслисты клиента. Т.е. индивидуальный прайслист всегда приоритетнее всех остальных ценовых настроек.

3.1.2.2 Релиз Бета Необходимо подключить функционал считывание штрих-кодов, а также занесения товаров при принятии товаров от поставщика.

Тут очень важно ещё уточнить как это в действительности происходит. У складовщика скорее всего будет просто считыватель штрих-кодов? Также скорее всего он должен будет просто принять данные (считать коды), а потом у себя на компьютере увидеть список товаров, которые он принял и начать их обработку

3.1.2.3 Релиз Гамма

3.1.3 Просмотр товаров

Просмотр товара осуществляется в стандартном табличном виде с делением на категории и подкатегории. При нажатии на товар можно просмотреть его детальные параметры, а также удалить товар.

3.1.3.1 Скоропортящийся товар Важно вынести отдельным элементом (по сути это будет простой предзаготовленный фильтр) - скоропортящиеся продукты. Нажав на данный фильтр видны все скоропортящиеся товары, отсортированные по дате истечения срока годности (самая ближайшая дата в самом начале). Если по конкретному скоропорту занесено несколько накладных, то в общем обзоре берем самую ближайшую дату истечения срока годности.

3.1.3.2 Релиз Бета или Гамма Добавляем просмотр в виде плиток.

3.1.4 Редактирование товара

Для редактирования доступны не только сами товары, но и опции для категорий и подкатегорий. Так например нужно иметь возможность менять цены для всей категории товаров (добавить 30% например). Также очень важно для каждого поля (параметра) товара иметь отдельные права на редактирование. Параметры, доступные для просмотра/редактирования:

- Название товара (только через выпадающий список)
- Количество - только администратор может менять это поле! Обязательно нужна проверка на права!
- Цена на продажу
- Закупочная цена - как и при занесении товара, тут можно оставить как есть либо обновить (опять же - либо абсолютным, либо процентуальным значением)
- Задолженность по товару - только админ может редактировать
- Все выплаты по товару - только админ
- Описание товара

На этой же странице доступна история изменений в товаре.

3.1.4.1 История изменений товара История фиксирует все возможные манипуляции с конкретным товаром. Приход, уход (продажу), изменение параметров, все фиксируется и выводится в формате:

(дата)(значение)(действие)(было)(стало)

Например: 17/04/2015 - количество - приход - было: 0 - стало: 640 Для истории для каждого поля нужно хранить два его значения (действие). Так, для количества это может быть - приход (мы получаем товар) и продажа (мы расстаемся с товаром). Для задолженности это будут - оплата (мы платим по счетам) и кредиторка (мы откладываем платеж, задолженность растет).

3.1.5 Накладные

Отдельным от товаров табом находятся накладные. Здесь виден обзор всех накладных, которые были занесены через "оприходывание товара". Для обзора доступны следующие поля:

- Номер накладной
- Кол-во занесенных в накладную товаров
- Общая закупочная цена по накладной
- Дата поступления накладной
- Сколько оплачено по накладной (берем значение, которое занес пользователь, делим его на общую закупочную сумму)
- Если в накладной есть скоропорт, какая самая ранняя дата истечения срока годности

Просмотр накладной полностью идентичен занесению новой накладной, только с разницей, что значения уже не редактируемы.

3.1.5.1 Фильтры для скоропорта Очень важно для накладных добавить две дефолтные опции фильтра - "показать только накладные со скоропортом" и "показать накладные с просроченным товаром". Данные виды подгружают все накладные со скоропортящимися товарами, а также сортируют их по дате истечения срока годности.

3.1.5.1.1 Просроченные товары Здесь необходимо вывести только товары, истечение срока годности которых уже в прошлом. Также подсвечиваем строки с таким товаром красным цветом.

3.1.5.1.2 Просто скоропорт Здесь выводим все товары, срок годности у которых ещё не истек. Те товары, которые увядают в ближайшие три дня подсвечиваем темножелтым. В ближайшие 7 дней - бледножелтым.

3.2 Касса

Данный модуль позволяет оформить заказ и оплатить его в режиме оффлайн. Для создания заказа у данного модуля есть свой интерфейс, позволяющий продавцу искать необходимые товары и добавлять их в виртуальную корзину. После того, как продавец перешел к оплате он выбирает метод "наличные" или "банковской картой" и в зависимости от выбора срабатывает драйвер кассового аппарата и открывает кассу либо драйвер устройства, которое считывает данные банковской карты.

Для поиска товара продавцу нужен простой интерфейс - поиск товара по названию, а также вывод товаров простым списком, поделенным на категории/подкатегории. Методы оплаты на данный момент:

- Наличный расчет
- Оплата банковской картой
- Безналичный расчет (выставляем счет)

Касса ни в коем случае не должна иметь возможность пробивать больше товара, чем находится в модуле "товары/склад".

3.2.0.2 Идея для релиза Бета/Гамма Может быть здесь сразу сделать вывод шаблона интернет-магазина? Там есть и поиск и вывод товара по категориям/подкатегориям.

3.3 Финансы

Финансы представляют собой обзор дебиторской и кредиторской задолженностей, а также набор отчетов.

3.3.1 Дебиторка/Кредиторка

В табличной форме показываем пользователю задолженность фирмы перед поставщиками и задолженность третьих лиц перед магазином.

3.3.1.1 Дебиторка(должны нам) Рассчитывается следующим образом - товар ушел, денег нет (например продажа по счету).

3.3.1.2 Кредиторка(должны мы) Рассчитываем так - товар оприходован, но не вся сумма оплачена (разница между полями "закупочная стоимость" и "фактическая оплата")

3.3.1.3 Отчеты

3.3.1.3.1 Доходы Все кассовые операции, которые являются фактическими платежами

3.3.1.3.2 Расходы Расходы представляют собой как отчетность, так и настройки по расходам. В настройках по расходам можно заносить фиксированные платежи (аренда, коммунальные услуги...). По умолчанию в расходы попадают закупочные стоимости товаров.

3.4 Сотрудники

Помимо стандартных CRUD действий для самих сотрудников имеем: настройки заработной платы и бонусы. Набор настроек для зарплаты:

- Фиксированная часть
- Проценты
- Период выплат

Для бонусов имеем следующие настройки:

- Опция - с общих продаж или с группы товаров (например только с продаж мяса)

3.5 Клиенты

По аналогии с сотрудниками - базовые CRUD действия для сущности "клиент". Помимо этого имеем привязку прайслиста к конкретному клиенту.

3.5.1 Индивидуальный прайслист

При создании прайслиста можно указать две опции - фикс. цены на каждый продукт или скидки в процентах. При этом для скидок работает ступенчатая модель - приоритетнее всего скидка на индивидуальный товар, за ней скидка на группу товаров, далее скидка на все товары. Пример - на весь товар клиенту ООО Ромашка даются 5% скидка. На алкогольную продукцию даются 10%. Соответственно 10 процентов на алкоголь превалиируют над 5 процентами на все товары в разделе "алкоголь". Скидки не суммируются.

3.6 Задачи

На данный момент просто устанавливаем битрикс-чат.

3.7 Производство

Данный модуль подразумевает собственный интерфейс (аналогично играм вроде "Собери Пиццу"), который позволяет пользователю изъять из оборота (зарезервировать для себя) определенные товары и на выходе получить новый товар.

Пример - изымаем 200 кг мяса, 100 кг муки и 100 шт. яиц. На выходе получаем 1000 бургеров.

3.7.1 Процесс изъятия

Для начала пользователь-производственник подает заявку товароведу на изъятие определенного перечня товара (в заявке указываем причину, конечный продукт, список ингредиентов). Эта заявка находится переходит несколько статусов - "ожидание" "принято" "отклонено (с причиной) "необходимо уточнение".

3.8 Поставщики

Самый обычный список фирм-поставщиков со стандартными CRUD-действиями.

3.9 Группы пользователей

Для разделения прав важно - каждый модуль обладает своими правами доступа. Например работнику производства нельзя давать доступ к модулю "касса а продавцу к модулю "производство". Для товаров важно - разделение прав по категориям товаров. Например сотрудникам мясного цеха нельзя давать доступ к молочной продукции.